4-D Tomographic Inference

Application to SPECT and MR-driven PET

Stefano Pedemonte

- Supervisors -

Prof. Sébastien Ourselin
Prof. Simon Arridge

Doctor of Philosophy

Department of Medical Physics and Bioengineering
University College London
University of London

March 2014

Abstract

Emission tomographic imaging is framed in the Bayesian and information theoretic framework. The first part of the thesis is inspired by the new possibilities offered by PET-MR systems, formulating models and algorithms for 4-D tomography and for the integration of information from multiple imaging modalities. The second part of the thesis extends the models described in the first part, focusing on the imaging hardware. Three key aspects for the design of new imaging systems are investigated: criteria and efficient algorithms for the optimisation and real-time adaptation of the parameters of the imaging hardware; learning the characteristics of the imaging hardware; exploiting the rich information provided by depth-of-interaction (DOI) and energy resolving devices. The document concludes with the description of the NiftyRec software toolkit, developed to enable 4-D multi-modal tomographic inference.

Contents

Notation

Boldface letters are used to denote vector and matrix quantities, while the scalar entries of a vector or matrix are denoted in regular type with subscripts. Curly brackets delimit sets and square brackets delimit vectors and matrices. The list below summarises the most frequently used symbols and their meaning (unless otherwise specified).

$\boldsymbol{\lambda} = [\lambda_1, .., \lambda_v, ..\lambda_{N_v}]$	$\boldsymbol{\lambda} \in \mathbb{R}^{+^{N_v}}$	Radio-pharmaceutical concentration.
$\boldsymbol{q} = [q_1, .., q_d, ..q_{N_d}]$	$\boldsymbol{q} \in \mathbb{N}^{N_d}$	Photon counts.
v	$v \in \{1, .., N_v\}$	Index of the voxels.
N_v	$N_v \in \mathbb{N}$	Number of voxels.
d	$d \in \{1, .., N_d\}$	Index of detector bins.
N_d	$N_d \in \mathbb{N}$	Number of detector bins.
$\mathbf{z} = [z_1, .., z_v, ..z_{N_v}]$	$z_v \in \{1, .., N\}$	Hidden anatomical-functional state in voxel v
N	$N \in \mathbb{N}$	Cardinality of the anatomical(-functional) states.
$\theta_y = \{\theta_{y_1}, .., \theta_{y_N}\}$	$\theta_y \in \mathbb{R}^{+^{2N}}$	Set of class-specific parameters of the MR acquisition system.
$\theta_{y,k} = \{\mu_{\lambda,k}, \sigma_{y,k}\}$	$\theta_{y,k} \in \mathbb{R}^{+^2}$	Class-specific parameters of the MR acquisition system.
$\mu_{y,k}$	$\mu_{y,k} \in \mathbb{R}^+$	First moment of the MR image intensity in class z.
$\sigma_{y,k}$	$\sigma_{y,k} \in \mathbb{R}^+$	Second moment of the MR image intensity in class z.
$\pi = \{\pi_1, .., \pi_k, .., \pi_N\}$	$\pi_k \in \mathbb{R}^{[0,1]}$	Multinomial prior probability associated to a discrete state z.
$\theta_\lambda = \{\theta_{\lambda_1}, .., \theta_{\lambda_N}\}$	$\theta_\lambda \in \mathbb{R}^{+^{2N}}$	Set of class-specific parameters of the pharmaceutical uptake model.
$\theta_{\lambda,k} = \{\mu_{\lambda,k}, \sigma_{\lambda,k}\}$	$\theta_{\lambda,k} \in \mathbb{R}^{+^2}$	Class-specific parameters of the pharmaceutical uptake model.
$\mu_{\lambda,k}$	$\mu_{\lambda,k} \in \mathbb{R}^+$	First moment of the radio-pharmaceutical concentration in class k.
$\sigma_{\lambda,k}$	$\sigma_{\lambda,k} \in \mathbb{R}^+$	Second moment of the radio-pharmaceutical concentration in class n.
$\theta = \{\theta_y, \theta_{\boldsymbol{\lambda}}\}$	$\theta \in \mathbb{R}^{+^{4N}}$	Set of parameters of the anatomical functional model.
$\mathbb{E}[x]$		Expectation of random variable x.
$E(x)$		Shannon Entropy associated to random variable x.
$\Phi(x)$		Gaussian kernel with unit variance and centred in $x = 0$.
$\mathcal{N}(x; \mu, \sigma)$		Normal probability distribution of x, with expectation μ and variance σ^2.
$\mathcal{P}(\lambda, k)$		Poisson probability distribution of k, with expectation λ.
$\frac{\partial f(\mathbf{x})}{\partial x_r}$		Partial derivative of function $f(\mathbf{x})$ (vector-valued $\mathbf{x}$) with respect to x_r.

Chapter 1

Introduction

1.1 PET-MR: The third life of emission computed tomography

1.1.1 PET and SPECT

Emission computed tomography is a type of tomography involving radio-active emission, including Positron Emission Tomography (PET) and Single-photon Emission Computed Tomography (SPECT). PET and SPECT are widely applied in the field of medical imaging for the diagnosis of ischemic heart disease, mapping of local brain metabolism, detection of tumors and areas of infection and drug development.

The PET and SPECT imaging systems detect the gamma photons emitted by a radio-active isotope (tracer), which is introduced into the body on a biologically active molecule (radio-pharmaceutical) or as a pure element. Three-dimensional images of the spatial concentration of the tracer are reconstructed indirectly from the observed high energy gamma radiation emitted in consequence of the decay of the radio-active isotope. By detecting the high energy photons and solving the *inverse problem* that arises, one can measure in vivo the spatial concentration of the tracer, obtaining insight of the metabolic and physiological processes. However, the computation of a solution to the inverse imaging problem presents numerous challenges that frustrate the quantitative assessment.

1. **Uncertainty of the measurements:** The dose of tracer administered to the patient is limited by safety constraints; consequently, the number of photons acquired during SPECT and PET medical scans is of the order of only a few millions to a few hundreds of millions. Discretising the imaging volume in a few millions of voxels, it is immediately understood how the uncertainty associated with the photon counting process constitutes a major limitation in emission computed tomography. While the random variations associated to the photon counting process may be considered less critical than eventual systematic errors, the dramatic extent of such uncertainty, as discussed in this thesis, is perhaps the major problem of emission computed tomography. Adopting a jargon common in the context of photo-detectors engineering, emission computed tomography is *photon limited*.

2. **Computational limitations:** The problem of reconstructing the spatial density of the tracer from the emission signals produced by the scanner can be described as an inverse problem, where the forward problem consists of computing the intensity of light (i.e., as will be clarified, the expected number of gamma photons) that one would observe for a given spatial distribution of the tracer. The forward problem consists in simulating the propagation of light through the propagating medium (the patient in the scanner) and the detection of light by the imaging hardware. In order to invert the problem, one must be able to compute easily the forward problem; inversion algorithms, such as the ones described in this thesis, in fact, typically entail the iterative computation of the forward problem. Simulating accurately the propagation of light is tremendously complex.

3. **Ill-posedness of the inverse problem:** In tomographic reconstruction, one attempts to estimate the rate of emission, continuous function of space and time, from discrete measurements (the photon interaction events). The reconstruction problem appears therefore inherently underdetermined. Furthermore, the design of the imaging hardware involves a trade-off between efficiency of the photon detection process and ill-posedness of the inverse problem.

4. **Mis-characterisation of the imaging systems:** Characterisation of the response of the imaging systems is commonly obtained by numerical simulation or direct measurement by means of arrays of point sources and other specially designed equipment. Both approaches are subject to errors. Furthermore, referring back to 2., computational complexity practically imposes the use of approximated system models. As discussed in this thesis, *scattered photons* and *randoms* may be considered to be in this category, whereas one is forced to adopt simplified forward models due to computational limitations.

5. **Exogenous variables:** The signals produced by the imaging system are affected by exogenous variables, not immediately of interest in the imaging process. In medical applications of Emission Computed Tomography, the optical characteristics of the propagating medium affect the measurement. Lack of knowledge of the spatially varying *attenuation and scatter coefficients* for gamma rays throughout the patient confounds the information that is available about the spatial density of tracer.

6. **Time-dependence:** *Motion* of the patient during the acquisition is an important confounding factor. The complex, generally non-rigid deformations that occur during the acquisition are challenging to model, measure and compute. Motion is perhaps more disruptive in emission computed tomography than in other imaging modalities, such as magnetic resonance imaging (MRI) and X-Ray computed tomography (X-Ray CT), due to the photon limited acquisition regime. Time frames short enough to render negligible the effect of motion, in fact, typically present insufficient information for reconstruction.

1.1.2 PET-CT and SPECT-CT

Hybrid PET-CT scanners, introduced commercially in 2000 [1, 2], and the immediately following SPECT-CT scanners (see [3]), have addressed some of the limitations of PET and SPECT. In these hybrid systems, one or multiple X-Ray sources and detectors are installed aside the detectors for emission imaging. The X-Ray acquisition enables accurate patient-specific estimation of the attenuation map. The use of the X-Ray CT derived attenuation map has greatly improved the quality of the emission images, partly solving problem 5. [4]. Consequently PET-CT and SPECT-CT have been widely accepted in clinical practice and in research laboratories, virtually all new emission imaging systems incorporating the X-Ray imaging device. Additionally to the improved quality of the emission images, the anatomical images provided by the X-Ray scanner have diagnostic value and have proven fundamental for visual localisation, aiding the quantification and the interpretation by human experts of the emission tomograms.

1.1.3 PET-MR: the PET perspective

Following the success of PET-CT and SPECT-CT, the introduction in 2012 of the first commercial PET-MR imaging system (the Siemens Biograph mMR) that integrates PET detectors inside the MR coil, has generated enthusiasm in the medical imaging community. The richness of the information provided by the combined, simultaneous, acquisition of PET and MR signals and the harmlessness of MR are enabling new applications. This thesis explores such new possibilities from the perspective of emission computed tomography.

4-D computed tomography

In order to meet the safety constraints regarding the dose of radiation administered to the patient, the X-Ray source of PET-CT and SPECT-CT scanners can only be practically switched on for a short time interval, typically before the emission scanner is activated. Therefore, PET-CT and SPECT-CT systems are currently designed as sequential systems, with the X-Ray scanner mounted right next to the PET/SPECT gantry. Integrated PET-MR systems enable simultaneous and inherently spatially aligned acquisition of the signals from the two modalities. The harmlessness of the non-ionising magnetic resonance imaging system enables continuous acquisition of information related to the underlying anatomy. Such information may aid the estimation of the spatial density of the tracer and may therefore render feasible four-dimensional (4-D) emission computed tomography.

Improved quantification

The flexibility of magnetic resonance, its harmlessness and the simultaneous and inherently inter-aligned acquisition of the emission information promise to enable new techniques for quantitative and qualitative analysis.

1.2 The unified modelling approach

This thesis addresses the challenges of emission imaging described above, exploring the combination of information from the two modalities in PET-MR. The mathematical formulations presented in this work are inspired by the integrated modelling paradigm outlined in the work of John Ashburner [5] in the context of MR imaging, describing generative models for PET-MR that capture the (hidden) relations between the measurements from the two imaging modalities. Heuristics are dispelled by adopting Probabilistic Graphical Models (PGMs) as a modelling tool and for the development of suitable inference algorithms. The formulations presented in this thesis are unifying in several ways, combining in a single model the information from multiple imaging modalities, and attempting to unify reconstruction and image analysis (registration and segmentation); 3-D and 4-D reconstruction; the parametric and non-parametric approaches; *data-driven* and *sensor-based* motion correction; scatter correction and scatter rejection.

1.3 Document outline

The first two chapters are introductory. Chapter 2 introduces basic concepts of Probabilistic Graphical Models (PGMs), framing tomographic reconstruction in such context. Chapter 3 frames the well known Maximum Likelihood Expectation Maximisation (MLEM) iterative algorithm for the computation of the Maximum Likelihood Estimator (MLE) and its variant Ordered Subsets Expectation Maximisation (OSEM) in the context of Probabilistic Graphical Models. The derivations presented in this introductory chapter serve as the basic ingredients for the formulations presented in the remainder of the thesis. The second part of the chapter traces the link between the imaging process, the *partial-volume effect*, *resolution* and *ill-posedness* of the reconstruction problem, investigating, through numerical experiments, the properties of the Maximum Likelihood Estimator (MLE) and of the Maximum A Posteriori (MAP) estimator with smoothness constraint. The study of the properties of the reconstruction algorithms presented in this introductory chapter serves as the rationale for the subsequent chapters, especially for chapters 4, 5, 6 and 9.

Chapter 4 introduces a modelling paradigm for multi-modal imaging and defines a model for PET-MR based on finite mixtures. Such model is aimed at enabling the use of anatomical information extracted from MR acquisitions to constrain the reconstruction of the emission tomograms.

Chapter 5 addresses the challenge of motion of the patient in the PET-MR scanner. The modelling paradigm described in Chapter 4 is applied to the 4-D tomographic reconstruction problem. The resulting model describes the interaction amongst the longitudinal measurements from the PET and MR acquisition systems. The unifying approach results in a number of (sub)models and algorithms: 1) four-dimensional MLEM reconstruction with non-rigid deformation parameters obtained from external motion capture devices; 2) direct estimation of the motion parameters from the emission data; 3) event-driven motion correction in PET-MR; 4) a model for longitudinal segmentation of MR-images. Such

model, which may be considered a side product of this chapter (as it does not involve PET imaging), extends the Unified Segmentation framework described by John Ashburner [5], addressing the problem of *plausibility* of the registration of longitudinal images.

Chapter 6 extends the finite mixture model described in chapter 4, re-framing the model in the *information theoretic framework* in order to introduce a *semi-parametric* variant. The side product of this chapter is a link between the parametric registration and segmentation method of the Unified Segmentation framework [5] and the Joint Entropy and Mutual Information methods commonly employed for the registration of medical images.

Chapter 7 describes a second type of inverse problems that occurs in emission imaging: the reconstruction of the coordinates of interaction and energy of the gamma photons from the electrical signals produces by nuclear imaging devices. In the first part of the chapter, such inverse problem is framed in the probabilistic (graphical) framework. The second part of the chapter describes two algorithms for the experimental characterisation of depth-of-interaction (DOI) resolving imaging devices. The first algorithm, based on Expectation Maximisation (EM), presents local convergence properties. The second algorithm, based on concepts of *manifold learning*, presents global convergence properties, enabling fully automated characterisation of gamma imaging devices. This new concept brings artificial intelligence near the hardware of the imaging device.

Chapter 8 considers at once the two inverse problems described in chapter 7 and in chapter 2, coupling the model of the emission process with a model of the imaging device. The unified approach suggests two algorithms: 1) an algorithm for automatic adaptation to disrupted detectors; 2) an iterative algorithm for the discrimination of scatter events that makes use of the rich information from DOI and energy resolving detectors and may adapt to the available computating resources.

Chapter 9 extends the investigations on the resolution of Emission Imaging systems presented in Chapter 3, focusing on the challenging problem of optimum system design. *Information*, as defined by Shannon, is adopted as the figure of merit for optimal design of tomographic imaging systems. An efficient algorithm for the approximate calculation of the information retrieval based on the Laplace approximation and sparsification of the Fisher Information Matrix is described and applied to the exploration of new system designs.

Chapter 10 describes the NiftyRec reconstruction toolkit, which implements the algorithms described in the previous chapters. The design of the NiftyRec software is inspired by the unifying approach and by the graphical models described throughout this thesis.

Chapter 2

Probabilistic graphical models

2.1 Brief introduction to probabilistic graphical models

A unified theory of causation and inference has emerged in the last 20 years from the works of Judea Pearl [6], Spirtes [7], Scheines [7][8] and others. In this context, *probabilistic graphical models* (PGMs) combine principles from probability theory, graph theory and computational complexity theory to provide a powerful declarative representation of systems, where semantics are clearly separated from the inference algorithms. While probabilistic graphical models have permeated and revolutionised the field of computational science, they have not been applied systematically to tomographic imaging.

The models and algorithms introduced in this thesis, though addressing different aspects of computed tomography, are described in a unified mathematical framework, adopting the formalism of Probabilistic Graphical Models (PGMs). This first chapter introduces some basic concepts and terminology of PGMs that will be adopted in the remainder of this thesis and frames the problem of tomographic reconstruction in the context of PGMs. The concise introduction of PGMs that follows is meant to enable the understanding of the concepts utilised in the subsequent chapters for the definition of models and for the development of inference algorithms in the context of emission computed tomography.

2.1.1 Directed acyclic graphs and Markov random fields

Probabilistic graphical models are graphical constructs that represent knowledge about an uncertain domain. A *graph* is a representation of a set of objects where some pairs of the objects are connected by links. The interconnected objects are represented by mathematical abstractions called *vertexes* or *nodes*, and the links that connect the pairs of nodes are called *edges*. The edges of a graph represent relations between the nodes and may be undirected to express a symmetric relation, or directed to express an asymmetric relation. When the vertexes of a graph are associated to *random variables*, the graph assumes the denomination of *probabilistic graphical model* (PGM). The edges of a PGM express the probabilistic dependence between the random variables associated to the graph. In the context of probabilistic graphical models, a graph whose edges are all directed is denominated *directed acyclic graph* (DAG), while a graph whose edges are all undirected is denominated *Markov Random Field* (MRF).

A mixed graph, in which some edges are directed and some are undirected, is often referred to, in the context of probabilistic graphical models, as *hybrid Bayesian network*.

2.1.2 Conditional independence

One of the basic concepts of probabilistic reasoning is the relation of *dependence/independence* between random variables. Two variables subject to uncertainty are considered independent if knowledge of the state of one of the two variables does not introduce any information about the state of the other variable. The variables are dependent if knowledge of the state of one of the two variables narrows the uncertainty associated to the other variable. E.T. Jaynes, in his seminal work that has culminated in the Principle of Maximum Uncertainty [9], has exposed the idea that the very concept of dependence and independence of random variables implies that probability is *subjective*. The essence of the Bayesian framework is indeed the subjectivity of probability: the uncertainty associated to a variable depends on the state of knowledge of the observer, on the variables that she/he/it has observed.

The ideas of E.T. Jaynes have greatly contributed to the development of the modern conception of probability. Not only the uncertainty about the state of a variable is subjective (dependent on what the observer knows about other variables that may be related by probabilistic dependence to the variable of interest), but the relations of dependence and independence are equally subjective. Kiiveri and Speed in 1982 [10] have defined the concept of *conditional independence*. They formulated the idea that if variable Y is not a cause of X and X influences Y, if at all, only through an intermediary set Z of direct causes of Y, so that if the variables in Z are held constant, no variation in X will produce a variation in Y, then X and Y are *independent conditionally* on Z. Such relation between X, Y, Z, indicated with $X \perp Y|Z$ (X is independent from Y when Z is known), implies that if one knows Z, knowledge of the state of X does not affect the state of knowledge about Y.

Expressing the relations of conditional independence is essential in order to reason about complex systems and, as it turns out, in order to devise computationally efficient inference strategies. The joint probability distribution of two independent variables *factorises* into the product of two terms. Equivalently, as discussed in the following, relations of conditional independence correspond to *factorisations* of the *joint probability distribution* of the variables. PGMs express precisely the set of conditional independence relations between the variables. The next paragraph adopts the *causal* point of view to introduce a methodology to define probabilistic models (*model entry*) and describes, with the aid of three examples, the relation between the set of conditional independencies associated with a set of variables and the structure of the associated directed and undirected graphs.

2.1.3 Causal modelling

Probabilistic graphical models may be used to represent causal relationships, associating a directed edge to a relation of causality. Though causal interpretation of probabilistic graphical models may be considered here a matter of philosophical disquisition (see e.g. [8]), acquiring indeed more relevance in the

context of *causal inference* (i.e. problems in which one attempts to infer the structure of the graph), for our purpose it suffices to recognise, from the simple examples that follow, that a set of causal relations implies a unique set of conditional independencies; however, a given set of conditional independencies does not imply a unique set of causal relations. A hint of the formal definition of causality in the context of graphical models is given at the end of this paragraph; let us start out with some examples to clarify why we are interested in the causal interpretation of graphical models. The examples in Fig. 2.1 and the discussion that follows in paragraph 2.1.5 clarify that a graph of the type $X \rightarrow Y \rightarrow Z$ is equivalent to a graph $X \leftarrow Y \leftarrow Z$, encoding the same set of conditional independencies, though representing different causal relations. The difficulty with determining the causal relations from multiple observations of a set of random variables (causal inference), a problem that is not considered in this work, arises from the non-bijective correspondence between the set of the causal relations and the set of the conditional independencies. However the causal interpretation guides the process of *model entry*. If one knows or assumes the causal relations, the graph (the set of conditional independencies), is uniquely defined.

In order to obtain the set of the conditional independencies implied by the relations of causality, one constructs a DAG by drawing directed edges from cause to effect. Fig. 2.1 reports three simple graphical models of three variables. The DAGs encode, from left to right, a nested causal effect, a common cause and a common effect. For these simple examples, it is relatively straightforward to encode the relations of conditional independence implied by the causal structure. One may think of common problems that are known to have such causal structures and verify the relations of conditional independence reported in Fig. 2.1. An example of nested causal effect (Fig. 2.1-left) is $short\ circuit \rightarrow fire \rightarrow fire\ alarm$; an example of common cause (Fig. 2.1-centre) is $yellow\ fingers \leftarrow heavy\ smoker \rightarrow cancer$ and an example of common effect (Fig. 2.1-right) is $time\ of\ the\ year \rightarrow average\ rainfall \leftarrow country$. One can easily verify the conditional independence clauses reported in Fig. 2.1. When hearing the fire alarm, one may think that there is a chance that a short circuit has occurred, associating a larger probability to such event than when the fire alarm is off ($X \neg \perp Z|\emptyset$ in Fig. 2.1-left) (the symbol $\neg \perp$ reads 'is not independent from'). If one sees fire, however, hearing the fire alarm does not add any information regarding the eventuality of a short circuit and being told that there was a short circuit does not add any information regarding the eventuality of the fire alarm being activated ($X \perp Z|Y$ in Fig. 2.1-left). If one does not know whether an individual is a smoker or not, observing that he has $yellow\ fingers$ affects the probability associated to the event that he has or will develop $cancer$. Also, one would conclude that the fingers are likely to be yellow if he has $cancer$, or at least they are more likely to be yellow than if he does not have $cancer$. However, if the observer knows that the individual is a $smoker$, looking at the colour of the fingers does not affect his state of knowledge about the eventuality of $cancer$. And in this case, the notion that the individual has $cancer$ does not affect the probability of $yellow\ fingers$. In other words, whatever the state of variable $smoker$, when such variable is known, $yellow\ fingers$ and $cancer$ are independent ($X \perp Z|Y$ in Fig. 2.1-centre), though they are not independent for the

observer who does not know the state of variable $smoker$ ($X \neg\perp Z|\emptyset$ in Fig. 2.1-centre). Considering an unknown $time\ of\ the\ year$ and $country$, if one is told the average $rainfall$ for the period under consideration, then being told the name of the country gives information about the $time\ of\ the\ year$ and equivalently, being told what is the $time\ of\ the\ year$ gives information about which $country$ it might be ($X \neg\perp Z|Y$ in Fig. 2.1-right). However, a priori, $time\ of\ the\ year$ and $country$ are not related ($X \perp Z|\emptyset$ in Fig. 2.1-right).

For simple problems of three variables, in conclusion, one can intuitively determine the conditional independence implied by the causal structure of the problem. For problems of arbitrary complexity, determining the relations of conditional independence between the variables from the causal structure is not always straightforward. Paragraph 2.1.5 describes how the set of conditional independencies, or equivalently the factorised expression of the joint probability distribution (and eventually its marginals), can be obtained from the DAG constructed by assigning directed edges to each relation of causality. The conditional independence clauses reported in Fig. 2.1 and justified in this paragraph via simple examples, can be obtained algorithmically with the methodology described in paragraph 2.1.5.

As a final remark, for the purpose of *model entry*, it is sufficient here to accept that a known causal structure implies a unique set of conditional independencies and that these are encoded by the DAG obtained by associating a directed edge to causal relations. A formal definition of *causality* is given e.g. in [8] and is based on *response to intervention*; relations of causality can be assessed only if the variables can be *controlled* independently. Whether smoking is a cause of cancer is not clear until one controls the variables independently in a controlled randomised experiment. While variables are not always controllable, for the observation model of emission imaging systems, described in detail in paragraph 2.3, there is no doubt that manipulation of the radio-activity implies a change in the probability distribution associated with the photon interaction events, while manipulation of the photon interaction events (i.e. forcing the values produced by the read-out electronics), does not affect the decay rate of the radio-pharmaceutical.

2.1.4 Completeness

One concept that is relevant for the derivations in the next chapters is that of *completeness* of causal models. As pointed out by Scheines [8], a causal graph is *incomplete* in the sense that not all causes need to be in the graph in order to be able to perform inference and must be *complete* in the sense that all the causal relations between the variables that are specified need to be represented in the DAG. This observation can be clarified with the classical example of the smoker [8]. Referring to the $yellow\ fingers \leftarrow smoker \rightarrow cancer$ example, the consideration of Scheines on completeness is that one can still reason about the variables when the model does not account for $air\ pollution$: the model can be *incomplete*; but if $air\ pollution$ is included in the model, in order to perform inference correctly, the graph has to be *complete*, specifying all the causal relations.

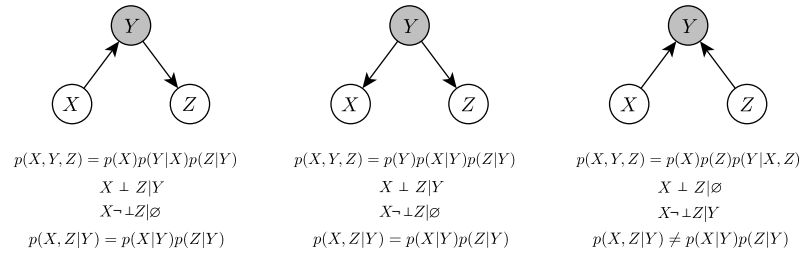


Figure 2.1: Directed Acyclical Graphs; under the causal interpretation, the three graphs represent, from left to right, indirect causal effect, common cause and common effect. The relations of conditional independence and the equivalent factorisations encoded in the three graphs are reported in the caption. Paragraph 2.1.3 reports examples that clarify how causality implies the relations of conditional independence. Paragraph 2.1.5 describes a procedure to extract the relations of conditional independence from a DAG via *moralisation* of the graph.

2.1.5 d-separation, moralisation and the Markov blanket

The causal structure of a problem can be encoded into a DAG by drawing a directed edge in correspondence of each causal relation. The DAG encodes the conditional independencies between the variables. For simple graphs, such as for the graphs in Fig. 2.1, which reports all the possible DAGs for the interaction between three variables, it is relatively straightforward to infer the conditional independencies. One can think of familiar examples for each case, to verify the conditional independencies reported in Fig. 2.1. However is it not always straightforward, given a DAG, to infer whether two variables are conditionally independent, given the state of a third variable or of a set of variables. Given a graph $\mathcal{G}$ and three sets of variables $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ corresponding to sets of nodes of the graph, one wants to know, e.g., if $\mathbf{X}$ is independent from $\mathbf{Y}$ when $\mathbf{Z}$ is known. This information is encoded in $\mathcal{G}$. Conditional independence, named *d-separation* in the context of DAGs, can be obtained algorithmically from the structure of the DAG. Procedures to calculate the d-separation for any three subsets are described in [11] - pp. 71. Such procedures, though useful e.g. to automate, by means of a computer, the extraction of the structure of a graph, e.g. from observations, are not easily applicable to determine by hand the set of conditional independencies of a graph. A simple alternative to calculating the conditional independencies from the DAG is to transform the DAG into a Markov Random Field (MRF), represented by a graph with non-directed edges. MRFs in fact provide a definition of conditional independence that is operatively simpler to apply than *d-separation*. DAGs and MRFs do not in general encode the same sets of conditional independencies (see e.g. [11] - pp. 69). That is, if one considers all possible graphs, where a given graph is identified by the set of the conditional independencies that it encodes, DAGs and MRFs may encode only subsets of all possible graphs and the two subsets do not coincide. However, the mapping between graphs and probability distributions is non injective; a graph encodes a certain probability distribution if the set of conditional independencies implied by the graph is a *subset* of the conditional independencies of the

distribution. In other words graphs may be intended as filters of probability distributions: the complete undirected graph, where every node is connected with every other node (the set of conditional independencies is empty) may encode any distribution. If $\mathcal{I}(\mathcal{G})$ is the set of conditional independencies implied by graph $\mathcal{G}$ and $\mathcal{I}(P)$ is the set of conditional independencies of the distribution P, $\mathcal{G}$ represents P if $\mathcal{I}(\mathcal{G}) \subseteq \mathcal{I}(P)$. This concept will be more clear in the next few chapters, where analytical expressions are associated to the factors of the probability distributions. Disregarding one or more, or eventually all, conditional independencies, may however result in over complicated inference algorithms.

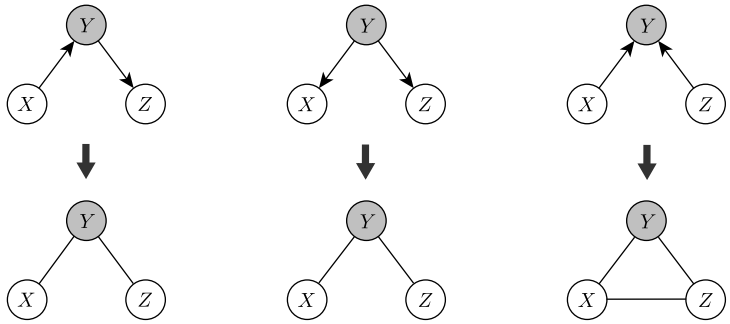


Figure 2.2: *Moralisation* of a DAG and *global Markov properties*. Moralisation consists in replacing all edges by undirected edges and connecting all parents of a same child. Global Markov properties: the relations of conditional independence can be easily extracted from the Markov Random Field obtained by moralisation, even for large graphs: two nodes or sets of nodes X and Z are conditionally independent given a set of variables Y, if, shading the nodes corresponding to the given variables, there is no path that connects X and Z and does not cross any shaded node. The three graphs reported in figure are the moralised graphs of the DAGs of Fig. 2.1. The relations of conditional independence reported in of Fig. 2.1 are immediately verified to correspond to the global Markov properties.

The *moralised* undirected graph $\mathcal{M}[\mathcal{G}]$, obtained by *connecting all parents of a same child* in $\mathcal{G}$ *and removing all arrows*, can be shown to be such that $\mathcal{I}(\mathcal{M}[\mathcal{G}]) \subseteq \mathcal{I}(\mathcal{G}) \subseteq \mathcal{I}(P)$. In other words, one can obtain the set of conditional independencies of the probabilistic model from the moralised graph; these correspond to the conditional independencies implied by the causal structure that corresponds to the DAG, though some of the conditional independencies might be lost. However, all relations of conditional dependence are preserved (Fig. 2.2). More specifically, the conditional independencies corresponding to the structures known as *v-structures* are lost.

The undirected graph, when interpreted as a probabilistic model, is denominated Markov Random Field (MRF). The expression *global Markov property* is used in the context of MRFs to express conditional independence: two sets of variables X and Y are conditionally independent (i.e. possess the global Markov property) given a separating subset Z if every path from a node in X to a node in Y passes through Z. The global Markov property is simple to verify even for large and complicated graphs

and will be used extensively in the derivations in the next chapters. It is straightforward to extract the set of conditional independencies from the DAGs of Fig. 2.1 via moralisation and extraction of the global Markov properties (see Fig. 2.2).

A property of the variables associated to a node or to a set of nodes that will be used extensively in the subsequent chapters is the *Markov blanket*. Referring to Fig. 2.3, let us consider the variables at node X. The Markov blanket is defined as the set of variables that, when known, make X independent from every other set of nodes in the graph. In other words, the Markov blanket of X is the only knowledge needed to predict the state of X. In a directed acyclic graph, the Markov blanket of X is the set of nodes composed of X's parents, its children, and its children's other parents. In a Markov random field, the Markov blanket of a node is simply its set of neighboring nodes. Inspecting Fig. 2.3 one may verify that the Markov blanket of a node can be calculated, equivalently, from the DAG or from the Markov random field obtained by moralisation of the DAG.

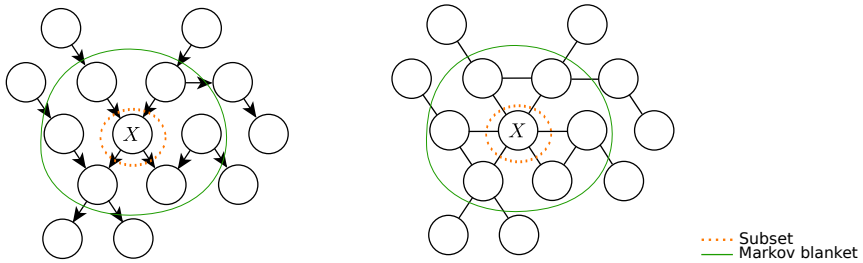


Figure 2.3: Markov blanket: X is independent from all sets of nodes in a graph conditionally to its Markov blanket. The Markov blanket of a node can be calculated, equivalently, from the DAG or from the Markov random field obtained by moralisation of the DAG, as explained in 2.1.5.

2.1.6 Model entry

A probabilistic model is essentially a joint probability distribution over a set of variables (some of the variables may be considered unknown and some may be considered observations). Probabilistic models have a structure; the structure, which can be represented as a graph, corresponds to factorisations of the joint probability distribution. A fully connected graph encodes a probability distribution that cannot be expressed as a product of terms (i.e. it does not factorise). If the joint probability distribution factorises, the probabilistic model is expressed by a collection of (conditional) probability distributions. Given a set of variables (observations and unknown variables), the first step towards performing statistical inference is the choice of a probabilistic model. The choice of a suited probabilistic model may be obvious when the variables are regulated by simple physical phenomena, but it is often less obvious. In the following we will refer to the choice of a probabilistic model as *model entry*. Model entry comprises the choice of the structure of the model and of the functional forms of the factors. As discussed in paragraph 2.1.3, relations of causality correspond to factorisations of the probability distributions. Although estimating

the relations of causality between variables is generally a difficult task, when relations of causality are known, these correspond to a set of factorisations (i.e. conditional independencies). The approach to model entry adopted in this thesis is to assume relations of causality. The analytical expression of the joint probability distribution of all the variables is then obtained via the Directed Acyclical Graph and the Markov properties of the moralised Markov Random Field, as described in paragraphs 2.1.3 and 2.1.5. The criteria for the choice of the functional forms of the factors vary. The choice of the functional form can be obvious when the variables are regulated by simple physical phenomena. In other less informed cases, the main criterium adopted in this work is to chose the probability distribution that yields the simplest inference algorithm. In summary, the models described in this work are obtained with the following procedure:

1. Assume a causal structure (relations of causality between the variables);
2. Encode the DAG from the causal structure;
3. Transform the DAG to a MRF by moralisation;
4. Obtain the set of conditional independencies from the MRF (global Markov properties);
5. Express the conditional independencies as factorisations of the joint probability distribution;
6. Assign functional forms to the conditional probability distributions;
7. Derive, from the factorisations, expressions of the conditional, marginal, and joint probabilities associated to the variables.

2.1.7 Notations

As discussed in the examples in paragraph 2.1.3, variables may be observed or unobserved. Observed variables are denoted, by convention, as shaded nodes, as in Fig. 2.4. In a Bayesian model, every unobserved variable is associated to a prior probability distribution. When the prior probability distribution is not expressed explicitly, it is implicit that the prior probability distribution is uninformative, that is all configurations of the variables are assumed to have the same probability. While the prior probability distributions associated to the unobserved variables of a Bayesian model are often omitted and therefore implicit, the explicit representation in the graphical models may clarify the factorisation of the prior probability distributions. In this document, the prior probability distributions of the unobserved variables are therefore, in most places, expressed explicitly in the graphical models by means of a dot (•). Fig. 2.4-left, for example, reports the prior probability distribution of a vector-valued variable $\mathbf{x} = [x_1, x_2, \ldots, x_i, \ldots, x_N]$. One dot per node in figure 2.4-left would have signified that the state of x_i is independent a-priori from the state of x_j. The first order quadratic Markov Random Field described in the next chapter is an example of a prior probability distribution that does not factorise in such a way,

therefore it is represented with a single dot as in Fig. 2.4-left. Notice, however, that, as explained, graphs may be considered filters of probability distributions; the full graph, i.e. a single dot connected to all variables (as in Fig. 2.4-left), expresses any prior distribution for $\mathbf{x}$.

Another convention adopted throughout this document is to express vector-valued variables concisely as in Fig. 2.4-centre and Fig. 2.4-right. Though the concise forms of the graphical model are not always as expressive as the expanded form, they are used for notational convenience when details about the conditional independencies of subsets of the variables are unimportant.

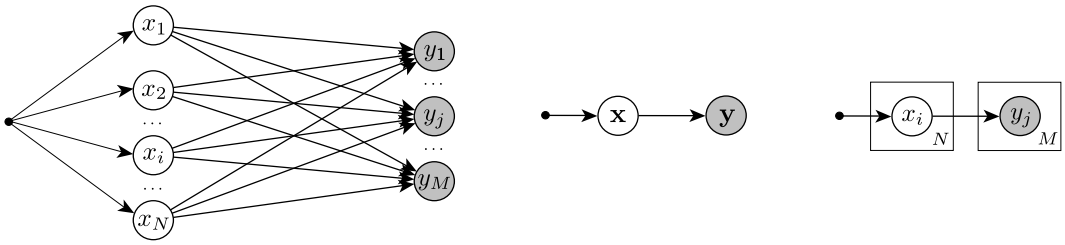


Figure 2.4: Prior probability: prior probabilities of unobserved variables are indicated with a dot. Compact notations: the graph on the left can be represented in vector form with the graphs in the centre and on the right. The notation on the right explicits the dimensionality of the vectors. This document makes use of both compact notations.

2.1.8 Inference

A Bayesian model expresses the probabilistic dependence amongst a set of variables. In general, some of the variables associated to a model are considered to be observed (i.e. their state is known exactly) and other variables are considered unobserved. The Bayesian model allows the observer to express analytically or compute numerically the probability associated to different possible states of the variables that are not observed. Computing the probability associated to the possible states of the unobserved variables is in general referred to as *inference*.

As will be evident in the chapters that follow, the structure of the probabilistic model affects the complexity of the inference algorithms. In the case of two dependent unobserved variables, the state of knowledge about one of the variables depends on the state of knowledge about the other variable, therefore the probability associated to a given state for one of the two variables is a function of the state of the other variable. If the two unobserved variables are independent, the probability associated to the state of one of the two variables does not depend the state of the second variable, therefore the joint probability distribution factorises into a product of two terms. In the case of more than two variables, the relations of (now conditional) dependence and independence are more complex but nevertheless they imply a factorisation of the joint probability distribution associated to the unobserved variables.

The factorisation of the joint probability distribution of the variables of the model, which may be

obtained from the properties of the associated graph, affects the design of the inference algorithms, playing an important role in determining their complexity. If, for example, the variables of interest are independent from one another given the observations, then the state of each of the unobserved variables, trivially, can be computed independently from the state of the other unobserved variables. Another problem, which applies in the case of computed tomography and which is the focus of section 2.3, is one in which the structure is such that the observations are independent conditionally to the unknown variables. Though, in this case, the state of one of the unobserved variables may no longer be computed independently from the state of the other unobserved variables, the knowledge that the observations would be independent if one knew the state of the unobserved variables turns out to simplify inference.

The graphical model encodes the factorisation of the joint probability and its marginals, playing a fundamental role in the development of the inference algorithms. Regardless of the structure of the problem, inference may take many forms.

Many forms of maximum a posteriori

Let us define an experiment as the observation of a set of variables. In the Bayesian framework, after performing the experiment, the posterior probability distribution of the unobserved variables, whose factorisation may be represented as a graph, characterises the state of knowledge of the observer. The observer may be interested in the state of all the unobserved variables, or in a subset. The probability distribution associated to a subset of the unobserved variables is given by the marginalisation of the posterior probability distribution over the variables not considered of interest.

Numerical characterisation of the posterior (or marginal of the posterior) may take many forms, such as the extraction of random samples (posterior sampling techniques), the estimation of the parameters of a constrained posterior probability distribution (Laplace approximation, variational methods), and maximum a posteriori estimation (MAP). MAP is perhaps the simplest synthetic characterisation of a posterior probability distribution: one considers only the point that maximises the posterior probability distribution, disregarding all other solutions and their probability. Depending on whether one is interested in the state of all the unknown variables or only in a subset, there are multiple forms of MAP estimation for a given model and set of experiments. Referring to Fig. 2.5:

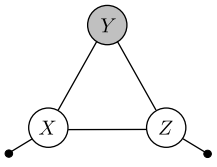


Figure 2.5: Simple model of 3 variables (or sets of variables). X and Z are unobserved, while Y is observed. One may be interested in both X and Z, or only in one of the two.

1. **Marginal MAP**: maximiser of the marginal posterior probability distribution of a subset of the variables:

$$\hat{X} = \underset{X}{\operatorname{argmax}} \int p(X, Z|Y)\mathrm{d}Z \tag{2.1}$$

and eventually:

$$\hat{Z} = \underset{Z}{\operatorname{argmax}} \int p(X, Z|Y)\mathrm{d}X \tag{2.2}$$

2. **Joint MAP**: maximiser of the joint posterior probability distribution of the unobserved variables:

$$\hat{X}, \hat{Z} = \underset{X,Z}{\operatorname{argmax}}\, p(X, Z|Y) \tag{2.3}$$

3. **Approximate marginal MAP**: maximiser of the posterior probability distribution of a subset of the unknowns conditional to the maximiser of the marginal probability of the remaining unknowns:

$$\hat{Z} = \underset{Z}{\operatorname{argmax}} \int p(X, Z|Y)\mathrm{d}Z \tag{2.4}$$

$$\hat{X} = \underset{X}{\operatorname{argmax}}\, p(X|Y, \hat{Z}) \tag{2.5}$$

This may be considered an approximation of the marginal MAP (1.), where one is interested only in the state of X, not in Z:

$$\hat{X} = \underset{X}{\operatorname{argmax}} \int p(X, Z|Y)\mathrm{d}Z \tag{2.6}$$

This can be rewritten as:

$$\hat{X} = \underset{X}{\operatorname{argmax}} \int p(X|Y, Z)p(Z|Y)\mathrm{d}Z \tag{2.7}$$

with $p(Z|Y) = \int p(X, Z|Y)\mathrm{d}X$. If the marginal posterior of Z is a narrow distribution around its mode $\hat{Z}$ (2.4):

$$p(Z|Y) \approxeq \delta(Z - \hat{Z}) \tag{2.8}$$

Substituting in (2.7), one obtains the expression (2.5).

For certain models in which one is interested in X, it is relatively simple to maximise the marginal posterior of Z obtained by integrating over X, and to compute the maximiser of the conditional probability distribution of X given Z, but it is much more difficult to compute a maximiser of the marginal posterior of X obtained by integrating over Z. For such models, amongst which are finite mixtures (considered in more detail in the appendix of chapter 4), this approximation enables the otherwise untractable estimation of the maximiser of the marginal posterior probability of X.

Whether 1. is better than 2. depends on the purpose of the inference task, therefore there is not always a clear answer.

If one considers all possible graphs and functional forms for the factors of the associated probability distributions, the resulting probabilistic model may be applied to problems of arbitrary complexity. Algorithms for the calculation of the MAP estimates vary greatly. While there are some general purpose algorithms for inference in graphs of discrete variables and for particular structures of the graphs, such as trees, in general one has to device an ad-hoc inference procedure.

In the following chapters each model is accompanied by the derivation of a suited optimisation algorithm for numerical calculation of the MAP estimates. The next two paragraphs describe two recipes for the development of optimisation algorithms used throughout this document: Expectation Maximisation (EM) and Iterated Conditional Modes (ICM).

While most of this work describes procedures for MAP estimation, chapter 9 explores approximated fully Bayesian inference.

Maximum Likelihood Estimation

When the prior probability distribution over the variables of interest is considered uninformative (i.e. all states are assigned an equal probability a priori), then the maximum a posteriori (MAP) estimate is often referred to as *maximum likelihood* (ML) estimate.

2.1.9 The Expectation Maximisation algorithm (EM)

It is in general not easy to maximise numerically the posterior of X when the model contains hidden variables, due to the integral in (2.1). If the model does not include hidden variables, the maximisation of the posterior of X is generally tractable when $p(X|Y)$ belongs to the *exponential family* or is a product of distributions that belong to such family. In this case, in fact, one may resort to maximising the $\log$ posterior, cancelling the exponentials with the log. A recurring situation is one in which the likelihood $p(Y|X)$ (i.e. the probability distribution that one would associate to the variable that has been observed if the state of the variable that has not been observed were known) belongs to the exponential family. In this case, the posterior probability distribution $p(X|Y)$ belongs to the exponential family if the prior probability distribution $p(X)$ belongs to the exponential family. Let us apply the fundamental law of total probability:

$$p(Y, X) = p(Y|X)p(X) = p(X|Y)p(Y) \tag{2.9}$$

Rearranging the terms gives Bayes formula:

$$p(X|Y) = \frac{p(Y|X)p(X)}{p(Y)}, \tag{2.10}$$

with

$$p(Y) = \int p(X, Y)\mathrm{d}X = \int p(Y|X)p(X)\mathrm{d}X \tag{2.11}$$

If we are interested in the MAP estimate of X:

$$\hat{X} = \arg\max_X p(X|Y), \tag{2.12}$$

taking the $\log$ of $p(X|Y)$ and substituting (2.10) in (2.12) (note that the denominator of (2.10) is not a function of X), we obtain:

$$\hat{X} = \arg\max_X \log p(Y|X) + \log p(X) \tag{2.13}$$

Therefore, if the likelihood $p(Y|X)$ belongs to the exponential family, the choice of a prior probability distribution $p(X)$ that belongs to the exponential family makes the maximisation problem tractable.

If the model includes hidden variables, the posterior distribution has an integral form. Referring to Fig. 2.5 and applying again Bayes formula (2.10):

$$\hat{X} = \arg\max_X \log \int p(Y, Z|X)\mathrm{d}Z + \log p(X) \tag{2.14}$$

This maximisation is problematic due to the $\log$ appearing before the integral. Even if the integrand $p(Y, Z|X)$ belongs to the exponential family, this expression cannot be simplified. The Expectation Maximisation (EM) algorithm, described by Dempster and Laird [12] and others, however, provides a simple update formula for the maximisation of the $\log$-posterior of X for problems for which $p(Y, Z|X)$, often denominated *complete-data likelihood*, is easy to maximise with respect to X (e.g. if it belongs to the exponential family or is a product of terms that belong to the exponential family).

The intuition behind EM is that if it is easy to maximise the complete data $\log$-likelihood $p(Y, Z|X)$ with respect to X, then if one knew Z, it would be straightforward, (given a good choice of the prior, to maximise the posterior of X, proportional to the complete-data likelihood and to the prior probability: $\log p(X|Y, Z) \propto \log p(Y, Z|X) + \log p(X)$. However Z is not known, but if one knew X, the model would express the probability distribution of Z. The EM algorithm then consists in assuming a provisional value for the most probable state of X and then maximising the *expectation* of the $\log$-posterior of X with respect to the posterior distribution of the hidden variable Z calculated with the current estimate of X:

- Start with an initial estimate of X:

$$X^{(n)} = X^{(0)} \tag{2.15}$$

- Expectation:

$$p(Z|Y, X^{(n)}) = \frac{p(Y, X^{(n)}|Z)p(Z)}{p(Y, X^{(n)})} \tag{2.16}$$

- Maximisation:

$$X^{(n+1)} = \arg\max_X \int p(Z|Y, X^{(n)}) \log p(Y, Z|X)\mathrm{d}Z + \log p(X) \tag{2.17}$$

Demonstration of convergence of $X^{(n+1)}$ to the maximiser of the posterior probability distribution $p(X|Y)$ is reported, e.g. in [12] and [13]-pag. 450-455. In this document, the EM algorithm is utilised in a number of occasions, including the optimisation of the activity in emission computed tomography (chapter 3, paragraph 3.1.1), the optimisation of the parameters of Gaussian mixture models (chapter 4, appendix 4.A) and of Poisson mixture models (chapter 7, paragraph 7.4). In paragraph 3.1.1, appendix 4.A and paragraph 7.4, the update formulae for iterative MAP estimation of the parameters of the respective models are derived, step by step, from equations (2.15)-(2.17). The reader is invited to consider such derivations as examples to clarify the use of (2.15)-(2.17).

Note on the many applications of EM:

More than a specific algorithm, EM can be considered a recipe for the development of algorithms. The first two applications of interest in this work, well known in the field of medical imaging, refer to its use for the Poisson model for tomographic imaging and for the optimisation of the parameters of finite mixture models. The application of the EM recipe to different models produces quite different algorithms dependeing on the model. Interestingly, in both the aforementioned cases, direct optimisation of (2.14) is actually tractable for two different reasons. In the case of finite mixture models, the integral in (2.14) is substituted by a finite sum, yielding a simple form of the gradient of the $\log$-posterior, which enables gradient descent type optimisation. In the case of the Poisson model, the likelihood of the photon counts Y given the photon counts X indeed has a simple analytical form that does not entail any hidden variable. While the gradient of the $\log$-likelihood (and eventually of the $\log$-posterior) is again differentiable and easy to compute, one adds, somewhat artificially, hidden variables (data augmentation) for the purpose of obtaining an EM update formula. In both such applications, described in 3.1.1 and 4.A, EM produces gradient type optimisation algorithms with the advantage of being free from nuisance parameters (step size of the optimisation algorithm), providing essentially automatic line search, and also guaranteeing positivity of the solution where required.

2.1.10 The Iterated Conditional Modes algorithm (ICM)

The mathematical optimisation problem that arises from MAP estimation can be solved with a variety of algorithms, depending on the characteristics of the posterior distribution. When the optimisation problem involves multiple variables (e.g. joint MAP estimation or marginal MAP estimation of a multivariate set), it may be addressed by *alternating optimisation*. The simple idea underlying alternating optimisation is to replace the sometimes difficult joint optimisation of the cost function over all variables with a sequence of easier optimisations involving grouped subsets of the variables.

Referring to the hidden variables model described in paragraph 2.1.9, one may formulate the problem of computing the MAP estimate of X, entailing, as described in 2.1.9, the integration over the hidden variables Z; another inference task for such a model is the estimation of the joint state of X and Z that

maximises their posterior probability distribution. Such a model seems to provide a natural distinction between X and Z, suggesting possibly to alternate the optimisation of the two sets of variables. However it has to be noted that the problem, from the perspective of mathematical optimisation is not different from the problem of maximising the posterior with respect to X alone. In the case of joint estimation, in fact, trivially, X and Z can be renamed as X'. The only difference that occurs between the two problems is that in the case of the optimisation with respect to X alone, the expression of the posterior may be more complicated due to the integral over Z. In both cases the optimisation problem can be addressed by alternating optimisation; in the case of optimisation of X alone, vector-valued X can possibly be divided in subsets that are optimised in sequence; in the case of joint optimisation of X and Z, both X and Z can possibly be divided in subsets. However the natural separation of the unknowns X and Z that appears in the model often makes the alternating optimisation of such subsets the simplest alternative.

In the context of MAP estimation, it is important to stress that two different choices are involved when devising the inference strategy. The first choice relates to the type of inference task that one is interested in. Joint MAP estimation is often simpler to perform than MAP estimation of one variable (or a subset of the unknown variables) alone. It is not always obvious what type of inference is preferred, however the decision to calculate the MAP estimate of only a subset of the variables, integrating the posterior over all the other unknown variables, corresponds to the statement that one is not interested in the state of the remaining unknowns. The second choice relates to the type of optimisation algorithm and includes the possibility to alternate the optimisation of the variables to be estimated, or to optimise for all the variables at once. The combination of these two choices produces different algorithms for common models. For example, in the Gaussian mixture model described in chapter 4, 5, one may estimate the parameters of the Gaussian functions, marginalising over the unknown discrete labels, or estimate the most probable value of the labels jointly with the parameters of the Gaussians. In the first case, considering the discrete labels as hidden variables, the optimisation may be addressed with the EM algorithm. In the second case, a simple optimisation algorithm can be obtained by alternating the optimisation of the parameters of the Gaussian functions with the optimisation of the discrete labels, an algorithm often referred to as *hard EM*.

In the probabilistic setting, alternating optimisation is often referred to as *Iterated Conditional Modes* (ICM), name given by Besag in [14]. Alternating the optimisation of the posterior with respect to subsets of the variables is equivalent, in the probabilistic setting, to the sequential optimisation of the probability distribution of each of the subsets, *conditional* to the provisional estimates of the remaining variables. The graphical model aids the development of ICM algorithms. By definition, the Markov blanket of a node expresses the set of variables that, when known, make the node independent from any other variable. In other words, the expression of the probability distribution of the variables associated to the node, conditioned to all other variables of the model, depends only on the variables that belong to the Markov blanket of that node.

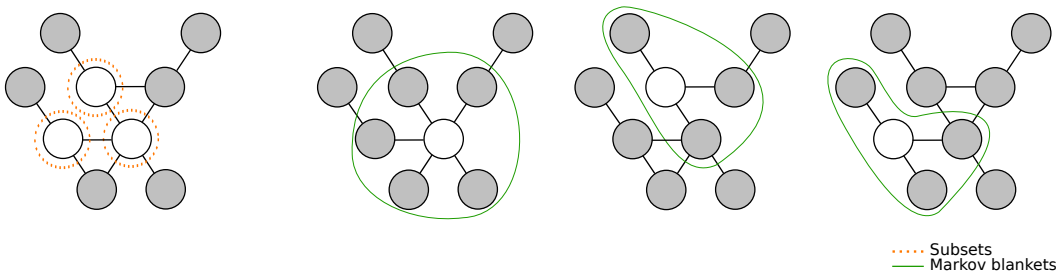


Figure 2.6: The Iterated Conditional Modes algorithm: rather than maximising directly the joint probability distribution of the three nodes, the states of the nodes are updated by optimising, sequentially, the probability distribution of each of the nodes conditional to the provisional estimates of the other nodes. The conditional probability distributions each node depends only on the state of the nodes that belong to its Markov blanket.

Another choice that occurs when the ICM algorithm is applied to more than two subsets is the definition of a scheme for updating the variables. The subsets may be updated in a systematic way, or by choosing nodes at random. Besag [14] initially applied ICM to image de-noising, optimising sequentially the conditional pdf of one pixel at the time, following a raster scanning pattern.

In this work, as explained in the subsequent chapters, where possible, the variables are divided in subsets such that the conditional probability distribution of each subset can be maximised with the Expectation Maximisation algorithm. Such approach simplifies the joint optimisation, eliminating the need for line search algorithms.

While one might consider that joint optimisation is generally a better idea than alternating optimisation, the experiments in chapter 5 suggest to split the parameters in subsets characterised by gradients of the same order of magnitude, in order to accelerate convergence of the optimisation algorithms.

It is worth to mention that, as pointed out by Green and Besag [14], ICM is a special case of Simulated Annealing, corresponding to instantaneous freezing. More computationally complex solutions that prevent local maxima can eventually be obtained considering the solutions proposed by Geman and Geman [15] and many others after them.

2.2 Brief introduction to emission computed tomography

Emission imaging systems (SPECT and PET) detect the photons emitted by a spatially distributed radioactive source. While the spatial density of the radio-tracer cannot be measured directly, the photons detected by the imaging hardware in the periphery allow one to infer the rate of emission within the imaging volume, proportional to the density of the radio-tracer. The problem of computing the rate of emission in the imaging volume may be considered an inverse problem, whereas the direct problem consists in predicting the photons observed at the periphery, given the rate of emission in the imaging volume. In the early years of SPECT and PET, the forward problem was formulated as a Radon inte-

gral transform, initially described by Radon in 1917, decades before the invention of emission imaging systems (the early developments of PET scanners started in the early 1950's [16]). While the inverse Radon transform is still perhaps the most widely adopted approach to tomographic reconstruction in emission and transmission imaging, the Radon transform is a very idealised model of the tomographic imaging processes, based on the assumption of noiseless measurements and of infinitesimal detectors. In the early 1980's the tomographic reconstruction problem was formulated in the probabilistic framework (Shepp and Vardi's article on the Maximum Likelihood Expectation Maximisation algorithm [17] is recognised as one of the earliest contributions). The probabilistic formulation introduces several advantages over the integral formulation (Radon transform), including the following: 1) the probabilistic formulation enables the construction of algorithms that account for arbitrarily complex models of the forward problem; 2) the probabilistic formulation allows us to account for the extent of the uncertainty associated to the measurements; 3) rather than computing a single solution of the inverse problem, one may compute the probability associated to multiple solutions; 4) incomplete angular sampling (see e.g. limited angle tomography [18, 19] and interior tomography [20, 21]) can be addressed straightforwardly in the probabilistic framework; 5) one may pose the problem of devising imaging systems with optimal characteristics; 6) the probabilistic framework lends itself to the integration of information from multiple sources, including multiple imaging modalities and population-derived information.

In order to address these aspects of the probabilistic formulation of emission tomographic imaging, it is useful to make use of the expressive power of probabilistic graphical models (PGMs). Furthermore, as we will see, the use of PGMs exposes the structure of the probabilistic models, aiding the construction of efficient inference algorithms. Paragraph 2.3 formulates the SPECT and PET imaging processes in the probabilistic graphical framework. The current paragraph serves as an introduction to paragraph 2.3, presenting a brief introduction of the principles of SPECT and PET imaging. The concepts introduced in this paragraph also serve as an introduction for the unifying computational model of scatter described in chapter 8 and for the projection and back-projection algorithms described in chapter 10. The sub-paragraph 2.2.1 introduces the light propagation processes involved in SPECT and PET imaging and their equations. The sub-paragraphs 2.2.2 and 2.2.3 describe the decay processes involved in SPECT and PET and summarise some of the techniques currently used for the detection of the gamma photons in SPECT and PET.

2.2.1 Interaction of photons with matter

The introduction to the photon transport equations reported in this paragraph is mainly derived from two sources: the MIT Nuclear Science and Engineering Reading Room lecture notes, openly available on the website of MIT [22], and the articles by J.H. Hubbell, openly available on the website of the National Institute of Standards and Technology (NIST) [23, 24].

Photons are electromagnetic radiation with zero mass, zero charge, and velocity c, the speed of light. Because they are electrically neutral, they do not steadily lose energy via Coulomb interactions

with atomic electrons as do charged particles. Instead they travel some considerable distance before undergoing a more "catastrophic" interaction. All of the photon interactions of interest in the range of energies involved in SPECT and PET (i.e. $511\ KeV$ for PET and $10\ KeV$ to $5\ MeV$ for SPECT, depending on the tracer) lead to partial or total transfer of the photon energy to electron energy. Thus the history of a photon in a material is characterised by the sudden disappearance of the photon or by scattering with loss of energy. Penetration of photons is governed statistically via interaction probabilities which depend on the energy of the photons and on the material. Since photons do not continuously lose energy via Coulomb interactions, the energy of the photon does not change unless an interaction occurs; therefore, as expressed by the equations in the following two paragraphs, unlike with charged particles, the probability of interaction per unit length of material traversed is not a function of the distance travelled. In the range of energies involved in SPECT and PET, the two main mechanisms of interaction are *photoelectric absorption* and *Compton scattering*.

Photoelectric absorption

In the photoelectric absorption process, a photon undergoes an interaction with an absorber atom in which the photon completely disappears. In its place, an energetic photoelectron is ejected from one of the bound shells of the atom. The interaction leaves an ionized absorber atom with a vacancy in one of its bound shells. This vacancy is quickly filled through the capture of a free electron from the medium and/or rearrangement of electrons from other shells of the atom. The probability that a photon interacts with an electron depends on the atomic configuration (i.e. on the atomic number Z in case of a pure element) and on the energy of the photon. No single analytic expression is valid for the probability of photoelectric absorption per unit length of material traversed over all ranges of energy E_γ and atomic number Z. Starting from the late 1800's, physicists and engineers have compiled increasingly accurate data-bases reporting the photoelectric characteristics of materials at different energies. While computers are promising accurate model-based computation of the photoelectric characteristics of materials (see [**?**] and [**?**]), current measurements are obtained by means of semi-empirical approaches [**?**] (one approach is described in section 'Note on the characterisation of the properties of a material' in the following). The photoelectric characteristics of a material are commonly reported in terms of the *linear attenuation coefficient*, symbolised in the following with μ_{PE}. In order to frame the concept of linear attenuation coefficient, let us consider the simple experimental setup in Fig. 2.7. The photons generated by a monochromatic source of radiation with energy E_γ are collimated into a narrow beam directed on a sample of material; a second collimator and a detector are placed on the other side of the sample. The purpose of the collimators, as clarified in the next paragraph, is to discard the scattered photons. Let the average number of photons entering the sample be $\bar{N}_0$. The probability that a photon interacts when traveling by a distance $\mathrm{d}x$, as discussed, depends solely on the energy E_γ and on the properties of the material, not on the distance already travelled. Let μ_{PE} (the subscript $_{\text{PE}}$ stands for photoelectric) be

the probability that a photon interacts by photoelectric interaction while traveling by $\mathrm{d}x$; if in average $\bar{N}_0$ photons enter the slab of material (e.g. per unit time), the average number of photons that have annihilated due to photoelectric absorption after traveling by $\mathrm{d}x$ is $\bar{N}_0\mu_{\mathrm{PE}}$. This is true for every location x, therefore, we can write the following differential expression:

$$\frac{\mathrm{d}\bar{N}(x)}{\mathrm{d}x} = -\bar{N}(x)\mu_{\mathrm{PE}}(x) \tag{2.18}$$

Integrating equation (2.18), the photon flux at location x is given by

$$\bar{N}(x) = \bar{N}_0\, e^{-\int_0^t \mu_{\mathrm{PE}}(x)\mathrm{d}x} \tag{2.19}$$

Note that μ_{PE}, the probability of interaction per unit length or *linear attenuation coefficient* has dimensions of an inverse length, i.e. $[\mathrm{cm}^{-1}]$ if length is expressed in cm:

$$\mu_{\mathrm{PE}}\left[\mathrm{cm}^{-1}\right] \tag{2.20}$$

If the material is homogeneous (μ_{PE} is constant), equation (2.19) simplifies to

$$\bar{N}(x) = \bar{N}_0\, e^{-\mu_{\mathrm{PE}}\, x}, \tag{2.21}$$

referred to as the Beer-Lambert exponential law of absorption. The linear attenuation coefficient μ_{PE} is proportional to the density of the absorber ρ $[\mathrm{g/cm}^3]$, which usually does not have a unique value but depends on the physical state of the material (e.g. consider varying pressure in the case of a gas). It is common practice, therefore, for the purpose of tabulation, to remove the dependence of μ_{PE} from the density, reporting μ_{PE}/ρ:

$$\frac{\mu_{\mathrm{PE}}}{\rho}\left[\frac{\mathrm{cm}^2}{g}\right] \tag{2.22}$$

The attenuation coefficient normalised by the density μ_{PE}/ρ is referred to as the *mass attenuation coefficient*. Using a source with known $\bar{N}_0$ in the experiment of Fig. 2.7 and measuring $\bar{N}(t)$ and the density ρ, one can think of obtaining a measurement of the mass attenuation coefficient of a material (a pure

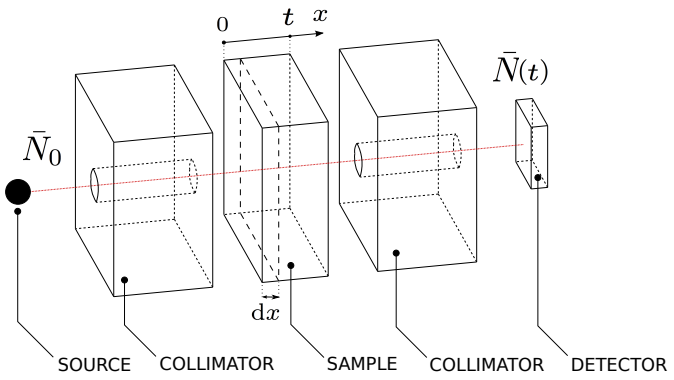


Figure 2.7: Experimental setup for the measurement of the linear attenuation coefficient.

element or a compound) by calculating μ_{PE} with equation (2.21) and dividing by the measured density of the material. However, the photoelectric effect is not the only interaction mechanism. In the range of energy typical of SPECT and PET, the second predominant mechanism of interaction is Compton scattering, described in the next section.

In the range of energies typical of SPECT and PET, the molecular interactions have negligible effect on the photoelectric interaction process and on Compton scattering. Therefore the linear attenuation coefficient for compounds can be obtained, with high accuracy, by the weighted sum of the linear attenuation coefficients of the constituting chemical elements. Tables reporting the mass attenuation coefficient for a large variety of chemical elements and for a wide range of energies have been compiled over the years and it is ongoing work to produce more accurate estimates (see [24, 25]).

Let us define the concept of *cross-section*, which will be useful to describe Compton scattering. As, in the range of energies of interest, we can express the characteristics of compounds as a weighted sum of the linear attenuation coefficients for each of the elements, let us consider the linear attenuation coefficient of a pure element at first. Considering the atomic weight $u\ M$, where $u = 1.6605402 \times 10^{-24}$[g] is the atomic mass unit (defined by convention as $\frac{1}{12}$-th of the mass of an atom of the nuclide ^{12}C - i.e. u is the inverse of Avogadro's number) and M is the relative atomic mass of the element, the mass attenuation coefficient can be written as

$$\frac{\mu_{\text{PE}}}{\rho}\left[\frac{\text{cm}^2}{g}\right] = \sigma_{PE}\left[\frac{\text{cm}^2}{\text{ATOM}}\right]\frac{1}{u\ M}\left[\frac{\text{ATOMS}}{\text{g}}\right] \tag{2.23}$$

Equation (2.23) defines the *cross-section* σ_{PE} for the photoelectric effect, with unit measure of area. The cross-section is the effective area that governs the probability of the interaction event. When particles in the beam are thrown against the sample, the cross-section is a hypothetical area measured around the atoms of the substance that represents a surface. One way to think of the cross-section is that if a photon of the beam crosses this surface, the interaction will occur, otherwise it will not.

In summary, the photo-electric cross-section σ_{PE} (function of the energy) characterises a pure element, independently of its density. The linear attenuation coefficient for compounds can be computed, for light in the energy range $10\ KeV$ to $100\ MeV$, as the sum of the linear attenuation coefficients of the constituent elements:

$$\mu_{\text{PE}} = \sum_i \mu_{\text{PE}_i} = \sum_i \sigma_{\text{PE}_i}\frac{\rho_i}{u\ M_i} \tag{2.24}$$

The propagation of photons by photo-electric effect through a non-uniform compound is described by the following differential equation:

$$\frac{\text{d}\bar{N}(x)}{\text{d}x} = -\bar{N}(x)\sum_i \sigma_{\text{PE}_i}(x)\frac{\rho_i(x)}{u\ M_i(x)} \tag{2.25}$$

Data-bases typically report the cross-section (or equivalently the mass attenuation coefficient) for the pure elements and the mass attenuation coefficient μ_{PE}/ρ for compounds, with μ_{PE} given by equation

(2.24) and ρ being the density of the compound in $[g\ cm^{-3}]$.

Compton scattering

The interaction of Compton scattering takes place between the incident gamma photon and an electron in the material. In Compton scattering, the incoming gamma photon is deflected through an angle θ with respect to its original direction. The photon transfers a portion of its energy E_γ to the electron, which is then known as a recoil electron. The energy transferred to the electron can vary from zero to a large fraction of the energy of the gamma photon. The energy of the scattered photon E'_γ and the scattering angle θ are related by a deterministic function, derived by Compton in 1923:

$$E'_\gamma = E_\gamma \frac{1}{1 + \frac{E_\gamma}{m_e c^2}(1 - \cos\theta)}, \tag{2.26}$$

where m_e is the mass of an electron ($\sim 511\ KeV/c^2$). The energy transferred to the recoil electron is equal to the difference $E_\gamma - E'_\gamma$. If the photon interacts with the electron in such a fashion that the angle of recoil is forward at $0°$, then the scattered photon will be scattered straight back; that is, $\theta = 180°$. This is the interaction that leaves the minimum energy with the scattered photon: $E'_\gamma(180) = E_\gamma \frac{1}{1+\frac{2E_\gamma}{m_e c^2}}$. The scattered photon has maximum energy when it propagates in the same direction as the incoming photon: $E'_\gamma(0) = E_\gamma$. When a photon interacts with an electron by Compton scattering, the angle of the scattered photon can be anywhere between 0 and $2\ \pi$. In 1928 Klein and Nishina derived, under the simplifying assumption that the electrons are unbound and at rest, the probability density associated with the scattering angle. The Klein-Nishina formula is expressed in terms of the *differential cross-section* (see e.g. [23]):

$$\frac{\mathrm{d}\sigma_{\mathrm{C}}}{\mathrm{d}\Omega} = \frac{1}{2} Z r_e^2 \left(\frac{1}{1 + \frac{E_\gamma}{m_e c^2}(1 - \cos\theta)} \right)^2 \left(1 + \cos^2\theta + \frac{\frac{E_\gamma}{m_e c^2}(1 - \cos\theta)^2}{1 + \frac{E_\gamma}{m_e c^2}(1 - \cos\theta)} \right) \tag{2.27}$$

where the subscript $_{\mathrm{C}}$ is employed to denote the Compton interaction, Ω is the unit solid angle, Z is the atomic number of the element and r_e is the classical electron radius ($\sim 2.2179\ fm$). The unit measure of the differential cross section is that of an area per solid angle:

$$\frac{\mathrm{d}\sigma_{\mathrm{C}}}{\mathrm{d}\Omega} \left[\frac{\mathrm{cm}^2}{\text{steradian ATOM}} \right] \tag{2.28}$$

The integral of equation (2.27) with respect to Ω (see [23]) gives the total cross-section $\sigma_{\mathrm{C}}^{\mathrm{TOT}}$ for the Compton scattering:

$$\sigma_{\mathrm{C}}^{\mathrm{TOT}} = 2\pi Z r_e^2 \left(\frac{1 + \frac{E_\gamma}{m_e c^2}}{\frac{E_\gamma^2}{m_e^2 c^4}} \left(\frac{2(1 + \frac{E_\gamma}{m_e c^2})}{1 + 2\frac{E_\gamma}{m_e c^2}} - \frac{\ln(1 + 2\frac{E_\gamma}{m_e c^2})}{\frac{E_\gamma}{m_e c^2}} \right) + \frac{\ln(1 + 2\frac{E_\gamma}{m_e c^2})}{2\frac{E_\gamma}{m_e c^2}} - \frac{1 + 3\frac{E_\gamma}{m_e c^2}}{(1 + 2\frac{E_\gamma}{m_e c^2})^2} \right) \tag{2.29}$$

The unit measure of the total cross section for the Compton scattering is the unit of area:

$$\sigma_{\mathrm{C}}^{\mathrm{TOT}} \left[\frac{\mathrm{cm}^2}{\mathrm{ATOM}} \right] \tag{2.30}$$

The total cross-section for the Compton scattering can be interpreted as the hypothetical surface around each atom such that if a photon of the beam crosses the surface, it scatters, otherwise it does not. The differential cross section expressed by the Klein-Nishina formula in equation (2.27) represents the infinitesimal fraction of such surface that corresponds to the photon being scattered in direction θ.

The Klein-Nishina formula, based on the approximating assumption that the electrons are unbound and at rest, describes very accurately Compton scattering in the range of energy $10\ KeV$ to $100\ MeV$, with less than 1% of error in the total cross-section [24]. Therefore, one can use equation (2.27) to compute the probability associated to direction θ for a photon of energy E_γ and equation (**??**) to compute the energy of the photon scattered in direction θ. Data-bases typically report the total cross-section (2.29) for different elements for a wide range of energies E_γ. As noted in the paragraph 'Simulating the propagation of high-energy radiation in a non-homogeneous medium', the total cross-section can be used to predict the propagation of light under certain simplifying assumptions, avoiding the numerical integration of the Klein-Nishina equation (2.27).

Note on the characterisation of the properties of a material

In the range $10\ KeV$ to $100\ MeV$ of interest to SPECT and PET, the photoelectric effect and Compton scattering are the predominant mechanisms of photon interaction. Equations (2.25) and (2.27) express the light transport equations for the photoelectric effect and for Compton scattering. The energy-dependent photoelectric cross-section σ_{PE} of a material (equation (2.25)) expresses the probability that a photon interacts with an electron by photoelectric effect. The photoelectric cross-section σ_{PE} depends on the atomic number and no single analytic expression is valid for σ_{PE} over all ranges of energy E_γ and Z. One may resort to measure the cross section for a given material, at a given energy, with an experiment analogous to the experiment represented in Fig. 2.7. However the measured signal $\bar{N}(t)$ depends not only on σ_{PE}, but also on the scattering properties of the material.

The Klein-Nishina formula of equation (2.27) predicts accurately the probability density associated to the scatter angle for Comption scattering, for every value of Z and for a wide range of energies ($10\ KeV$ to $100\ MeV$). While the cross-section for the photoelectric effect does not have a simple analytical expression, the Klein-Nishina formula of equation (2.27) may be employed to predict numerically the total cross-section and the differential cross-section of a chemical element with high accuracy in the range $10\ KeV$ to $100\ MeV$. The integral of the Klein-Nishina formula, reported in equation (2.29), expresses, when normalised by the density and by the atomic weight, the average number of photons that scatter when traversing a distance $\mathrm{d}x$ The number of photons that reach the source in the experiment of Fig. 2.7 is given by the number of incoming photons, minus the photons that interact by photoelectric effect, minus the total number of photons that scatter by Compton scattering, plus the photons that scatter with recoil angle 180^o (these can be neglected if the collimators have very thin holes or included in the calculation by integrating the Klein Nishina formula (2.27) over the solid angle subtended by the detec-

tor to the source). Neglecting the photons scattered in the forward direction (i.e. under the assumption of thin-hole collimator), σ_{PE} is obtained by subtracting the total Compton scattering cross-section $\sigma_{\mathrm{C}}^{\mathrm{TOT}}$ from the measured cross-section, obtaining, in case of a single element:

$$\frac{\mathrm{d}\bar{N}(x)}{\mathrm{d}x} = -\bar{N}(x)\left(\sigma_{\mathrm{PE}} + \sigma_{\mathrm{C}}^{\mathrm{TOT}}\right)\frac{\rho}{u\ M} \tag{2.31}$$

Integrating equation (2.31) in case of a uniform sample and solving for σ_{PE}, we obtain:

$$\sigma_{\mathrm{PE}} = \frac{u\ M}{\rho\ t}\ln\frac{\bar{N}_0}{\bar{N}(t)} - \sigma_{\mathrm{C}}^{\mathrm{TOT}} \tag{2.32}$$

Computing σ_{C} with the integral Klein-Nishina formula of equation (2.29) and measuring $\bar{N}_0$ and $\bar{N}(t)$, equation (2.32) may be employed to obtain a semi-empirical measurement of σ_{PE} with an experiment analogous to the experiment schematised in Fig. 2.7. This is one of the approaches employed to characterise the scattering and attenuation properties of materials. Note that the total cross-section parameter expresses the total probability of scattering per unit length of material traversed by a photon, however it does not express the probability per unit of angle. Therefore, as described in the next paragraph, the total cross-section can be employed to integrate the light propagation equations only for certain simple problems.

Simulating the propagation of high-energy radiation in a non-homogeneous medium

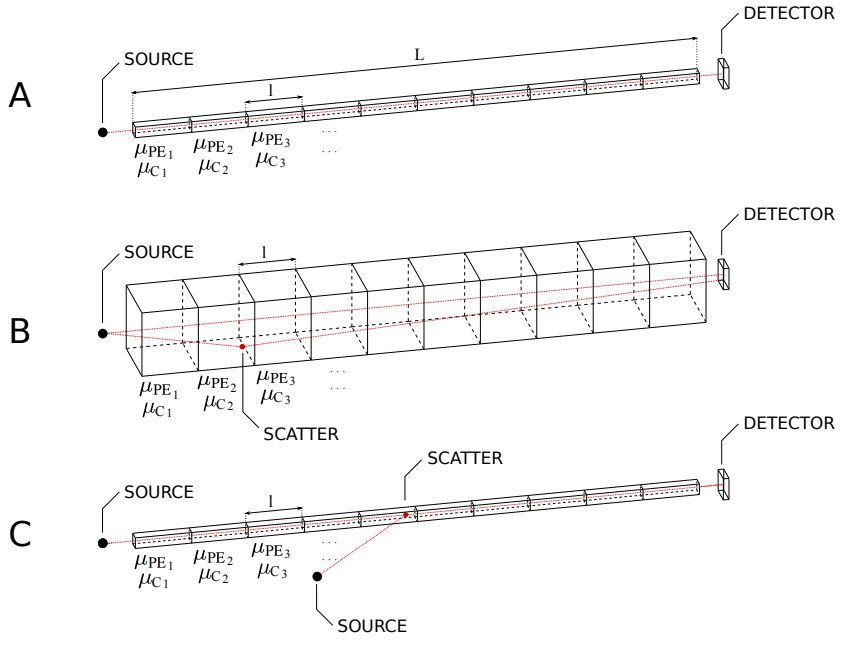


Figure 2.8: Numerical integration of the light propagation equations. A) Single source of radiation and thin propagating rod. B) Single source of radiation and thick propagating rod. C) Two sources of radiation and thin propagating rod.

Equations (2.25) and (2.27) describe the propagation of light for the range of energies $10\ KeV$ to $100\ MeV$, where the dominating interaction processes are the photoelectric effect and Compton scattering. The two differential equations (2.25) and (2.27) can be integrated numerically for the purpose of simulating the propagation of light in a homogeneous or inhomogeneous material. In emission computed tomography, the estimation of the rate of emission of photons in the imaging volume may be regarded as an inverse problem, with the direct problem consisting of calculating the photon flux in each of the detectors (i.e. simulating the propagation of light). Computing a solution of the inverse problem, as will be clarified in chapter 3, typically entails a sequence of simulations of the forward process. Chapter 10 describes an algorithm for the efficient simulation of light propagation in SPECT and PET. In order to gain some insight of the computational complexity associated to the simulation of the light propagation process in SPECT and PET and in order to introduce the methodology described in chapter 10, let us consider the three examples depicted in Fig. 2.8. In the experiment in Fig. 2.8-A, the light emitted by a source propagates through a thin rod of inhomogeneous material. While the photons propagate in the rod, they may interact by photoelectric effect or scatter by Compton scattering. If the rod is sufficiently thin, the majority of the scattered photons are not detected. Under the assumption of thin rod, therefore, the number of photons entering the surface of the detector is equal to the number of photons entering the rod minus the photons that interact by photoelectric interaction, minus the scattered photons. In the case of thin rod, therefore, $\bar{N}(L)$ can be calculated using the total Compton scattering cross section:

$$\bar{N}(L) = \bar{N}_0 e^{-\int_0^L \mu_{\mathrm{PE}}(x)\mu_{\mathrm{C}}(x)\mathrm{d}x}, \tag{2.33}$$

where L is the length of the rod, μ_{PE} is related to the photoelectric cross-section by the expression in equation (2.25), and μ_{C} is defined as $\mu_{\mathrm{C}} = \sigma_{\mathrm{C}}^{\mathrm{TOT}} \frac{\rho}{u\ M}$ in the case of a pure element, or by the weighted sum over μ_{C_i}, ρ_i and M_i is case of a compound:

$$\mu_{\mathrm{C}} = \sum_i \sigma_{\mathrm{C}_i}^{\mathrm{TOT}} \frac{\rho_i}{u\ M_i} \tag{2.34}$$

Assuming, for the purpose of numerical integration, that $\mu_{\mathrm{PE}}(x)$ and $\mu_{\mathrm{C}}(x)$ are piecewise constant in each of N regions of length $l = L/N$, as depicted in Fig. 2.8-A, the integral of equation (2.33) is expressed by

$$\bar{N}(L) = \bar{N}_0 e^{-\sum_{i=1}^{N}(\mu_{\mathrm{PE}_i}+\mu_{\mathrm{C}_i})l} \tag{2.35}$$

In Fig. 2.8-B, the thin rod is replaced by a thick rod. In this case a non-negligible fraction of the scattered photons enters the surface of the detector (the figure depicts one such photon). In this case the total Compton scattering cross-section is not sufficient to compute the number of photons entering the surface of the detector; instead one needs to integrate the Klein-Nishina formula considering the geometry of the system. Finally, consider the example of Fig. 2.8-C, identical to the case of Fig. 2.8-A, but with a second isotropic source of radiation. In this case some of the photons emitted by the second source scatter within the rod and are deviated in the direction of the detector. In order to estimate the number of

photons entering the surface of the detector one needs again to devise an integration scheme to integrate the Klein-Nishina formula. These examples pinpoint the limited use of the total Compton scattering cross-sections. Due to the characteristics of the SPECT and PET imaging hardware, as discussed in the next sections, the computation of the forward problems in SPECT and PET may partly rely on the thin rod approximation. As a final note, consider the case in which the detector, for the three geometries A,B and C measures the energy of each photon and discards the photons whose energy is lower then the energy of the source E_γ. If the detector could measure the energy exactly, then only the scattered photons would be discarded, therefore the number of accepted photons would be accurately predicted, in all three cases A,B and C, by the thin rod approximation.

2.2.2 SPECT

In SPECT, the unstable nuclei of the radio-tracer undergo *gamma-decay*, emitting a gamma photon (and an α or β particle). Each photon being emitted from a different atom, the emission events are independent from one another and the direction of the gamma photons is random with uniform distribution on the surface of the unit sphere. Therefore, macroscopically, each point in space is an isotropic source of photons, with emission rate proportional to the density (number of unstable atoms) of radio-tracer. The gamma photons are detected by a planar device, the Anger gamma-camera. Chapter 7 describes in detail how the Anger gamma-camera detects the high energy photons; it suffices here to say, referring to Figure 2.9, that the gamma camera records the coordinates of interaction and energy of each gamma photon that interacts with the scintillator crystal. Let us assume here that the scintillator crystal is 2-dimensional (the scintillator crystal typically has a thickness of a few mm to a few cm); therefore considering the gamma-camera a planar imaging device (i.e. the device records the coordinates of interaction of the photon on the plane). Chapter 7 describes in details the interaction process of the gamma photons with the scintillator crystal, accounting for the depth of interaction of the gamma photons within the scintillator crystal. The collimator, which may have a variety of configurations, the most common of which are pin-hole, multiple pin-holes, parallel-holes and slit-slat, has the function of restricting the sensitivity of each region of the camera imaging-plane to a small portion of the imaging volume. Fig. 2.9-left Fig. and 2.10 depict a parallel-hole collimator (typically made of lead or tungsten); with such design of the collimator, photons that travel in the direction perpendicular to the plane of the gamma camera have high probability to pass through one of the holes and be detected by the camera, while photons traveling in other directions are likely to annihilate in the collimator and therefore to be discarded (SPECT has very low efficiency, with the majority of the photons being discarded). With a parallel-hole collimator, therefore, given a location of interaction on the camera plane, only the photons emitted from locations in the imaging volume near the line normal to the camera plane and intersecting the given location on the camera plane have high probability to be detected at that location. This is clarified by Fig. 2.10. The trade-offs involved in the design of the collimators and other parameters of the acquisition systems are discussed in Chapter 9, which describes a criterium (based on information theoretic concepts) and an algorithm for the selection

of optimal design parameters.

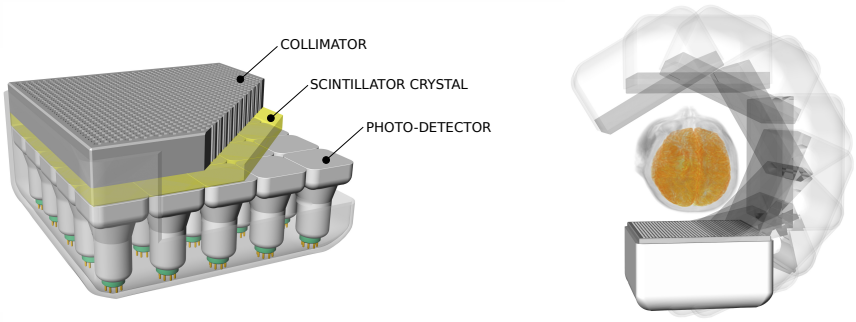


Figure 2.9: Left: gamma camera for Single Photon Emission Computed Tomography (SPECT) with parallel-hole collimator. The collimator absorbs photons that are not perpendicular to the imaging plane. The gamma photons interact with the scintillator crystal causing a scintillation. The scintillation is detected by the array of photo-detectors. Finally the electronics process the data collected by the photo-detectors, computing the location of interaction of the gamma photon and its energy. Right: one or multiple (e.g. three) gamma cameras rotate around the imaging volume during the SPECT acquisition. Typical acquisition times are in the range of a few minutes to one hour, with a total of 180 angular positions within a 180^o range.

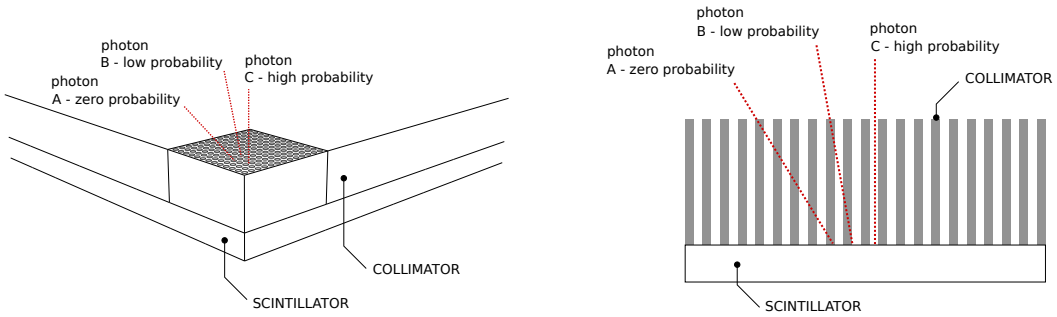


Figure 2.10: Schamatic of a parallel-holes collimator. The drawing displays only a small section of the collimator. Photons propagating in the direction perpendicular to the camera plane are likely to be detected (C); photons propagating in other directions are likely to annihilate within the collimator (A,B).

During a SPECT scan, the gamma camera moves around the imaging volume, typically in a circular trajectory with an angular span of 180^o or 360^o (see Fig. 2.9-right). The typical pattern of acquisition consists in rotating rapidly the camera by a small angle, e.g. 1^o , and then collecting photons until the camera moves to the next position. A list of the coordinates of interaction of the gamma photons with the scintillator crystal is recorded for each position of the camera. Let us consider a small volumetric region within the imaging volume and let us denote its location (e.g the coordinate of its center) by v. Considering the camera at one of its angular positions, let us consider a small area of the surface of the

scintillator crystal and let us denote by d its location on the plane (e.g. its center). The photons emitted within the small volume located in v propagate through the imaging volume and through the collimator according to the light transport equations (2.25) and (2.27). Due to the collimator, if the location v is near the line normal to the camera plane and intersecting d, then the number of photons that, after being emitted from v, are detected in d d is relatively high: it depends on the ratio between the integral area of the collimator surface and the integral area of the holes of the collimator (α) and on the absorbing and scattering properties of the propagating medium (the tissue of the subject). If v is far from the normal intersecting d, then the number of photons that, after being emitted from v, are detected in d is low, due to the thick layer of lead or tungsten that the photons find in their paths. Let us denote by a_{vd} the ratio between the number of photons emitted in v in average (e.g. per unit time), and the number of photons detected in d in average. If photons are emitted from v with rate λ_v, then the number of photons detected in average in d is $a_{vd}\lambda_v$.

If the collimator has ideal properties, i.e. it is very thick and has very small holes and very thin septa, then the probability that a photon emitted in v is detected in d is non-zero only if v is exactly on the line normal to the gamma camera plane intersecting d. Modern SPECT detectors measure the energy of the gamma photons. Assuming that the detector measures exactly the energy of the gamma photons and rejects the photons that have lost energy due to scattering, neglecting the photons that scatter with angle $\theta = 180^o$, the light propagation between v and d is analogous to the thin-rod approximation described in paragraph 'Simulating the propagation of high-energy radiation in a non-homogeneous medium'. If v in on the line normal to the camera plane and passing by d, the average number of photons detected in d is proportional to λ_v by the factor a_{vd} obtained by integrating the light propagation equations under the thin-rod approximation along the line that goes from v to (the 3-dimensional coordinates of) detector location d:

$$a_{vd} = \alpha e^{-\int_{L_{vd}} \mu_{\mathrm{PE}}(x)\mu_{\mathrm{P}}(x)\mathrm{d}x} \tag{2.36}$$

$$v \in L_d, \tag{2.37}$$

In equation (2.37), α is the sensitivity of the collimator; $\mu_{\mathrm{PE}}(x)$ is the linear attenuation coefficient for the photoelectric effect and $\mu_{\mathrm{C}}(x)$ is the linear attenuation coefficient for the Compton scattering, obtained from the total Compton scattering cross-section as expressed in equation (2.34). The line integral can be obtained numerically by sampling $\mu_{\mathrm{PE}}(x)$ and $\mu_{\mathrm{C}}(x)$ along the line and computing the finite sum as in equation (2.35). In practice, the collimator does not have ideal characteristics and the energy of the photons cannot be measured exactly. Chapter 10 describes the implementation of an algorithm for the computation of a_{vd} that accounts for a realistic model of the collimator by means of a depth-dependent convolution operator. Chapter 7 will describe in detail the gamma detection process, highlighting how the measurement of energy is affected by considerable uncertainty. Due to the uncertainty associated with the measurement of energy, when photons are selected by their energy, a certain number of photons

emitted from v may scatter one or multiple times and then be detected in d. In this case the thin-rod approximation fails and the accurate estimation of a_{vd} involves the integration of the Klein-Nishina formula in 3-dimensions. Algorithms for the efficient integration of the Klein-Nishina formula have been proposed, mostly based on Monte Carlo integration techniques [26, 27]. It suffices here to say that a_{vd} can eventually be computed with a high degree of accuracy, given sufficient computing resources. When performing probabilistic reasoning, one may discard some of the information. In the rest of this chapter and in chapter 3, the formulation will focus on the photons 'detected' in d, where a photon is considered detected if its energy is within a given energy threshold. However one may consider the photons detected at different levels of energy, or even make use of the continuous measurement of energy when estimating the rates of emission. Chapter 8 discusses these possibilities, proposing a so far unexplored unified computational model of scatter for computed tomography that generalises the model described in this chapter.

2.2.3 PET

In PET, the unstable nuclei of the radio-tracer undergo *positron decay*. A proton inside the nucleus of the radionuclide in converted into a neutron while releasing a positron and a neutrino. The positron collides with an electron. The result of the collision is the annihilation of the positron and of the electron, and the creation of two gamma photons, nearly in opposite directions, each with energy equal to the rest energy of the electron or positron ($511\ KeV$). Figure 2.11 reports a schematic representation of the decay process. The two gamma photons are detected by the PET imaging system, enabling the estimation of

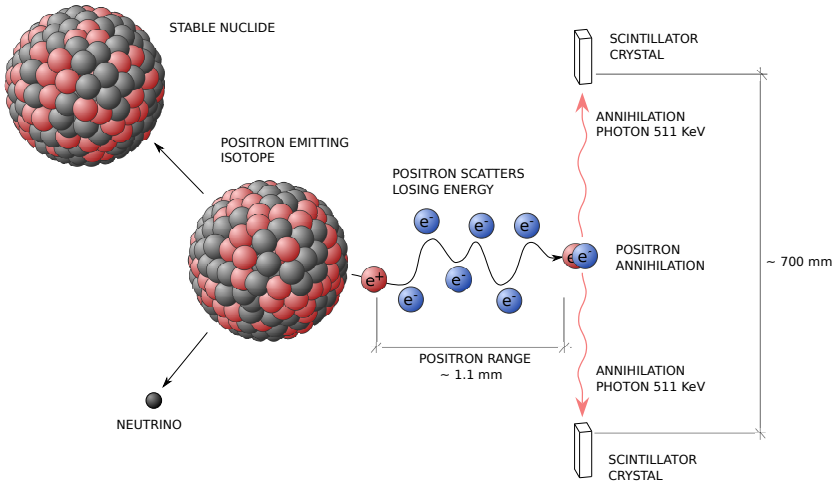


Figure 2.11: Positron decay. A proton inside the nucleus of the radionuclide in converted into a neutron while releasing a positron and a neutrino. The positron then collides with an electron. The result of the collision is the annihilation of the positron and of the electron, and the creation of two gamma photons, each with energy $511\ KeV$.

the spatially varying rate of decay per unit volume of positrons and consequently of the density of radio-

tracer, proportional to the number of unstable nuclei per unit volume and therefore to the rate of decay per unit volume. While in the case of single photon emission (SPECT) it is necessary to use a collimator in order to make the imaging process invertible, the PET imaging system does not require the use of a collimator (although early PET systems did use collimators). Figure 2.12 displays a schematic drawing of a modern PET scanner; the detectors are displaced on a cylinder around the imaging volume. In this example, the detectors are arranged, for engineering reasons, is blocks of 64 (Fig. 2.12-left). The system comprises 8 rings of detector modules, with 60 modules per ring, for a total of 30720 detectors. When a gamma photon interacts with one of the detectors, the electronics record the event (single interaction). When two detectors are activated within a short time interval (it takes about 3 $nsec$ for a photon to travel 1 m), the electronics record the coincidence event (i.e. both locations are recorded). As for the SPECT system, let us consider a small volumetric region within the imaging volume and let us denote its location (e.g the coordinate of its center) by v. Let us denote a pair of detector by d. Note that for the system in this example the total number of possible pairs is $30720 * (30720 - 1)/2 = 471843840$. While often the imaging systems record only a subset of the possible pairs, the tendency in modern PET systems is to record all possible events; therefore, in the example of Fig. 2.12, d is considered an integer between 1 and 471843840.

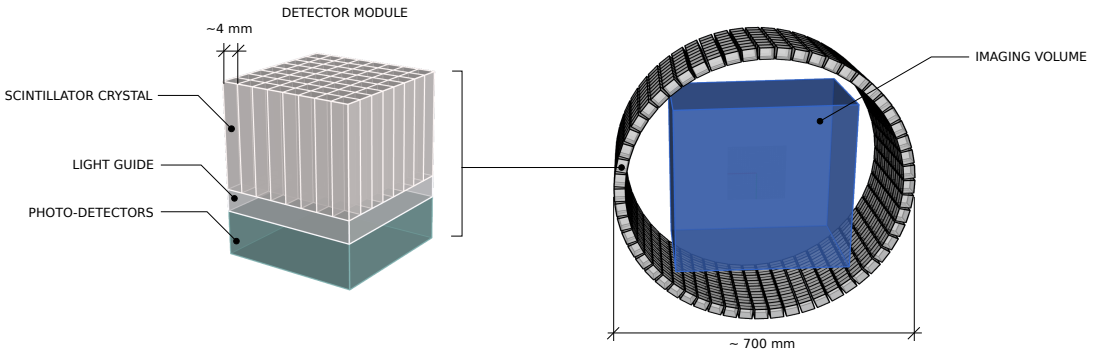


Figure 2.12: Left: drawing of a PET detector module, composed of 64 crystals. The interaction of a gamma photon with one of the crystals is recorded. Right: drawing of a PET imaging system. The system comprises 8 rings of detector modules, with 60 modules per ring, for a total of 30720 crystals. The electronics detect and record single events of interaction (i.e. the location of the activated crystal is recorded) and the coincident interaction of two photons (i.e. the position of the two crystals is recorded). The number possible detector pairs is $30720 * (30720 - 1)/2 = 471843840$.

The positron emitted from within the small volume located in v diffuses until it collides with an electron (the mean free path in human tissue is about 1.1 mm), then it annihilates emitting two gamma photons nearly in opposite directions (the angle is not exactly 180^o, but approximately 180^o). The two photons emitted within the small volume located in v propagate through the imaging volume. Each of the photons may leave the scanner undetected (through one of the two openings or passing undetected

through the detectors), or it may interact with one of the detectors (and therefore be detected). Each of the photons, before being detected, may scatter one or multiple times. Integrating the light transport equations equations (2.25) and (2.27), one can compute the number of photons detected by the detector pair d per number of photons emitted in v. Considering a given pair of detectors d, only the photons emitted from the region of the 3-dimensional space located near the line that connects the two detectors (referred to as line-of-response - LOR, and denoted here by L_d) may be detected but such pair. For this reason, the PET imaging system, is often said to be *self-collimated*. Let a_{vd} be the ratio between the number of photon pairs detected per unit time in d and the number of decay events produced in the unit time in v. PET detectors measure the energy of the gamma photons and discard the events for which one or both photons have scattered. If the measurements of energy were exact, the ratio a_{vd} between the number of photon pairs detected per unit time in d and the number of decay events in the unit time in v would be given, for locations v intersected by the line-of-response L_d, by integrating the Klein-Nishina equations under the thin-rod approximation and 0 everywhere else. Notice that the two photons undergo independent interaction processes, therefore the number of lost events is obtained integrating over the entire line-of-response L_d. If the location v is on the line of response L_d, the ratio between the number of photon pairs detected by detector pair d per unit time and the number of decays per unit time in v is given, under the thin-rod approximation, by the integral form:

$$a_{vd} = \alpha_d e^{-\int_{L_d} \mu_{\mathrm{PE}}(x)\mu_{\mathrm{P}}(x)\mathrm{d}x} \tag{2.38}$$

$$v \in L_d, \tag{2.39}$$

where α_d is the sensitivity of detector pair d. This can be integrated numerically similarly to equation (2.35). Notice that while in the case of SPECT the line is integrated from the location of d to the location v, in the case of PET the integral is identical for all locations v along the line-of-response L_d (not dependent on the position of v along the line-of-reponse). An algorithm for the efficient numerical integration of (2.39) is described in Chapter 10.

The number of photons detected in the unit time in d depends on the rate of emission in locations other that the line of response L_d. There is, in fact, a non-zero probability that two photons emitted by two different positron annihilations are considered a coincidence event. Such events are referred to as *randoms*. One of the two photons (or both) emitted in consequence of a positron annihilation may scatter, therefore changing direction. The electronics select events based on the energy of the detected photons, however, due to the uncertainty associated to the measurement of energy, the two photons may trigger a photon pair interaction event, despite the loss of energy due to scattering. Such events are referred to as *scatter events*. Randoms and scatter contribute to the ratio a_{vd}. The effect of randoms and scatter may be accounted for when computing a_{vd}, by integrating the Klein-Nishina formula, although the numerical integration of the Klein-Nishina formula requires a great deal of computing resources. Chapter 8 discusses this possibility and describes a generalisation of the model described in this chapter

which accounts for the continuous measurement of energy (i.e. the event is accepted regardless of the measured energy and the energy information for each interaction is recorded). It should be noted that the measurement of single interaction events may be used to estimate the contribution to a_{vd} due to randoms. Such technique and other techniques such as the *delayed window* technique are not discussed further in this thesis. As a final note, we are interested in the positron decay rate, therefore a_{vd} indicates the ratio between the number of pairs detected along line or response d in the unit time and the number of positron decays per unit time in v. Positrons undergo Coulomb interaction, losing energy along their way, until they annihilate with an electron, losing the remaining energy and determining the emission of the two gamma photons. The mean free path of the positrons is about $1.1\ mm$. In order to compute a_{vd} one should integrate the positron propagation equations, Typically, and in the algorithm described in Chapter 10, the positron propagation is approximated by means of a convolution.

2.3 Graphical models for emission computed tomography

In paragraphs 2.2.2 and 2.2.3, we have given a brief description of the physics of SPECT and PET. To summarise, we have denoted by a_{vd} the probability that a decay event in location v determines a detection event in detector (line-of-response in the case of PET) d. For a given location v and a given detector (line-of-response in the case of PET) a_{vd} can be computed, in first approximation, using the thin-rod approximation of the light propagation equations, described in paragraph 2.2.1. Chapter 10 describes the design and implementation of efficient algorithms for the approximate integration of the light propagation equations based on the thin-rod approximation. More accurate numerical estimates of a_{vd} can be obtained by integrating the Klein-Nishina equation, e.g. by means of Monte Carlo integration methods, although this is out of the scope of this thesis. While the emission processes and detection techniques differ in SPECT and PET, the formulation that follows applies equally to SPECT and PET. As described in paragraphs 2.2.2 and 2.2.3, the matrix $A = \{a_{vd}\}$ encompasses the characteristics of the imaging systems and the characteristics of the propagating medium. The remaining of this chapter, after introducing briefly the Poisson distribution, formulates the emission imaging problem in the probabilistic graphical framework.

2.3.1 The Poisson distribution

The Poisson distribution expresses the probability of a given number of events q occurring in a fixed interval of time (or space, but time is of interest here) if these events occur with a known average rate λ and independently of the time since the last event:

$$\mathcal{P}(q;\lambda) \triangleq \frac{e^{-\lambda}\lambda^{q}}{q!} \tag{2.40}$$

A simple derivation of the Poisson distribution as a limiting case to the binomial distribution can be found, for example, in [28]. The following three properties of the Poisson distribution will be used in the remainder of this chapter:

- Expectation: the expected value of a Poisson-distributed random variable is equal to λ (a proof in given in the appendix 3.A.3 of Chapter 3).
- Sum: the sum of two independent Poisson distributed variables is Poisson distributed with expectation given by the sum of the expectations (a proof in given in the appendix 3.A.4 of Chapter 3).
- Thinning: if the events generated by a Poisson process with expectation λ are accepted with probability α (rejected with probability $1 - \alpha$), then the number of accepted events is Poisson distributed with expectation $\alpha\lambda$.

2.3.2 Probabilistic model of emission imaging systems

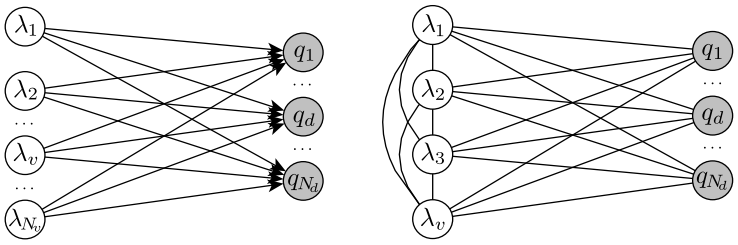


Figure 2.13: Left: voxel-based probabilistic graphical model (PGM) of PET and SPECT imaging systems. Activity is assumed to be a grid of point sources of radiation. The photon counts along the line of response d in PET and in detector bin d in SPECT q_d are *caused* by the activity in all voxels λ_v. Right: *moralisation* of the graph shows that $\lambda_v \not\perp \lambda_{v'}|q$ and $q_d \perp q_{d'}|\boldsymbol{\lambda}$. This is due to the causal structure of the model and implies the factorisation in (2.45).

The rate of decay $\tilde{\boldsymbol{\lambda}}$ is a continuous function of 3-dimensional space. However, in order to discretise the problem, let us assume that the events are emitted by point sources located on a regular grid in the 3-dimensional imaging volume. Let us denote by $\boldsymbol{\lambda} = \{\lambda_1, \lambda_2, \ldots, \lambda_v, \ldots, \lambda_{N_v}\}$ the rate of decay (i.e. number of events per unit time) at location v, with $v = \{1, 2, \ldots, v, \ldots, N_v\}$ indexing the discrete locations on the grid (e.g. with a raster scanning pattern). Let us denote by $\boldsymbol{q} = \{q_1, q_2, \ldots, q_d, \ldots, q_{N_d}\}$ the number of events detected at location d, with $d = \{1, 2, \ldots, d, \ldots, N_d\}$ indexing the detector locations in SPECT and the lines-of-response in PET. Letting a_{vd} be the probability that an event emitted in v is detected in d, the number of events per unit time in d is given by

$$\bar{q}_{vd} = \sum_{v=1}^{N_v} a_{vd}\lambda_v \tag{2.41}$$

Applying the sum and thinning properties of the Poisson distribution, the probability distribution associated to the number of events that one observes in d in the unit of time is

$$p(q_d|\boldsymbol{\lambda}) = \mathcal{P}\left(q_d; \bar{q}_{vd}\right) \tag{2.42}$$

Under the causal interpretation, we can construct a probabilistic graphical model of the imaging process. The probabilistic graphical model of figure 2.13 encodes the notion that $\boldsymbol{\lambda}$ is the unique cause of $\boldsymbol{q}$. The global Markov properties of the graph imply the following conditional independence:

$$q_i \perp q_j | \boldsymbol{\lambda} \tag{2.43}$$

This conditional independence is often considered a simplifying assumption. Such conditional independence, however, more precisely, derives from the assumption that $\boldsymbol{\lambda}$ is the unique cause of $\boldsymbol{q}$. If one removes $\boldsymbol{\lambda}$ from the graph in Fig. 2.13, it appears that q_i and q_j are independent. In fact, if one ignores that the photon count measurements in the two locations are caused by the hidden spatial distribution of activity, knowledge of q_i does not increase the state of knowledge about q_j. Now, introducing the variable $\boldsymbol{\lambda}$, hidden cause of $\boldsymbol{q}$, as expressed by the d-separation (or equivalently by the global Markov property), q_i and q_j are no longer independent when the value of $\boldsymbol{\lambda}$ is *not* specified. The knowledge expressed by the generative model, makes the variables interdependent: upon considering that there exists a hidden variable $\boldsymbol{\lambda}$, when given the value of q_i, one acquires information about q_j. It is often stated that q_i and q_j are not independent, but that they are assumed to be independent for simplification. However, as expressed by the graphical model, q_i and q_j, although not independent in absolute terms, are *conditionally* independent given $\boldsymbol{\lambda}$ under the assumption that they are common causes of the hidden variable $\boldsymbol{\lambda}$. The relation between causality and conditional independence is better understood, here, in the context of the Poisson distribution, with a toy example. Let us enclose in a sealed box a light bulb that emits $\lambda\ photons/sec$ and two photo-detectors that operate in photon-counting regime, both facing the light bulb and detecting each 1% of the light emitted. The battery and the knob that regulates the intensity λ are hidden inside the box. Two wires A and B carry the impulses generated from each photon interaction from the two photo-detectors. Experiment 1) Let us set the knob so that the light bulb emits precisely $100000\ photons/sec$. Let us now give the box to Alice, without telling her what is inside, ask her to count the events from wire A in a one second lapse with her pulse-counter and ask her how many events she guesses wire B would have generated in the same time lapse. She would not know what to say. Experiment 2) let us describe to Alice what is in the box and ask her the same question, without telling her the position of the knob. Now, upon counting 1054 events from wire A, she would guess the number of events in B to be near 1054. Experiment 3) Let us now tell Alice that the knob is set to $100000\ photons/sec$ and ask her again to count events in A for one second and to guess what she would read from B. Upon counting 856 events in A, she will guess that counts in B would be somewhere near 1000, regardless.

According to the conditional independence of equation (2.43), the probability distribution of $\boldsymbol{q}$ given $\boldsymbol{\lambda}$ is a product of N_d terms:

$$p(\boldsymbol{q}|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} p(q_d|\boldsymbol{\lambda}) \tag{2.44}$$

Expanding equations (2.44) and (2.42), we obtain:

$$p(\boldsymbol{q}|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} \frac{e^{-\sum_{v=1}^{N_v} a_{vd}\lambda_v} \left(\sum_{v=1}^{N_v} a_{vd}\lambda_v\right)^{q_d}}{q_d!} \tag{2.45}$$

The system matrix $A = \{a_{vd}\}$ encompasses the characteristics of the imaging system (location, size, sensitivity, spatial resolution of the Gamma detectors, eventual collimator; positron range, collinearity, sensitivity of the PET detectors, randoms and dead-time in PET) and the attenuation and scattering of γ radiation through the patient and the imaging hardware.

2.3.3 Derivatives of the log-likelihood

The next chapter describes first-order algorithms for the numerical computation of MAP estimates of the activity $\boldsymbol{\lambda}$. First-order optimisation algorithms, such as gradient descent, make use of the first order derivatives of the log-likelihood (2.44). Inserting (2.45) in (2.44) and applying the chain rule of differentiation:

$$\begin{aligned}
\frac{\partial}{\partial\lambda_r}\log p(\boldsymbol{q}|\boldsymbol{\lambda}) &= \frac{\partial}{\partial\lambda_r}\sum_{d=1}^{N_d}\log\frac{e^{-\sum_{v=1}^{N_v} a_{vd}\lambda_v}\left(\sum_{v=1}^{N_v} a_{vd}\lambda_v\right)^{q_d}}{q_d!} \\
&= \sum_{d=1}^{N_d}\left[\frac{\partial}{\partial\lambda_r}\left[-\sum_{v=1}^{N_v} a_{vd}\lambda_v + q_d\log\left(\sum_{v=1}^{N_v} a_{vd}\lambda_v - \log q_d!\right)\right]\right] \\
&= -\sum_{d=1}^{N_d} a_{rd} + \sum_{d=1}^{N_d} a_{rd}\frac{q_d}{\sum_{v=1}^{N_v} a_{vd}\lambda_v}
\end{aligned} \tag{2.46}$$

2.3.4 Concavity

A twice differentiable function is concave on an interval if and only if its second derivatives are non-positive there. This property gives a practical test for concavity. Differentiating twice the log-likelihood of the Poisson model:

$$\frac{\partial^2}{\partial\lambda_r\partial\lambda_s}\log p(\boldsymbol{\lambda}|\boldsymbol{q}) = -\sum_{d=1}^{N_d} q_d\frac{a_{rd}a_{sd}}{\left(\sum_{v=1}^{N_v} a_{vd}\lambda_v\right)^2} \tag{2.47}$$

The photon counts are greater of equal to zero $q_d \geq 0$; therefore with $a_{vd} \in [0,1]\forall v, d$, the log-likelihood is *non-strictly concave*. The first part of chapter 3 describes well known strategies for the maximisation of the log-likelihood and of the log-posterior. The second part of chapter 3 investigates the properties of maximum likelihood and maximum a posteriori estimators.

Chapter 3

Reconstruction, resolution and the partial volume effect

The first part of this chapter (paragraph 3.1) describes iterative reconstruction algorithms for maximum likelihood estimation (ML) and maximum a posteriori estimation (MAP) of the rate of emission. Three well known optimisation algorithms are derived in the context of probabilistic graphical models: maximum likelihood expectation maximisation (MLEM), ordered subsets expectation maximisation (OSEM) and one step late maximum a posteriori expectation maximisation (OSL-MAPEM). This first part of the chapter is intended to introduce the building blocks used for the development of the algorithms in the subsequent chapters.

The second part (paragraph 3.3) applies the algorithms derived in the first part of the chapter investigating, through numerical simulations, the properties of the ML and MAP estimators and tracing the link between resolution, ill-posedness of the reconstruction problem and *partial-volume effect*. This second part of the chapter adopts initially the frequentist perspective and then introduces the fully Bayesian perspective, expanded in chapter 9. The observations presented in this chapter motivate the modelling paradigm for multi-modal imaging described in the next chapter and the system design optimisation approach described in chapter 9.

3.1 Unconstrained optimisation

The log-likelihood of the Poisson model (2.45) is differentiable and (non-strictly) concave. With non-informative prior probability distribution of $\boldsymbol{\lambda}$ (i.e. all values of $\boldsymbol{\lambda}$ have the same probability, therefore the derivatives of the prior are zero in all directions for every value of $\boldsymbol{\lambda}$), in order to compute the MAP estimate (in this case referred to as maximum likelihood estimate), one may proceed with a projected gradient ascent algorithm with a suitable projection in order to impose the non-negativity constraint on the activity $\boldsymbol{\lambda}$:

$$\hat{\boldsymbol{\lambda}} = \underset{\boldsymbol{\lambda} \in \mathbb{R}^{N_v}_{\geq 0}}{\operatorname{argmax}} \log p(\boldsymbol{q}|\boldsymbol{\lambda}) \tag{3.1}$$

$$\lambda_r^{(n+1)} = I\left(\lambda_r^{(n)} + \Delta t \frac{\partial}{\partial \lambda_r} \log p(\boldsymbol{q}|\boldsymbol{\lambda})|_{\boldsymbol{\lambda}^{(n)}}\right) \tag{3.2}$$

$$I(x) = \begin{cases} x & if \quad \geq 0 \\ 0 & if \quad x < 0 \end{cases} \tag{3.3}$$

The main difficulty with gradient ascent, for such very high dimensional problem, is to decide the size of the step Δt. Algorithms with fixed Δt are unacceptably slow, while computing the optimum step size with line search algorithms at every iteration can be very expensive computationally. The high dimensionality of the problem prohibits the use of second order algorithms.

The difficulty of computing the optimum step size for the gradient descent algorithm is perhaps the reason of the popularity of the maximum likelihood expectation maximisation (MLEM) algorithm. Firstly, MLEM, when applied to the Poisson model, guarantees non-negativity of the solution; secondly, MLEM is free from nuisance parameters. By comparing the MLEM update formula with the gradient ascent update formula, MLEM appears to be a gradient-type optimisation algorithm with embedded line-search.

A derivation of MLEM is reported in the following, as it is rarely reported in the literature and it serves as a useful exercise for the development of the algorithms in the subsequent chapters, clarifying the role of the EM algorithm in the more complex models that will be introduced in chapter 4 and chapter 5.

3.1.1 Maximum likelihood expectation maximisation - MLEM

The MLEM algorithm is an interesting instance of the Expectation Maximisation (EM) algorithm described in paragraph 2.1.9. The Poisson model, introduced in chapter 2, and expressed synthetically be the graph in Fig. 3.1, expresses the probabilistic dependence between the rate of emission $\boldsymbol{\lambda}$ and the photon counts $\boldsymbol{q}$. Although this paragraph considers the case of uninformative prior (therefore it is not necessary to explicit $p(\boldsymbol{\lambda})$ in the equations), the formulation that follows maintains the explicit expression $p(\boldsymbol{\lambda})$ in anticipation of paragraph 3.2.

Though when one intends to maximise the likelihood of $\boldsymbol{\lambda}$, the model does not seem to involve hidden variables, these can be introduced somewhat artificially for the sole purpose of deriving an update formula with the Expectation Maximisation recipe.

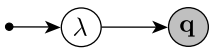


Figure 3.1: Probabilistic graphical model of the emission imaging system. Activity $\boldsymbol{\lambda}$ determines the photon counts $\boldsymbol{q}$.

Shepp and Vardi [17] proposed to introduce the hidden set $\mathbf{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_v, \ldots, \mathbf{z}_{N_v}\}$ with $\mathbf{z}_v = \{z_{2v}, z_{1v}, \ldots, z_{dv}, \ldots, z_{N_d v}\}$, where z_{vd} is a discrete variable indicating the number of photons

emitted from spatial location v and detected at detector bin d. One may say that the introduction of such set of hidden variables is artificial as, as easily verified, the probability distribution that one would associate to the photon counts $\boldsymbol{q}$ if the hidden variables were known is a delta function. In fact:

$$q_d = \sum_{v=1}^{N_v} z_{vd} \tag{3.4}$$

This implies also that the joint probability distribution of $\boldsymbol{\lambda}, \boldsymbol{q}, \mathbf{Z}$ factorises. In fact, given Z, one knows everything about $\boldsymbol{q}$. Such conditional independence is encoded in the probabilistic graphical model in Fig. 3.2, which can easily be obtained under the causal interpretation.

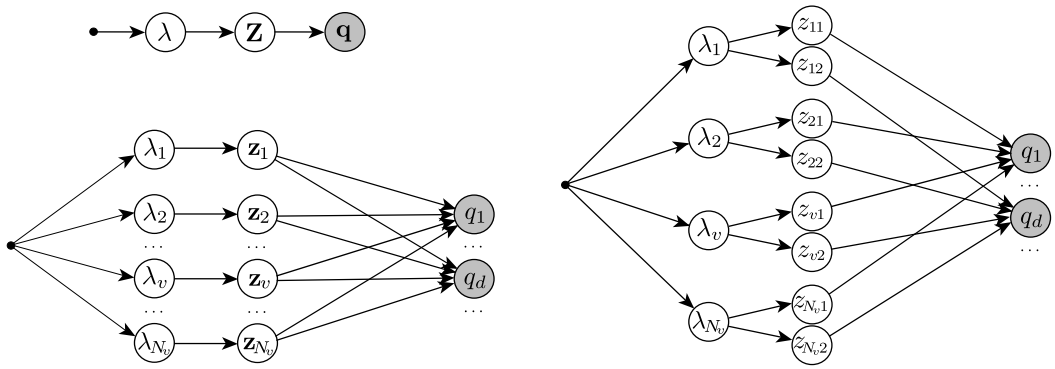


Figure 3.2: Graphical model explicitly showing the hidden variable $\mathbf{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_v, \ldots, \mathbf{z}_{N_v}\}$ with $\mathbf{z}_v = \{z_{1v}, z_{1v}, \ldots, z_{dv}, \ldots, z_{N_d v}\}$, introduced artificially for the purpose of applying the EM algorithm. The discrete variable z_{vd} indicates the number of photons emitted from spatial location v and detected at detector bin d.

From the thinning property of the Poisson distribution, the conditional probability $p(z_{vd}|\boldsymbol{\lambda})$ is a Poisson distribution with expectation $a_{vd}\lambda_v$ (see paragraph 2.3):

$$p(z_{vd}|\boldsymbol{\lambda}) = \mathcal{P}\left(z_{vd}; a_{vd}\lambda_v\right) \tag{3.5}$$

and the structure of the graph implies the conditional independence $z_{vd} \perp z_{v'd'}|\boldsymbol{\lambda}$:

$$p(\mathbf{Z}|\boldsymbol{\lambda}) = \prod_{v=1}^{N_v}\prod_{d=1}^{N_d} \mathcal{P}\left(z_{vd}; a_{vd}\lambda_v\right) \tag{3.6}$$

From (3.4), one may say that (3.6) implicitly defines the probability distribution over $\boldsymbol{q}$ and thus over $\mathbf{q}, \mathbf{Z}$:

$$p(\mathbf{q}, \mathbf{Z}|\boldsymbol{\lambda}) = p(\mathbf{Z}|\boldsymbol{\lambda}) \tag{3.7}$$

All the ingredients for the Expectation Maximisation recipe are ready. EM prescribes to compute the new estimate of the unknown variables ($\boldsymbol{\lambda}^{(n+1)}$) by maximising the expectation of the complete data

log-posterior $\log p(\boldsymbol{q}, \mathbf{Z}|\boldsymbol{\lambda}) + \log p(\boldsymbol{\lambda})$ with respect to the posterior probability of the hidden variables $\mathbf{Z}$ calculated with the current estimate of the unknown variables: $p(\mathbf{Z}|\boldsymbol{q}, \boldsymbol{\lambda}^{(n)})$. Since the hidden variables are discrete, the integral of (2.17) is replaced by the finite sum:

$$\begin{aligned} \boldsymbol{\lambda}^{(n+1)} &= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}}\, \mathbb{E}_{\mathbf{Z}}\left[\log p(\mathbf{q}, \mathbf{Z}|\boldsymbol{\lambda})\right] + \log p(\boldsymbol{\lambda}) \\ &= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{q}, \boldsymbol{\lambda}^{(n)}) \log p(\mathbf{q}, \mathbf{Z}|\boldsymbol{\lambda}) + \log p(\boldsymbol{\lambda}) \end{aligned} \tag{3.8}$$

Inserting (3.7) and expanding (3.6):

$$\begin{aligned} \mathbb{E}_{\mathbf{Z}}\left[\log p(\mathbf{q}, \mathbf{Z}|\boldsymbol{\lambda})\right] &= \mathbb{E}_{\mathbf{Z}}\left[\sum_{d=1}^{N_d}\sum_{v=1}^{N_v} \log \frac{e^{-a_{vd}\lambda_v}\left(a_{vd}\lambda_v\right)^{z_{vd}}}{z_{vd}!}\right] \\ &= \mathbb{E}_{\mathbf{Z}}\left[\sum_{d=1}^{N_d}\sum_{v=1}^{N_v} -a_{vd}\lambda_v + z_{vd}\log\left(a_{vd}\lambda_v\right) + const\right] \\ &= \sum_{d=1}^{N_d}\sum_{v=1}^{N_v} \mathbb{E}_{\mathbf{Z}}\left[z_{vd}\right]\log\left(a_{vd}\lambda_v\right) - a_{vd}\lambda_v + const \end{aligned} \tag{3.9}$$

The expectation of z_{vd} given $\mathbf{q}$ and $\boldsymbol{\lambda}$ is non-trivial to derive (a derivation is reported in the appendix 3.A), but has a simple expression:

$$\mathbb{E}_{\mathbf{Z}}\left[z_{vd}\right] = \sum_{\mathbf{Z}} p(\mathbf{Z}|\mathbf{q}, \boldsymbol{\lambda}^{(n)}) z_{vd} = a_{vd}\lambda_v^{(n)} \frac{q_d}{\sum_{v'=1}^{N_v} a_{v'd}\lambda_{v'}^{(n)}} \tag{3.10}$$

Inserting (3.10) in (3.9):

$$\begin{aligned} \mathbb{E}_{\mathbf{Z}}\left[\log p(\mathbf{q}, \mathbf{Z}|\boldsymbol{\lambda})\right] + \log p(\boldsymbol{\lambda}) &= \sum_{d=1}^{N_d}\sum_{v=1}^{N_v} a_{vd}\lambda_v^{(n)} \frac{q_d}{\sum_{v'=1}^{N_v} a_{v'd}\lambda_{v'}^{(n)}} \log\left(a_{vd}\lambda_v\right) + \\ &+ \sum_{d=1}^{N_d}\sum_{v=1}^{N_v} -a_{vd}\lambda_v + \log p(\boldsymbol{\lambda}) \end{aligned} \tag{3.11}$$

Setting the derivatives with respect to vector valued $\boldsymbol{\lambda}$ to 0:

$$\frac{\partial}{\partial \lambda_r}\left(\mathbb{E}_{\mathbf{Z}}\left[\log p(\mathbf{q}, \mathbf{Z}|\boldsymbol{\lambda})\right] + \log p(\boldsymbol{\lambda})\right) = 0 \tag{3.12}$$

$$\sum_{d=1}^{N_d} a_{vd}\lambda_r^{(n)} \frac{q_d}{\sum_{v'=1}^{N_v} a_{v'd}\lambda_{v'}^{(n)}} \frac{1}{\lambda_r} + \sum_{d=1}^{N_d} -a_{vd} + \frac{\partial}{\partial \lambda_r}\log p(\boldsymbol{\lambda}) = 0 \tag{3.13}$$

With uninformative prior, the derivatives of the log-prior $\frac{\partial}{\partial \lambda_r}\log p(\boldsymbol{\lambda})$ are zero and the maximiser has closed form solution:

$$\lambda_r^{(n+1)} = \lambda_r^{(n)} \frac{1}{\sum_{d=1}^{N_d} a_{rd}} \sum_{d=1}^{N_d} a_{rd} \frac{q_d}{\sum_{v=1}^{N_v} a_{vd}\lambda_v^{(n)}} \tag{3.14}$$

The expression (3.14), often named Maximum Likelihood Expectation Maximisation (MLEM), constitutes a fixed-point equation for $\boldsymbol{\lambda}$; starting with an initial guess $\boldsymbol{\lambda}^{[0]}$, one may iteratively update $\boldsymbol{\lambda}$ applying (3.14). The properties of Expectation Maximisation (see paragraph (2.1.9)) guarantee that each new application of (3.14) does not decrease the $\log$ likelihood. The MLEM update step entails a projection and a back-projection.

3.1.2 MLEM as Preconditioned Gradient Ascent

By comparing (3.14) with (2.46), the MLEM update formula appears to entail the same type of calculations involved in computing the gradient of the log likelihood. Indeed MLEM is a preconditioned gradient ascent optimisation algorithm. By substituting (2.46) in (3.14):

$$\lambda_r^{(n+1)} = \lambda_r^{(n)} + \frac{\lambda_r^{(n)}}{\sum_{d=1}^{N_d} a_{rd}} \frac{\partial \log p(\boldsymbol{q}|\boldsymbol{\lambda})}{\partial \lambda_r}\Big|_{\boldsymbol{\lambda}^{[n]}} \tag{3.15}$$

This expression, equivalent to (3.14), exposes how the MLEM update formula is a gradient ascent update step with position-dependent diagonal preconditioner.

3.1.3 Ordered subsets expectation maximisation - OSEM

As with any probabilistic model, one may perform the inference task disregarding part of the observations. One can for example imagine that half of the detectors of the emission imaging system are not functioning. In order to reduce the computational complexity of iterative reconstruction, the Ordered Subsets Expectation Maximisation algorithm [29] consists in updating $\boldsymbol{\lambda}$ using only a subset of the observations, changing the subset at each iteration. The OSEM algorithm described in [29] cycles, with a predefined order, through pre-defined subsets and uses the EM algorithm to update the activity at each step. However one may choose the subsets randomly and eventually use algorithms other than EM. The OSEM algorithm is very popular in clinical and research applications, reducing drastically the computation times. The number of detector bins utilised at one given iteration (i.e. the size of the subset) is a fraction of the total number of detector bins. With subsets of size $1/8$,$1/16$ and $1/32$ of the total number of detector bins, OSEM achieves roughly a speedup of 8, 16 and 32 times. The algorithm typically behaves poorly with small subsets, increasing the noisy look of the image and presenting random or cyclical variations of the likelihood. It is typical to adopt subsets of size $1/8$,$1/16$, $1/32$ or even $1/64$ is case of fully 3D PET where the number of detector bins is high. Fig. 3.3 depicts OSEM with 2 non-overlapping subsets.

$$\lambda_r^{(n+1)} = \lambda_r^{(n)} \frac{1}{\sum_{d\in\mathcal{D}^{(n)}} a_{vd}} \sum_{d\in\mathcal{D}^{(n)}} a_{vd} \frac{q_d}{\sum_{v=1}^{N_v} a_{vd}\lambda_v^{(n)}} \tag{3.16}$$

$$\mathcal{D}^{(n)} \subseteq \{1, 2, \ldots, N_d\} \tag{3.17}$$

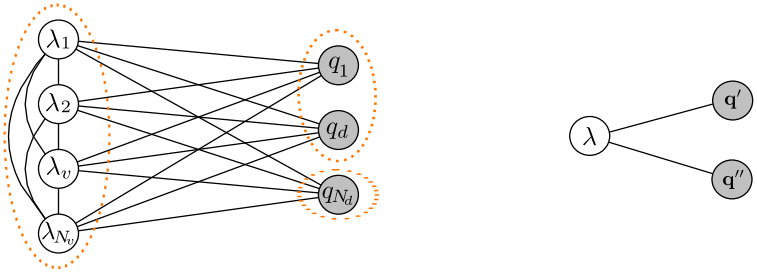


Figure 3.3: Ordered Subsets Expectation Maximisation (OSEM) algorithm. The observed variables are divided in subsets (2 subsets in figure). At each iteration $\boldsymbol{\lambda}$ is updated using the observations that belong to one of the subsets and disregarding all other subsets.

The NiftyRec software, described in the last chapter, implements the traditional OSEM algorithm and a version based on random subsets. This second solution is adopted in the experiments described in the subsequent chapters and preferred over the deterministic version as it appears to reduce artefacts. The random OSEM algorithm is also applied in chapter 5 to the four dimensional models, cycling through random subsets in space and time.

While OSEM is invaluable in order to reduce the computational complexity and works well for large subsets (for example $1/8$, $1/16$, $1/32$ of the total number of detector bins), convergence is not guaranteed even if the $\log$-likelihood is well behaved (i.e. the system matrix is invertible). OSEM with small subsets presents a cyclical behavior when subsets are cycled in a predefined order, as the algorithm produces estimates of the activity that are close to the most likely value according to the observations contained in each of the subsets in turn. For this reason the experiments described in the second part of this chapter, which investigate the properties of the maximum-likelihood and MAP estimators at convergence, make use of full MLEM.

3.1.4 Iterated conditional modes - ICM

The ICM algorithm described in 2.1.10 may be applied to the unconstrained tomographic reconstruction problem, as depicted in Fig. 3.4.

The Poisson log-likelihood, as derived in 2.3.4, is non-strictly concave. If the system matrix is invertible, the likelihood is concave, presenting a unique maximiser. In the case of concave cost function (i.e. in the case of invertible system matrix or in case the problem is regularised with a concave log-prior term as in paragraph 3.2.2), each of the subsets is concave, therefore ICM is guaranteed to converge to the solution for any choice of the order of subsets, as long as the optimisation algorithm adopted at each step of ICM is convergent (e.g. gradient descent or MLEM).

One application of ICM is to parallelise the reconstruction problem. This topic is discussed in

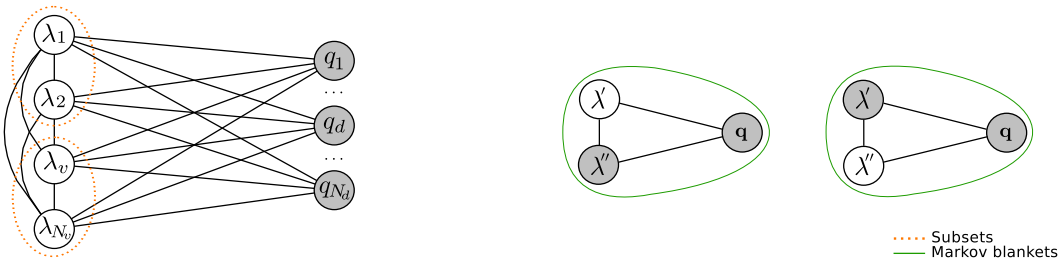


Figure 3.4: Iterated Conditional Modes algorithm (ICM) applied to unconstrained tomographic reconstruction. Subsets $\boldsymbol{\lambda}'$ and $\boldsymbol{\lambda}''$ are updated in sequence.

chapter 10. Projection and back-projection are inherently parallelisable essentially due to the conditional independence of the photon counts given the activity. Therefore with a distributed computing system, one may distribute the projection and back-projection on multiple computing devices. However, in a distributed computing system, at each iteration of an optimisation algorithm such as MLEM, the new estimate of the activity must be communicated to all the computing devices. This limitation, intrinsic of joint optimisation, may be relaxed by using the ICM algorithm, i.e. by updating subsets of the unknowns cyclically or randomly.

3.2 Maximum a posteriori estimation

Considering an arbitrary prior probability distribution $p(\boldsymbol{\lambda})$, the posterior probability distribution of $\boldsymbol{\lambda}$ given $\boldsymbol{q}$ is obtained by applying the law of total probability:

$$p(\boldsymbol{\lambda}, \boldsymbol{q}) = p(\boldsymbol{q}|\boldsymbol{\lambda})p(\boldsymbol{\lambda}) = p(\boldsymbol{\lambda}|\boldsymbol{q})p(\boldsymbol{q}) \tag{3.18}$$

Solving for $p(\boldsymbol{\lambda}|\boldsymbol{q})$ and maximising with respect to $\boldsymbol{\lambda}$, we obtain the following optimisation problem:

$$\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} p(\boldsymbol{q}|\boldsymbol{\lambda})p(\boldsymbol{\lambda}), \tag{3.19}$$

with $p(\boldsymbol{q}|\boldsymbol{\lambda})$ expressed by equation (2.44). As discussed in paragraph 3.1.1, with uninformative prior $p(\boldsymbol{\lambda})$, the M step of the EM algorithm has closed form solution (see (3.13)), producing the MLEM formula (3.14). However, for other choices of the prior, in general, there does not exist a closed form solution of (3.13). In order to compute the MAP estimate, then one may resort to adopting a gradient ascent optimisation algorithm, or eventually devise a generalised Expectation Maximisation algorithm: instead of computing the maximiser of (3.12) at each iteration, a new value that increases (3.12) is chosen at each iteration. An elegant and simple alternative was suggested by Peter Green in [30]: the one-step-late EM algorithm.

3.2.1 One-step-late maximum a posteriori expectation maximisation (OSL-MAPEM)

Since there does not exist a closed form solution for (3.13) for an arbitrary prior, Green [30] suggested to replace the gradient of the $\log$-prior in (3.13) with the gradient of the $\log$-prior evaluated for the previous estimate of $\boldsymbol{\lambda}$:

$$\frac{\partial}{\partial \lambda_r} \log p(\boldsymbol{\lambda}) \rightarrow \frac{\partial}{\partial \lambda_r} \log p(\boldsymbol{\lambda})|_{\boldsymbol{\lambda}=\boldsymbol{\lambda}^{(n)}} \tag{3.20}$$

The solution of (3.13) is then:

$$\lambda_r^{(n+1)} = \lambda_r^{(n)} \frac{1}{\sum_{d=1}^{N_d} a_{vd} - \frac{\partial}{\partial \lambda_r} \log p(\boldsymbol{\lambda})|_{\boldsymbol{\lambda}=\boldsymbol{\lambda}^{(n)}}} \sum_{d=1}^{N_d} a_{vd} \frac{q_d}{\sum_{v=1}^{N_v} a_{vd} \lambda_v^{(n)}} \tag{3.21}$$

One-step-late (OSL) is not guaranteed to converge and does not impose non-negativity. However the algorithm behaves well in most practical cases as long as the prior has a relatively small importance relative to the likelihood. While more sophisticated algorithms have been explored (e.g. [31]), the main advantage of the OSL algorithm is the absence of parameters.

3.2.2 The smoothness prior

Prior knowledge, or, from the frequentist perspective (further clarified in the next paragraph), the penalisation term, can be expressed in many forms. The selection of an appropriate form and parametrisation of the prior term is an open and challenging problem, discussed throughout this thesis. One of the critical factors for the selection of the form of the prior term is computational complexity.

One form of prior knowledge that leads to relatively simple inference algorithms is the assumption that the spatial concentration of the radio-pharmaceutical varies slowly in space. Considering the spatial concentration of the radio-pharmaceutical a continuous function of space $\tilde{\boldsymbol{\lambda}}(X)$, with X indicating the coordinates of 3-dimensional space, one may search for solutions of the type:

$$\hat{\tilde{\boldsymbol{\lambda}}} = \underset{\tilde{\boldsymbol{\lambda}}}{\operatorname{argmax}} \log p(\boldsymbol{q}|\tilde{\boldsymbol{\lambda}}) - \beta \|\nabla \tilde{\boldsymbol{\lambda}}(X)\|^2 \tag{3.22}$$

With $\nabla \tilde{\boldsymbol{\lambda}}(X)$ indicating the spatial gradient of $\tilde{\boldsymbol{\lambda}}$ and $\|\cdot\|^2$ indicating the L-2 norm:

$$\|\nabla \tilde{\boldsymbol{\lambda}}(X)\|^2 = \int [\nabla \tilde{\boldsymbol{\lambda}}(X)]^2 \mathrm{d}X \tag{3.23}$$

The term $-\beta \|\nabla \tilde{\boldsymbol{\lambda}}(X)\|^2$ promotes solutions that present, on average, small spatial derivatives. In order to discretise the optimisation problem, in 2.3.2, the activity $\tilde{\boldsymbol{\lambda}}$ has been considered in first place discrete (and denominated $\boldsymbol{\lambda}$). The same approach of considering the problem to be in first hand discrete can be adopted to discretize (3.22). The L2-norm of the gradient, in the discrete domain, is replaced by the sum of the finite differences between neighbouring voxels. Letting $c = \{c_1, c_2\}$ indicate a pair of neighbours

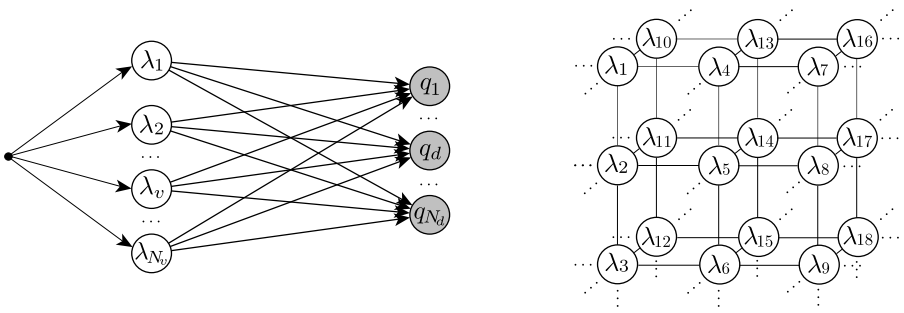


Figure 3.5: The graph on the left displays a generic prior probability distribution of the activity $\boldsymbol{\lambda}$. Prior assumptions about the statistics of the first order derivatives of the activity $\boldsymbol{\lambda}$ correspond, in the discrete domain, to a prior probability distribution that factorises according to the MRF reported in the graph on the right, denominated first order MRF. Equation (3.24) implies, as expressed by the derivative of its logarithm (3.25), that the activity at a given node (e.g. node 14) would be independent (a priori) from the activity in all other nodes if the activity in the neighbouring nodes (nodes 5, 11, 17, 23) were known. The graphical model encodes such assumption.

and $\mathcal{C}$ be the set of all pairs of neighbours, the probability distribution function whose $\log$ is the sum of the finite differences of neighbouring voxels multiplied by a constant β, is expressed by:

$$p(\boldsymbol{\lambda})_{\text{SMOOTHING}} = \prod_{c \in \mathcal{C}} \frac{1}{Z} e^{-\beta(\lambda_{c_1} - \lambda_{c_2})^2} \tag{3.24}$$

where Z is the normalisation factor that makes the integral of $p(\boldsymbol{\lambda})$ unitary, irrelevant in the maximisation with respect to $\boldsymbol{\lambda}$. Equation(3.24) is expressed graphically by the Markov Random Field with the structure reported in Fig. 3.5, often denominated first order MRF. Equation (3.24) can be interpreted globally or locally. Equation (3.24) expresses a probability distribution over all possible values of $\boldsymbol{\lambda}$, assigning high probability to values of $\boldsymbol{\lambda}$ that present, on average, small variations. The product of terms in (3.24), corresponds to the assumption that the difference between neighbouring voxels $d_c = \lambda_{c_1} - \lambda_{c_2}$ is a priori normally distributed and that $d_c \perp d_{c'}$. Note that though the dual local/global interpretation of MRFs just described may seem obvious, as pointed out by Besag [32], it is one of the fundamental findings that have emerged from the field of spatial statistics in the last few decades and it is expressed formally by the Hammersley - Clifford theorem [32].

Differentiating the log of (3.24) with respect to λ_r, only the terms that contain λ_r are non-zero. Therefore, indicating with η_r the set of indexes that correspond to voxels that are neighbours of voxel r:

$$\frac{\partial}{\partial \lambda_r} \log p(\boldsymbol{\lambda})_{\text{SMOOTHING}} = -2\beta \sum_{v \in \eta_r} (\lambda_r - \lambda_v) \tag{3.25}$$

This expression can be evaluated efficiently simply by convolution of $\boldsymbol{\lambda}$ with the Laplace filter, in the two-dimensional case, the $[3 \times 3]$ matrix:

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \tag{3.26}$$

and in the three-dimensional case a $[3 \times 3 \times 3]$ tensor with analogous structure. The MAP estimate can be computed with the OSL-MAPEM algorithm 3.21 with the gradient of the prior (3.25). Experiments are reported in paragraph 3.3.3.

3.3 Resolution, the partial-volume effect, information

Due to the occurrence of noise and to computational limitations, tomographic images constitute an approximate representation of the underlying quantity. The imaging system and the reconstruction algorithm determine the mapping between the underlying quantity and the numerical representation. Defining desired properties of the reconstructed images, of the reconstruction algorithms and of the imaging systems is essential in order to compare and parametrise the reconstruction algorithms, in order to evaluate, compare and design the imaging systems, and in order to inform image-driven decision processes.

The formulation of the inverse problem in the probabilistic framework enables us to relate the uncertainty associated to the acquisition of the emission data (i.e. the uncertainty that one would have about $\boldsymbol{q}$ if $\boldsymbol{\lambda}$ were known, expressed by the imaging model $p(\boldsymbol{q}|\boldsymbol{\lambda})$), to the uncertainty associated to the measurement of the activity $\boldsymbol{\lambda}$. Two main research paths have been delineated in occasion of the estimation of the uncertainty and optimisation of the design of the emission imaging systems. The first path, delineated by the work of J.A.Fessler [33, 34], J.W.Stayman [35, 36, 37], J.Nuyts [38, 39], K.Vunckx [40, 41, 42], l.Zhou [43], J.Qi [44], H.H Barrett [45, 46], D.W.Wilson [47], L.J.Meng [48, 49], N.Li [50], and others, defines the properties of the reconstruction algorithms and of the imaging systems in the frequentist probabilistic framework. Such an approach, summarised in paragraph 3.3.1, has exposed the fundamental limitations of computed tomography. However, the frequentist approach seems to fall short, as will be discussed in 3.3.1, when one attempts to define criteria for the choice of optimum imaging parameters. The second path, delineated by the work of H.H.Barrett [51, 52], P.Khurd and G.Gindi [53], H.C.Gifford, M.A.King, P.H.Pretorius and R.G.Wells [54], K.L.Gilland, B.M.W.Tsui, Y.Qi and G.T.Gullberg [55], A.Yendiki and J.A.Fessler [56] and others, frames the optimum parametrisation and system design problems in the decision theoretic framework. Criteria for the optimum selection of the parameters can be clearly formulated in the decision theoretic framework, on the basis of pre-defined imaging tasks. The imaging process is parametrised in order to maximise the performance of human observers or of numerical observers designed to perform such tasks.

This thesis focuses on a third path, unexplored in the context of tomographic reconstruction, the information theoretic perspective, immediate consequence of the fully Bayesian treatment of the imaging problem. The imaging model described in 2.3.2 expresses the probability distribution that one would

associate to $\boldsymbol{\lambda}$ for a given observation $\boldsymbol{q}$. Reconstruction is often posed, as is the case here in paragraphs 3.1 and 3.2, as the estimation of the value of $\boldsymbol{\lambda}$ that maximises such probability distribution. However, in the fully Bayesian approach, one shall consider the function as a whole, which expresses the state of knowledge that one possesses about $\boldsymbol{\lambda}$ upon observing $\boldsymbol{q}$. If one can characterise the full posterior, e.g. by drawing samples from it, and integrate it, then it is possible to compute answers to questions such as *"How much information have I acquired about the concentration of the radio-pharmaceutical in voxel v?"*: The observation of $\boldsymbol{q}$ determines the posterior probability distribution of the vector-valued variable $\boldsymbol{\lambda}$, therefore, upon observing $\boldsymbol{q}$, the probability distribution associated with the activity in voxel v is expressed by the marginal:

$$p(\lambda_v|\boldsymbol{q}) = \int\int\ldots\int p(\boldsymbol{\lambda}|\boldsymbol{q})\mathrm{d}\lambda_1\ldots\mathrm{d}\lambda_{v'\neq v}\mathrm{d}\lambda_{N_v} \tag{3.27}$$

Shannon's entropy, as explained in detail in Chapter 9, quantifies the information that one has about variable λ_v when the state of knowledge about λ_v is uncertain and therefore expressed by a probability distribution, such as, in this case, the probability distribution that occurs upon observing $\boldsymbol{q}$:

$$E_{\lambda_v} = \int p(\lambda_v|\boldsymbol{q})\log p(\lambda_v|\boldsymbol{q})\mathrm{d}\lambda_v \tag{3.28}$$

While this expression answers the question formulated above, evaluating numerically (3.28) and the similar expressions that arise from similar questions (some of which are addressed in Chapter 9), entails a tremendous computational complexity due to the integral over the high dimensional posterior probability distribution. Perhaps due to the prohibitive complexity that information criteria entail, the information theoretic formulation has not received interest from the computed tomography community. Chapter 9 defines the problem of optimum system design in the information theoretic framework and explores efficient algorithms for the numerical estimation of the information in emission computed tomography. The reminder of this chapter summarises the frequentist approach, contextualising the *partial volume effect* and discussing the use of the bias/variance curves, utilised in the remainder of this document for the evaluation of reconstruction algorithms.

3.3.1 Resolution

In photography, *resolution* indicates a quantitative measure of how close two parallel lines can be to each other and still be visibly *resolved*. In computed tomography, such a definition of resolution presents certain challenges.

In the literature of computed tomography, the resolution of the imaging system is often identified with the Full Width at Half Maximum (FWHM) of the *local impulse response* (LIR). In order to clarify the rationale behind such definition of resolution, let us start by considering the simple photographic imaging system of Fig. 3.6-left, composed of a converging lens and of a pixellated detector. The photons emitted (emission here includes reflection and refraction) from a point located on the imaging plane are ideally focused to a point on the focal plane. However, the lens is imperfect, determining *aberration*:

the photons are spread on the focal plane, as depicted in Fig. 3.6-left. Let A be the convolution operator expressed in matrix form. A discrete model of the imaging system is reported in Fig. 3.6-right, using the same notation as for the tomographic problem: $\boldsymbol{\lambda}$ for rate of emission (or intensity) and $\boldsymbol{q}$ for detector readout. While $\boldsymbol{\lambda}$, the true value of the light coming out of the imaging plane, is unknown, one has an *image* of $\boldsymbol{\lambda}$: $\boldsymbol{q}$. Notice that this problem differs from the tomographic problem only in the choice of the imaging matrix A. However there is a subtle difference between the two problems in that both $\boldsymbol{\lambda}$ and $\boldsymbol{q}$ live on a plane in the case of photography, while in the tomographic problem $\boldsymbol{\lambda}$ is a function of 3-D space and $\boldsymbol{q}$ is defined on a surface, such as the cylindrical bore of the PET scanner. The mapping between space and the vector valued functions does not appear explicitly in the mathematical formulation, but it is implicit in the structure of A.

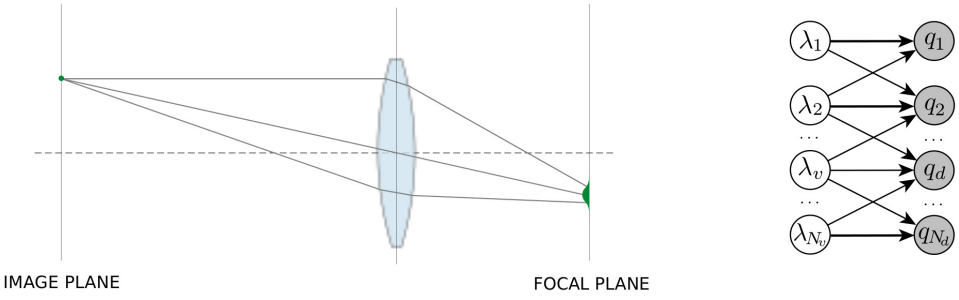


Figure 3.6: Left: Simple photographic imaging system. Right: discretised probabilistic model of the photographic imaging system. The intensity of light on the imaging $\boldsymbol{\lambda}$ plane determines the observation $\boldsymbol{q}$. Due to optical aberration, the image is blurred. Due to blurring, the arrows of the DAG extend from location v on the imaging plane, not only to the corresponding location on the detector plane, but also to nearby locations. Only the nearest locations are represented in the schematic graph.

In the frequentist view, one constructs a (deterministic) set of rules that produces the *estimate* of $\boldsymbol{\lambda}$ from the measurement $\boldsymbol{q}$. The set of rules, denominated *estimator* can be seen as a function that maps $\boldsymbol{q}$ to the estimate $\hat{\boldsymbol{\lambda}}$:

$$\hat{\boldsymbol{\lambda}}(\boldsymbol{q}) \tag{3.29}$$

The ML and MAP criteria define two of the many possible rules for constructing the estimates $\hat{\boldsymbol{\lambda}}$ from $\boldsymbol{q}$.

If the measurement is noiseless, the observation, indicated in this case with $\bar{\boldsymbol{q}}$, is equal to the transformation of $\boldsymbol{\lambda}$ by the imaging operator:

$$\bar{\boldsymbol{q}} = A\boldsymbol{\lambda} \tag{3.30}$$

When the measurement is corrupted by noise, the conditional probability distribution $p(\boldsymbol{q}|\boldsymbol{\lambda})$ expresses the noise characteristics. Common noise models, of interest in photography and in computed tomogra-

phy, are the *homoscedastic Gaussian model*:

$$p(\boldsymbol{q}|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} \mathcal{N}\left(q_d; [A\boldsymbol{\lambda}]_d, \sigma\right) \tag{3.31}$$

with noise level σ; and the *Poisson model*:

$$p(\boldsymbol{q}|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} \mathcal{P}\left(q_d; [A\boldsymbol{\lambda}]_d\right) \tag{3.32}$$

One can easily verify that in both cases, the expectation of $\boldsymbol{q}$ given $\boldsymbol{\lambda}$ is expressed by (3.30). For a given value of $\boldsymbol{\lambda}$, the measurement $\boldsymbol{q}$ may have many possible values, with probability density $p(\boldsymbol{q}|\boldsymbol{\lambda})$. The estimator produces deterministically a value of $\hat{\boldsymbol{\lambda}}$ given the measurement $\boldsymbol{q}$, therefore, given $\boldsymbol{\lambda}$, the estimator defines a probability distribution function of the estimates. Then the estimator, other than by the set of rules, can be characterised by the moments of the probability distribution associated with the estimates that it produces. The first two moments are:

$$\mu(\boldsymbol{\lambda}) = \mathbb{E}_{p(\boldsymbol{q}|\boldsymbol{\lambda})}[\hat{\boldsymbol{\lambda}}(\boldsymbol{q})] = \int \hat{\boldsymbol{\lambda}}(\boldsymbol{q}) p(\boldsymbol{q}|\boldsymbol{\lambda}) \mathrm{d}\boldsymbol{q} \tag{3.33}$$

$$\Gamma(\boldsymbol{\lambda}) = \mathbb{E}_{p(\boldsymbol{q}|\boldsymbol{\lambda})}[(\hat{\boldsymbol{\lambda}}(\boldsymbol{q}) - \mu(\boldsymbol{\lambda}))^2] = \int (\hat{\boldsymbol{\lambda}}(\boldsymbol{q}) - \mu(\boldsymbol{\lambda}))^2 p(\boldsymbol{q}|\boldsymbol{\lambda}) \mathrm{d}\boldsymbol{q} \tag{3.34}$$

The difference between the mean of the estimator and the true value of $\boldsymbol{\lambda}$ is denominated *bias* of the estimator:

$$B(\boldsymbol{\lambda}) = \mu(\boldsymbol{\lambda}) - \boldsymbol{\lambda} \tag{3.35}$$

Let us consider three choices of estimator for the noise models (3.31) and (3.32), while sticking to the parallel between photography and tomographic imaging.

The trivial estimator

In the case of photography, a possible choice of estimator is:

$$\hat{\boldsymbol{\lambda}}_{\text{TRIVIAL}}(\boldsymbol{q}) = \boldsymbol{q} \tag{3.36}$$

With this choice, one considers the true scene to be identical to the acquired image. Such choice of estimator is only allowed if the model has been discretised as in Fig. 3.6-right, with $N_v = N_q$ and clearly cannot be employed in the case of computed tomography. Substituting in (3.33):

$$\mu_{\text{TRIVIAL}}(\boldsymbol{\lambda}) = A\boldsymbol{\lambda} \tag{3.37}$$

for both noise models. This is of course a trivial result, but, in perspective, important to define resolution and to clarify the meaning of the LIR. Equation (3.37) says that if one considers the true value of $\boldsymbol{\lambda}$ to have the same value as $\boldsymbol{q}$, then, after performing many measurements, the representation that one constructs about $\boldsymbol{\lambda}$ using the estimator $\hat{\boldsymbol{\lambda}}_{\text{TRIVIAL}}(\boldsymbol{q})$ is $A\boldsymbol{\lambda}$, as smooth version of the true value of $\boldsymbol{\lambda}$.

The impulse response

If $\boldsymbol{\lambda}$ is an impulse, the mean of the estimator is

$$l_{v_{\text{TRIVIAL}}} \triangleq \mu(e_v) = Ae_v \tag{3.38}$$

where e_v is the v-th unit vector of length N_v. $l_{v_{\text{TRIVIAL}}}$ expresses the mean representation that one constructs when the true activity is an impulse centred in v and is therefore denominated *impulse response*. According to (3.38), $l_{v_{\text{TRIVIAL}}}$ is the v-th column of matrix A. The set of all impulse responses $\{l_v\}$, then, according to (3.37) can be thought of as the operator (matrix P, in the case of the trivial estimator) that applied to the true value of $\boldsymbol{\lambda}$ returns the mean of the estimator. Therefore $\{l_v\}$ characterises the mean of the estimator and the bias for any value of $\boldsymbol{\lambda}$. This is the case for all linear estimators (i.e. $\hat{\boldsymbol{\lambda}}(\boldsymbol{q})$ is a linear function of $\boldsymbol{q}$) such as $\hat{\boldsymbol{\lambda}}_{\text{TRIVIAL}}(\boldsymbol{q})$ and the ML estimator for the homoscedastic Gaussian model discussed in the next part of this paragraph, titled 'The ML estimator'. If the convolution operator is stationary, matrix A is Toeplitz and the convolution (3.37) can be written as:

$$\mu_{\text{TRIVIAL}}(\boldsymbol{\lambda}) = l * \boldsymbol{\lambda} \tag{3.39}$$

where l is the 2-D convolution kernel obtained by re-arranging the elements of A according to the indexing pattern assigned to the pixels, as described, e.g. in [57]. The 3-D Fourier transform of l is denominated *modulation transfer function* (MTF):

$$MTF(f) = \mathcal{F}[l] \tag{3.40}$$

Applying the convolution theorem, (3.39) is equivalent to:

$$\mathcal{F}[\mu_{\text{TRIVIAL}}(\boldsymbol{\lambda})] = MTF(f) \cdot \mathcal{F}[\boldsymbol{\lambda}] \tag{3.41}$$

If $\boldsymbol{\lambda}$ is, e.g., a sinusoidal function of (spatial) frequency $\bar{f}$, the representation that one constructs upon observing $\boldsymbol{\lambda}$ infinite times, is equal to $\boldsymbol{\lambda}$ attenuated by $|MTF(\bar{f})|$. If l is a 2-D Gaussian, then the MTF is a Gaussian centred in $f = 0$ and it is wider, the thinner is l. If l is wide (a common measure of width is the full width at half maximum - $FWHM$), then the MTF drops sharply, resulting in high spatial frequencies being attenuated.

In conclusion, in the case of the trivial estimator (3.36), the impulse response (defined as the mean estimator of the impulse) is an alternative representation of the imaging matrix A. The width of the impulse response (e.g. its $FWHM$) may be considered to be indicative of the ability that an observer would have to distinguish parallel lines when performing many measurements. However the trivial estimator cannot be defined in the case of tomography.

The ML estimator

The ML estimator applies equally to photographic and tomographic imaging.

$$\hat{\boldsymbol{\lambda}}(\boldsymbol{q}) = \underset{\boldsymbol{\lambda}}{\operatorname{argmax}}\, p(\boldsymbol{\lambda}|\boldsymbol{q}) \tag{3.42}$$

$$= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}}\, \log p(\boldsymbol{\lambda}|\boldsymbol{q}) \tag{3.43}$$

Let us consider the two noise models: homoscedastic Gaussian and Poisson.

Homoscedastic Gaussian model

$$p(\boldsymbol{q}|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(q_d - [A\boldsymbol{\lambda}]_d)^2}{2\sigma^2}} \tag{3.44}$$

If A is invertible, $\log p(\boldsymbol{q}|\boldsymbol{\lambda})$, as easily verified by differentiating twice as in 2.3.4, is concave. Its maximum therefore occurs for $\boldsymbol{\lambda}$ such that

$$\frac{\partial}{\partial \lambda_r} \log p(\boldsymbol{q}|\boldsymbol{\lambda}) = \sum_{d=1}^{N_d} -2a_{rd}q_d + \sum_{d=1}^{N_d}\sum_{v=1}^{N_v} 2a_{rd}^2\lambda_v = 0 \tag{3.45}$$

Solving for $\boldsymbol{\lambda}$ and rewriting in matrix form:

$$\hat{\boldsymbol{\lambda}}_{\text{ML GAUSSIAN}}(\boldsymbol{q}) = (A^T A)^{-1} A^T \boldsymbol{q} \tag{3.46}$$

The ML estimator for the homoscedastic Gaussian model is therefore a linear function of the observations. Substituting (3.46) in (3.33), the mean of the estimator (3.46) is:

$$\mu_{\text{ML GAUSSIAN}}(\boldsymbol{\lambda}) = (A^T A)^{-1} A^T A \boldsymbol{\lambda} \tag{3.47}$$

As expressed by (3.47) and (3.35), the ML estimator for the Gaussian noise model is *unbiased*, if $A^T A$ is invertible. The response to the impulse is the impulse:

$$l_{v_{\text{ML GAUSSIAN}}} = \mu(e_v) = e_v \tag{3.48}$$

Poisson model

$$p(\boldsymbol{q}|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} \frac{e^{-[A\boldsymbol{\lambda}]_d} [A\boldsymbol{\lambda}]_d^{q_d}}{q_d!} \tag{3.49}$$

$\log p(\boldsymbol{q}|\boldsymbol{\lambda})$ is again concave if A is invertible. The maximum occurs for $\boldsymbol{\lambda}$ such that:

$$\frac{\partial}{\partial \lambda_r} \log p(\boldsymbol{q}|\boldsymbol{\lambda}) = \sum_{d=1}^{N_d} -a_{rd} + \sum_{d=1}^{N_d} a_{rd} \frac{q_d}{\sum_{v=1}^{N_v} a_{vd}\lambda_v} = 0 \tag{3.50}$$

In matrix form:

$$-A^T \mathbf{1} + A^T \left(\frac{\boldsymbol{q}}{A\hat{\boldsymbol{\lambda}}} \right) = 0 \tag{3.51}$$

With the fraction indicating point-wise division and $\mathbf{1}$ indicating the all-ones vector. While for the Gaussian noise model $\hat{\boldsymbol{\lambda}}$ can be expressed explicitly, due to the point-wise division that appears in (3.51), in general an explicit expression of $\hat{\boldsymbol{\lambda}}$ cannot be derived for the Poisson model. Note that, as can be easily verified by manipulating (3.50), $\hat{\boldsymbol{\lambda}}$ can be expressed explicitly in case of diagonal system matrix with non-zero diagonal entries: $\hat{\lambda}_r = \frac{q_r}{a_{rr}}$ and in case of $N_v = 1$: $\lambda = \frac{\sum_d q_d}{\sum_d a_d}$. Such system matrices, however, are unrealistic in case of emission imaging. In general, the ML estimator for the Poisson model is therefore an *implicitly defined estimator*. The estimator is implicitly defined by (3.43) and (3.49), or,

equivalently, by (3.51). While the trivial estimator (3.36) and the ML estimator for the Gaussian noise model (3.46) are linear functions of the observations, the ML estimator for the Poisson model (3.51) is in general non-linear (given again exception in case of diagonal system matrix with non-zero diagonal entries and in case of $N_v = 1$).

With the estimator expressed implicitly, one cannot apply (3.33) and (3.34) to obtain explicit expressions of its mean and variance. What can be said about the ML estimator for the Poisson model is that it is *asymptotically unbiased*. Applying the central limit theorem, the Poisson probability distribution associated to q_d is asymptotically normal (i.e. for large values of $\boldsymbol{\lambda}$): $p(\boldsymbol{q}|\boldsymbol{\lambda}) \sim \mathcal{N}\left(q_d; \mu = [A\boldsymbol{\lambda}]_d, \sigma = \sqrt{[A\boldsymbol{\lambda}]_d}\right)$. One can derive, analogously to (3.47), the mean of the estimator for such *heteroscedastic Gaussian model*: $\mu_{\text{ML GAUSSIAN HETERO}}(\boldsymbol{\lambda}) = \boldsymbol{\lambda}$. Consequently:

$$\mu_{\text{ML POISSON}}(\boldsymbol{\lambda}) \to \boldsymbol{\lambda} \tag{3.52}$$

In absence of explicit expressions of the properties of the ML estimator for the Poisson model, Wilson and Tsui [58] and Fessler [34] have explored approximated expressions of $\mu_{\text{ML POISSON}}(\boldsymbol{\lambda})$ and $\Gamma_{\text{ML POISSON}}(\boldsymbol{\lambda})$, defining a local version of the IR, the *local impulse response* (LIR):

$$l_v(\boldsymbol{\lambda}) = \lim_{\delta \to 0} \frac{\mu(\boldsymbol{\lambda} - \delta e_v) - \mu(\boldsymbol{\lambda})}{\delta} = \frac{\partial}{\partial \lambda_v} \mu(\boldsymbol{\lambda}) \tag{3.53}$$

The LIR expresses how, on average, the reconstructed image is affected, at first order, by a perturbation of a particular element (i.e. voxel value) of the true underlying variable $\boldsymbol{\lambda}$. It is easily verified that the LIR is equivalent to the IR for linear estimators such as (3.36) to (3.46). Fessler has derived [34] the explicit analytical formulation of the LIR of the ML estimator applied to the Poisson model. Though the estimator is defined implicitly, the LIR, as defined in (3.53), can be expressed explicitly from the partial derivatives of the estimator, using the implicit function theorem. Such derivation is reported in [34] and yields:

$$l_{v_{\text{ML POISSON}}} \approx \left[A^T \text{diag} \frac{1}{A\boldsymbol{\lambda}} A\right]^{-1} \left[A^T \text{diag} \frac{1}{A\boldsymbol{\lambda}} A\right] e_v \tag{3.54}$$

If A is invertible, the LIR is an impulse.

The MAP estimator

The width of the IR (or of the LIR, if the IR cannot be expressed explicitly), is often considered, in the context of computed tomography, a measure of resolution, in analogy with the trivial estimator for the 2-D imaging problem discussed at the beginning of this paragraph. If the imaging matrix is invertible, the IR of the ML estimator for the Gaussian model (3.48) and the LIR of the ML estimator for the Poisson model (3.54) are delta functions, indicating infinite resolution. Indeed, the capacity to resolve lines is infinite in that, if one performs the measurement again and again, as expressed by the integration in (3.33), the representation that one constructs eventually becomes exact.

In practice, however, the imaging matrix A, as discussed in the next paragraph, is bound to be ill-conditioned. When the imaging matrix is not invertible, the IR and the LIR of the ML estimator are no

longer defined. One approach to make the problem well behaved (the problem is here considered not well behaved when the system matrix is not invertible) is to add a penalisation term to the log likelihood function.

$$\hat{\boldsymbol{\lambda}}_{\text{MAP}}(\boldsymbol{q}) = \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \log p(\boldsymbol{q}|\boldsymbol{\lambda}) + \log p(\boldsymbol{\lambda}) \tag{3.55}$$

Homoscedastic Gaussian model

There do not exist, in general, explicit expressions of $\hat{\boldsymbol{\lambda}}_{\text{MAP}}(\boldsymbol{q})$, $\mu_{\text{MAP}}(\boldsymbol{\lambda})$, $l_{v_{\text{MAP}}}(\boldsymbol{\lambda})$ and $\Gamma_{\text{MAP}}(\boldsymbol{\lambda})$ for the MAP estimator. However, there exist explicit expressions in case of homoscedastic Gaussian likelihood coupled with the quadratic MRF prior 3.2.2. The derivation of the IR is reported e.g. by Fessler [34]:

$$l_{v_{\text{MAP GAUSSIAN SMOOTHING}}} = \left(A^T A + R\right)^{-1} A^T A e_v \tag{3.56}$$

where R is the matrix of the second mixed partial derivatives of $\log p(\boldsymbol{\lambda})$:

$$R = r_{ij} = \frac{\partial^2}{\partial \lambda_i \partial \lambda_j} \log p(\boldsymbol{\lambda}) \tag{3.57}$$

Poisson model

The LIR of the MAP estimator with the smoothing prior has been derived by Fessler in [33]:

$$l_{v_{\text{MAP POISSON SMOOTHING}}} \approx \left[A^T \operatorname{diag}\frac{1}{A\boldsymbol{\lambda}} A + R\right]^{-1} \left[A^T \operatorname{diag}\frac{1}{A\boldsymbol{\lambda}} A\right] e_v \tag{3.58}$$

3.3.2 Bias / Variance

As discussed in the previous paragraph, the impulse response (IR) constitutes an alternative representation of the mean and bias of linear estimators. The mean is given by the true underlying activity convolved with the IR and the bias is defined as the mean minus the true activity. For estimators that do not accept an explicit representation, such as the ML estimator for the Poisson model, the local impulse response (LIR) expresses a first order approximation of the mean and bias. The IR of the ML estimator for the homoscedastic Gaussian model and the LIR of the ML estimator for the Poisson model are delta functions if the imaging matrix A is invertible. Such results signify that if one observes again and again the photon counts $\boldsymbol{q}$, with an invertible imaging matrix, the true value of $\boldsymbol{\lambda}$ eventually becomes known. However $\boldsymbol{q}$ is typically measured once and the system matrix is, in general, ill-conditioned. Though the IR and the LIR are related to the ability that an observer would have to *resolve* parallel lines from a single observation of $\boldsymbol{q}$, it is not straightforward to establish their relation to resolution.

Let us start by considering the case is which the system matrix is invertible. One does not seem to require regularisation and the impulse response is a delta function: if one could measure $\boldsymbol{q}$ infinite times, $\boldsymbol{\lambda}$ would become known exactly. This is expressed by the first order moment of the estimator being equal to the true value of Lambda. In order to characterize the single measurement, one has to consider the higher moments. The covariance matrix, defined in (3.34), expresses, in the $L2$-norm, the average distance of the estimates from the mean. If the covariance matrix were diagonal, the estimates of the elements of vector-valued $\boldsymbol{\lambda}$ would be independent from one another across multiple observations.

Small values of the diagonal elements of the covariance matrix imply that the estimates obtained from repeated measurements are near the mean and therefore near one another. Bias alone, expressed by the IR and the LIR, in other words, does not characterise the single measurement. Even in the case of invertible imaging matrix, the estimates of the elements of vector-valued $\boldsymbol{\lambda}$ are co-variated, making it difficult to express the ability to resolve parallel lines. While this depends on the properties of the estimator (the covariance, and eventually higher order momenta), relating the covariance matrix and the IR to the ability to resolve lines or eventually other features of interest is a challenging problem.

The problem is further complicated with ill-conditioned imaging matrix. The IR and LIR are not defined. The width of the IR and LIR of the MAP estimator with quadratic MRF penalty, expressed by (3.56) for the Gaussian noise model and by (3.58) for the Poisson noise model, have been considered, in recent literature, measures of resolution. These, however, express, again, the mean of the estimator, the representation that one would construct upon performing infinite measurements. The problem of selecting the weight of the smoothing term, problem that is largely unsolved in its own terms, complicates further the definition of a measure of resolution.

Finally, the problem of defining resolution becomes even more volatile considering arbitrary penalty functions, such as non-differentiable functions (e.g. sparsity constraints on the image gradient) or penalty functions that do not relate to the mapping between vector $\boldsymbol{\lambda}$ and 3-dimensional space.

While it is difficult to define a measure of resolution and to relate it to bias (IR, LIR) and covariance, a good estimator is characterised by low bias and diagonal covariance matrix with small values. Two useful tools for the visualisation and comparison of the properties of estimators and of the imaging systems are the grey scale visualisation of one or multiple columns of the covariance matrix and the plot of the norm of the bias versus the norm of the diagonal of the covariance matrix:

$$B = \|\mu(\boldsymbol{\lambda}) - \boldsymbol{\lambda}\|^2 \tag{3.59}$$

$$V = \|\text{diag}[\Gamma(\boldsymbol{\lambda})]\|^2 \tag{3.60}$$

The grey scale image of a column of the covariance matrix is obtained by reshaping the column into a 3-D image (2-D in case of 2-D imaging) according to the voxel indexing pattern. One of more slices of the resulting imaging volume is then displayed for the purpose of visualising how concentrated the values are near the voxel whose index corresponds to the selected column. Paragraph 3.3.3 makes use of such visualisation of the covariance matrix and of the bias/variance norm plots to characterise the properties of the ML estimator and of the MAP estimator with the smoothing prior.

3.3.3 The partial volume effect

PET-CT and, recently, PET-MR, have improved the visual interpretation of emission imaging data. When overlaid on the functional images, anatomical images support their qualitative interpretation, the delineation of anatomical structures and the quantitative assessment of the uptake in regions of interest. Overlaying the functional and the anatomical images, however, exposes the fundamental challenge of

quantification. An observer can distinguish underlying structures in common between the two images, but the emission image appears to be blurred, giving the observer the impression of spill out of the pharmaceutical. Such problem is often referred to as *spill out effect* or *partial volume effect* (PVE). The expression *partial volume effect* suggests that the problem is related to the discretisation of the imaging volume in voxels of finite size, which cover partially multiple areas with different levels of activity, as depicted in Fig. 3.7. While this is partially true, one may raise the objection that the choice of the size of the voxels is arbitrary.

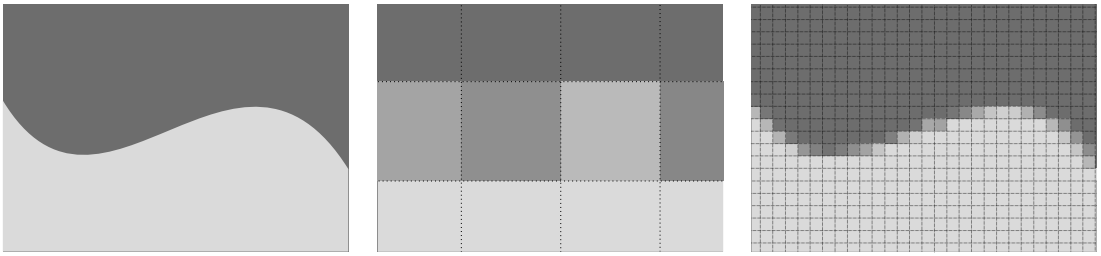


Figure 3.7: The expression *partial volume effect* suggests that the blurring effect in computed tomography occurs in consequence of the discretisation of the imaging volume in voxels of finite size, which cover partially multiple areas with different levels of activity. This is partially true, however the choice of the size of the voxels is arbitrary.

The tomographic imaging problem is inherently ill-posed, due to $\boldsymbol{\lambda}$ being a continuous function of space. In 2.3.2, we have formulated the imaging model in the discrete domain, considering the activity non-zero only in discrete locations. Let us consider the continuous formulation of the problem, with $\tilde{\boldsymbol{\lambda}}$ being an infinite dimensional vector. The ML estimation problem $\hat{\tilde{\boldsymbol{\lambda}}} = \operatorname{argmax}_{\tilde{\boldsymbol{\lambda}}} p(\boldsymbol{q}|\tilde{\boldsymbol{\lambda}})$ is clearly ill-posed, with infinite unknowns and finite (discrete) observations. The discrete model described in 2.3.2 can be considered to arise, in the continuous domain formulation, from a particular choice of prior $p(\tilde{\boldsymbol{\lambda}})$:

- A) values of $\tilde{\boldsymbol{\lambda}}$ such that $\tilde{\boldsymbol{\lambda}}$ is non-zero on a regularly spaced grid of points and zero everywhere else have non-zeros probability; every other configuration of $\tilde{\boldsymbol{\lambda}}$ has probability zero.

Another convenient choice of $p(\tilde{\boldsymbol{\lambda}})$ is the following:

- B) values of $\tilde{\boldsymbol{\lambda}}$ such that $\tilde{\boldsymbol{\lambda}}$ is constant within each of the rectangular regions of space indexed by v (voxels), have non-zero probability; every other configuration of $\tilde{\boldsymbol{\lambda}}$ has probability zero.

These two assumptions, while introduced solely for convenience of the numerical implementation, constrain the estimation problem. If one considers, e.g., the case of $N_v = 1$ voxel, the estimation problem is certainly well behaved with a unique solution (given in closed form, as one can easily verify, by the normalised integral sum of the photon counts $\sum_{d=1}^{N_d} q_d / \sum_{d=1}^{N_d} a_d$). While the constrained problem may be well behaved (invertible), such two assumptions are, in the Bayesian view, unrealistic, especially for

large voxels. Decreasing the size of the voxels, on the other hand, the problem becomes more and more ill-posed, eventually approaching the continuous formulation.

A second type of assumption, which, this time, relates to genuine prior belief about $\tilde{\boldsymbol{\lambda}}$, is that $\tilde{\boldsymbol{\lambda}}$ changes slowly in space. In order to introduce such assumption, as discussed in 3.2.2, one can follow two approaches. The first approach is to encode the assumption in the continuous domain: $\log p_{\text{MRF}}(\tilde{\boldsymbol{\lambda}}) \propto -\beta\|\nabla\tilde{\boldsymbol{\lambda}}\|^2$ and devise a discretisation scheme. The second approach is to introduce the assumption readily in the discrete domain. In the second case, which we have adopted in paragraph 3.2.2, the assumption of smoothness is encoded by the first order MRF with quadratic energy functionals, expressing the prior belief that the difference between the activity $\tilde{\lambda}(X_{c_1}) - \tilde{\lambda}(X_{c_2})$ in two locations X_{c_1} and X_{c_2} is normally distributed. The formulation of the assumption of smoothness in the discrete domain is equivalent to the prior $p(\tilde{\lambda})$ given by the product of two terms, a term of type A) and the MRF term:

$$p(\tilde{\boldsymbol{\lambda}}) = p_{\text{DISCRETISATION}}(\tilde{\boldsymbol{\lambda}}) p_{\text{SMOOTHING}}(\tilde{\boldsymbol{\lambda}}) \tag{3.61}$$

where $p_{\text{SMOOTHING}}(\tilde{\boldsymbol{\lambda}})$ was defined in (3.24) and $p_{\text{DISCRETISATION}}(\tilde{\boldsymbol{\lambda}})$ is defined according to A) or B). Shannon's information, described in depth in the context of emission tomography in Chapter 9, quantifies the information available about the vector-valued variable $\tilde{\boldsymbol{\lambda}}$ whose state of knowledge is described by the probability distribution $p(\tilde{\boldsymbol{\lambda}})$:

$$E_{p(\tilde{\boldsymbol{\lambda}})} = \int p(\tilde{\boldsymbol{\lambda}}) \log p(\tilde{\boldsymbol{\lambda}}) \mathrm{d}\boldsymbol{\lambda} \tag{3.62}$$

Information, as defined in (3.62), as can be easily verified, is additive:

$$E_{p(\tilde{\boldsymbol{\lambda}})} = E_{p_{\text{DISCRETISATION}}(\tilde{\boldsymbol{\lambda}})} + E_{p_{\text{SMOOTHING}}(\tilde{\boldsymbol{\lambda}})} \tag{3.63}$$

The choice of the number of voxels is irrelevant if the information associated to $p_{\text{SMOOTHING}}(\tilde{\boldsymbol{\lambda}})$ dominates the information associated to $p_{\text{DISCRETISATION}}(\tilde{\boldsymbol{\lambda}})$:

$$E_{p_{\text{DISCRETISATION}}(\tilde{\boldsymbol{\lambda}})} << E_{p_{\text{SMOOTHING}}(\tilde{\boldsymbol{\lambda}})} \tag{3.64}$$

The parameter β of the smoothing prior distribution regulates the extent to which the activity in nearby voxels is believed to be similar. While deciding the value of β is an open and challenging problem, the larger the value of β, the more informative is the prior $p_{\text{MRF}}(\tilde{\boldsymbol{\lambda}})$. The discretisation term is more informative the larger the size of the voxels d. The condition (3.64) is therefore equivalent to:

$$\frac{1}{\beta} >> d \tag{3.65}$$

In conclusion, the number of voxels must be large enough for the intended prior probability distribution (e.g. the smoothness prior) to hide the discretisation scheme. Note that this discussion is related to the voxel-based formulation of the discrete problem. In more general terms, voxels are one of the many possible choices of basis functions. Similar considerations apply, indeed, to other choices of the basis functions: if a particular choice of basis functions (or a given parametrisation of the basis functions)

induces a constraint in the continuous space of $\tilde{\boldsymbol{\lambda}}$, such constraint can be interpreted, in the Bayesian framework, as a prior probability distribution. One may opt for a specific choice of the basis functions for the purpose of obtaining a *regularisation* effect, however it is arguable that the effect of the discretisation algorithm should be kept separate from the prior belief.

Optimum design, regularisation (modelling), parameterisation, resolution, partial volume effect, are one problem. In the frequentist approach any set of rules is an estimator. Early-stopped ML or MAP, ML with large voxels, converged MAP are all candidate estimators. However it is difficult to define a criterion of optimality. The partial volume effect arises from using large voxels and from early stopping. From the Bayesian perspective, early termination of an algorithm has no justification and the size of the voxels must be chosen to be small enough for the discretisation scheme not to influence the decisions. The Bayesian framework places the inference task at the centre, separating clearly the model and the inference algorithm. Under the assumption of smoothness, smooth solutions are preferred. The partial volume effect arises from the assumption of smoothness.

The next section of this paragraph investigates the properties of the ML and MAP estimator with smoothing prior, determining a suitable size of the voxels and displaying how the partial volume effect arises from early termination and from convergent MAP.

Experiments

The experiments that follow investigate the effect of the smoothing prior, of early stopping and of the size of the voxels. Investigation of the properties of the reconstruction algorithms at convergence exposes the nature of the partial volume effect.

Uptake of FDG was simulated by assigning typical average values observed in PET FDG scans to the BrainWeb (http://mouldy. bic.mni.mcgill.ca/brainweb/) ground truth tissue model: 4 parts uptake in the gray matter (GM), 1 part in white matter (WM), 0 part in the cerebral-spinal fluid (CSF). The PET imaging process was simulated with the PET-Sorteo Monte Carlo simulator [59]. A single noiseless simulation of the PET imaging process was obtained with the PET-Sorteo Monte Carlo simulator (i.e. producing the term $A\boldsymbol{\lambda}$ in equation (3.32)). The noiseless measurement was scaled to have an integral sum equal to 100×10^6 photon counts. Finally, N instances of the PET measurement were generated by drawing N samples from the Poisson distribution (equation (3.32)) with expectation equal to the normalised noiseless measurement (notice that, due to the sum property of the Poisson ditribution, the total number of counts results Poisson distributed with mean 100×10^6). The random samples were obtained using a pseudo-random Poisson sampler (from the IRT toolbox [60]), based on the rejection sampling algorithm described at page 293 of [61].

Unconstrained MLEM is often considered non-convergent as at each iteration the image appears more and more noisy (see e.g. Fig. 3.12 at the end of the chapter). Such statement, however, is incorrect. If the system matrix is invertible, the Poisson log-likelihood is concave, presenting a unique maximiser. The Expectation Maximisation algorithm, as demonstrated by Dempster and Laird [12], guarantees,

given the properties of differentiability and continuity of the Poisson likelihood, convergence to the maximiser. With ill-conditioned system matrix, the log-likelihood is non-strictly concave, presenting an infinity of maximally likely solutions. In this case, the EM algorithm converges to one of the maximisers, depending on the initialisation. While the increasingly noisy aspect of the reconstruction is undesirable for visualisation, the likelihood increases with the number of iterations. The solution at convergence is indeed the most likely, or one of the many most likely. The poor appearance of the reconstruction at convergence is ultimately due to the small amount of information available from the measurement. Early stopping introduces a smoothing-like effect. While early stopping appears to be a valid alternative in the frequentist framework, defining an estimator, the Bayesian framework emphasises models and inference. There seems to be no question, in the Bayesian framework, of whether it is better to terminate early or not. The first question is how many iterations are required in order to achieve convergence of the MLEM algorithm. Fig. 3.8 displays the log-likelihood in the range 1 to 10000 iterations for varying size of the voxels. The reconstructions appears to achieve convergence in about 4000 iterations for large voxel sizes and 8000 iterations for smaller voxel sizes. The reason is discussed in paragraph 3.3.

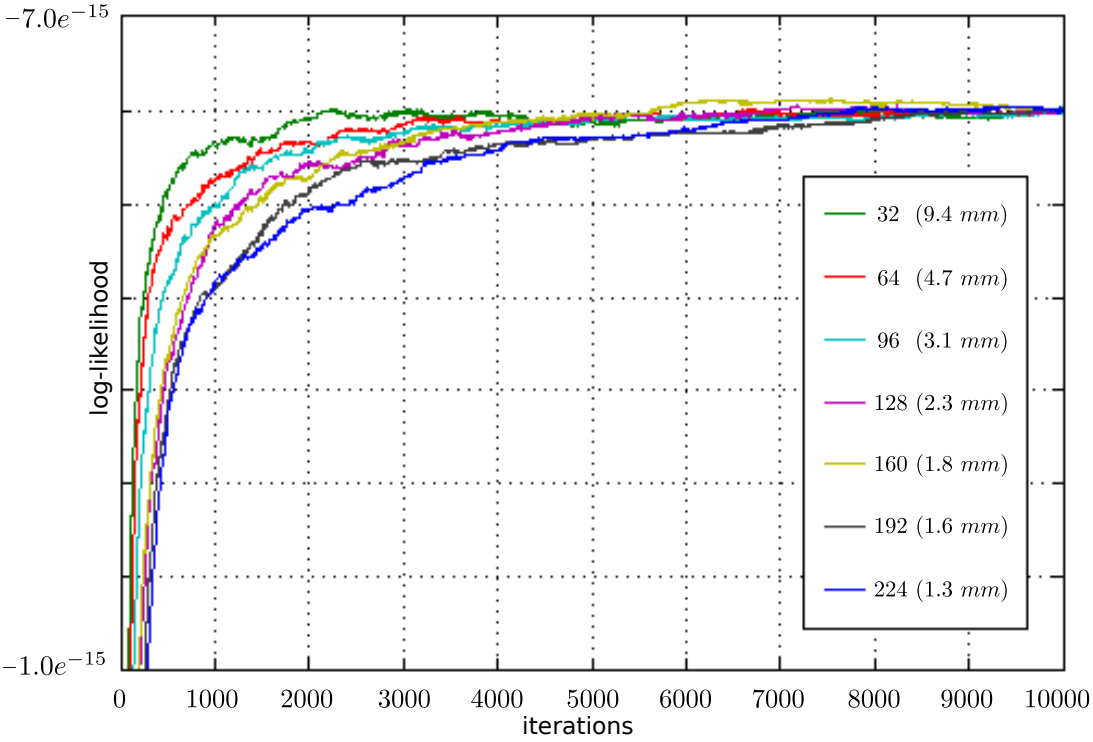


Figure 3.8: Convergence of the MLEM algorithm applied to 3-D PET with varying voxel size. The imaging volume of size $(300\ mm \times 300\ mm \times 300\ mm)$ is discretised in a regular grid of isotropic voxels, the number of voxels varying from $32 \times 32 \times 32$ to $224 \times 224 \times 224$. Convergence to machine precision (floating point precision $2.2e-16$) is achieved in $4000 - 8000$ iterations.

The second experiment investigates the effect of the smoothing prior and of early stopping. Bias/variance norm curves are computed, for different values of the smoothing parameter, from the reconstruction of N instances of the PET acquisition. In order to guarantee convergence, the OSL-MAPEM algorithm described in 3.2.2 was iterated for 10000 iterations. The voxel size was set to $1.6\ mm\ \ (192 \times 192 \times 8 voxels)$. Bias/variance norm curves are typically obtained from the reconstruction of 10 to 100 instances of the measurement. However, when the reconstruction algorithms are iterated until convergence, such small number of instances appears insufficient to provide an estimate of the covariance matrix. In order to be able to distinguish visually, from noise, the structure of the covariance matrix, it was found that the number of noise instances must be much larger. For the imaging system and average number of photon counts under consideration, $N = 10240$ produced satisfactory estimates of the covariance matrix, in terms of visual appearance. The visualisations of the covariance matrix reported in Fig. 3.11, in fact, appear dominated by random fluctuations for smaller values of N. Fig. 3.9 and the close up in Fig. 3.10 report the bias/variance norm curves obtained according to (3.59) and (3.60) from N reconstructions. The plot reports the curves obtained with unconstrained ML (red) and with MAP with quadratic first order MRF prior with 5 different values of the smoothing parameter β. The reconstruction algorithms utilised to compute the MAP estimates were MLEM and OSL-MAPEM, with 10000 iterations per reconstruction. The values of β were selected within a range selected by trial and error in order to obtain a considerable level of smoothing with the largest value β_5 and a small but visually noticeable level of smoothing with the smallest value β_1.

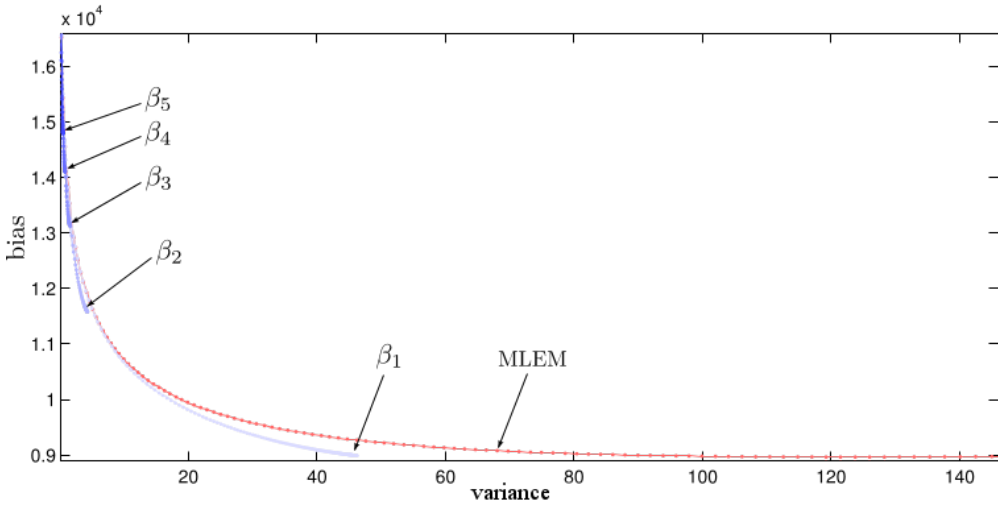


Figure 3.9: Bias/variance norm curves of MLEM (red) and of MAP with quadratic first order MRF (blue) with 5 values of the smoothing parameter β. The curves are obtained from 10240 reconstructions iterated 10000 times.

Interestingly, Fig. 3.9 and Fig. 3.10 suggests that MAP with the smoothness prior outperforms early stopping in terms of bias/variance for any choice of the smoothness parameter, achieving simultaneously

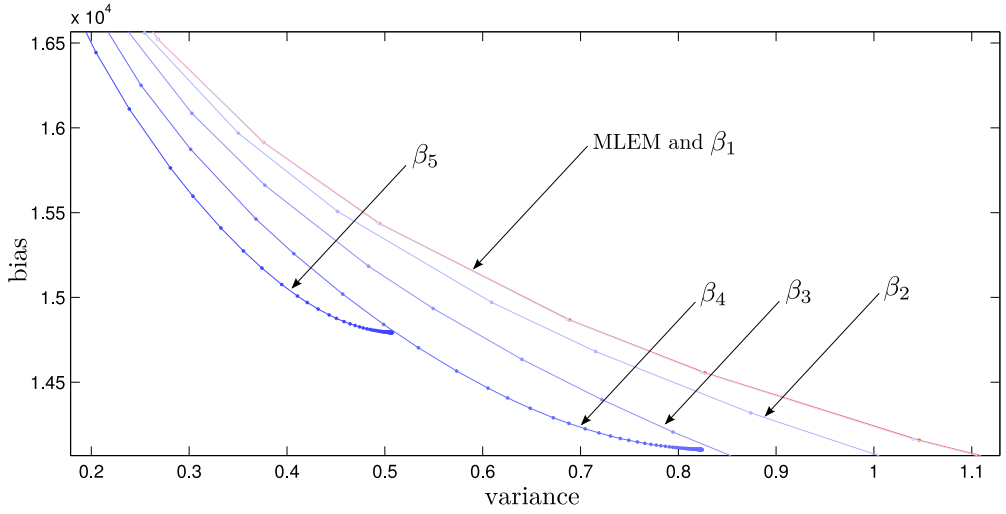


Figure 3.10: Bias/variance norm curves. Close up of the plot of Fig. 3.9.

lower bias and lower variance.

Fig. 3.11 displays the covariance matrices obtained from the multiple reconstructions. The blue curves display a column of the covariance matrix corresponding to a random point in the grey matter, while the red curves display a column corresponding to a random point in the white matter. Each of the two columns is re-arranged as a 3-D image, according to the voxel indexing pattern (raster scanning). The plots display a slice of such volume in the axial direction, passing through the voxel of interest. Semi-transparent plots refer to early stopping at $10, 100, 1000, 5000$ iterations. For high values of the smoothing parameters β_4 and β_5, the optimisation algorithm appears to converge within 100 iterations, as one deduces from the overlapping curves. The sharp peak of the covariance matrix in case of MLEM indicates that the estimates in different locations are nearly independent (voxels are indicated with a small dot in the plots of Fig. 3.11). Increasing values of the smoothing parameter spread the covariance matrix. Finally, one can observe that the covariance matrix obtained by early termination of MLEM (shaded curves) approximates the covariance matrix of the MAP estimator for varying values of the smoothing prior.

Fig. 3.12-3.17 display the mean of the 10240 reconstructions, the variance (diagonal of the covariance matrix, reshaped as an image), the bias (3.35) and a single reconstruction. Iteration until convergence of MLEM recovers the original image, if one performs the measurement again and again. However, the single reconstruction has an unappealing noisy appearance. For large values of the smoothing prior ($\beta_3, \beta_4, \beta_5$, the reconstruction from the single measurement appears visually appealing, though the mean of the reconstruction indicates the problem of the partial volume effect.

The reconstruction with the smoothing prior appears more appealing and easy to interpret. Clearly there is some truth in the model, as indicated by the lower bias for equal variance when compared with unconstrained (i.e. uninformed) optimisation (Fig. 3.9).

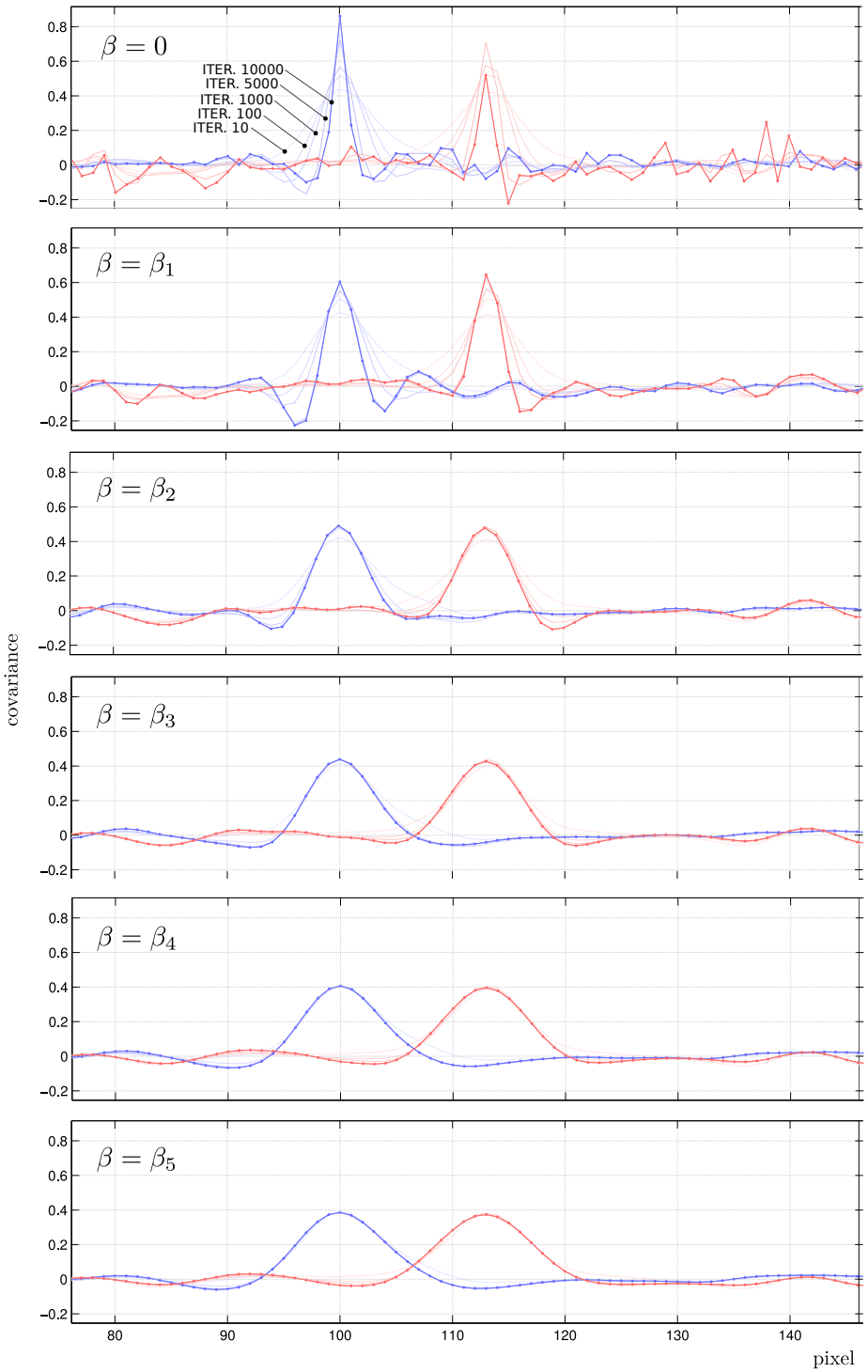


Figure 3.11: Visualisation of two columns of the covariance matrix $\Gamma(\boldsymbol{\lambda})$ for $N = 10240$ instances. Blue: random grey matter voxel. Red: random white matter voxel. Each of the two columns is re-arranged as a 3-D image, according to the voxel indexing pattern (raster scanning). The plots display a slice of such volume in the axial direction, passing through the voxel of interest. Semi-transparent plots refer to early stopping at $10, 100, 1000, 5000$ iterations.

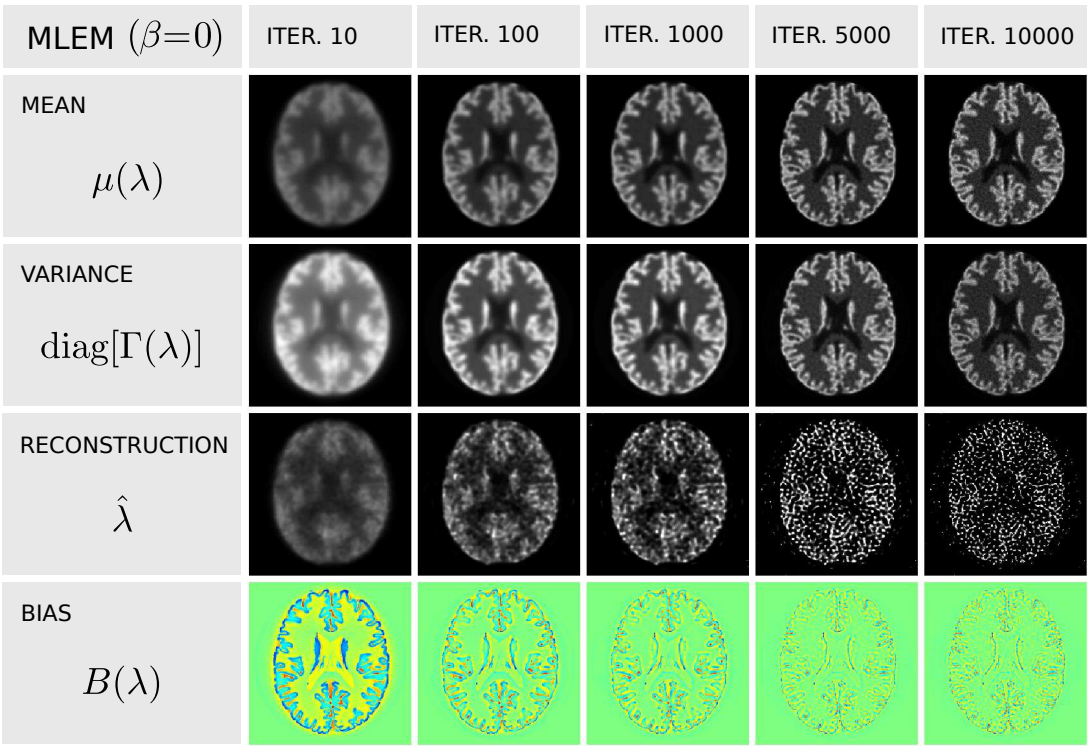


Figure 3.12: MLEM: $N = 10240$ instances, 10 to 10000 iterations.

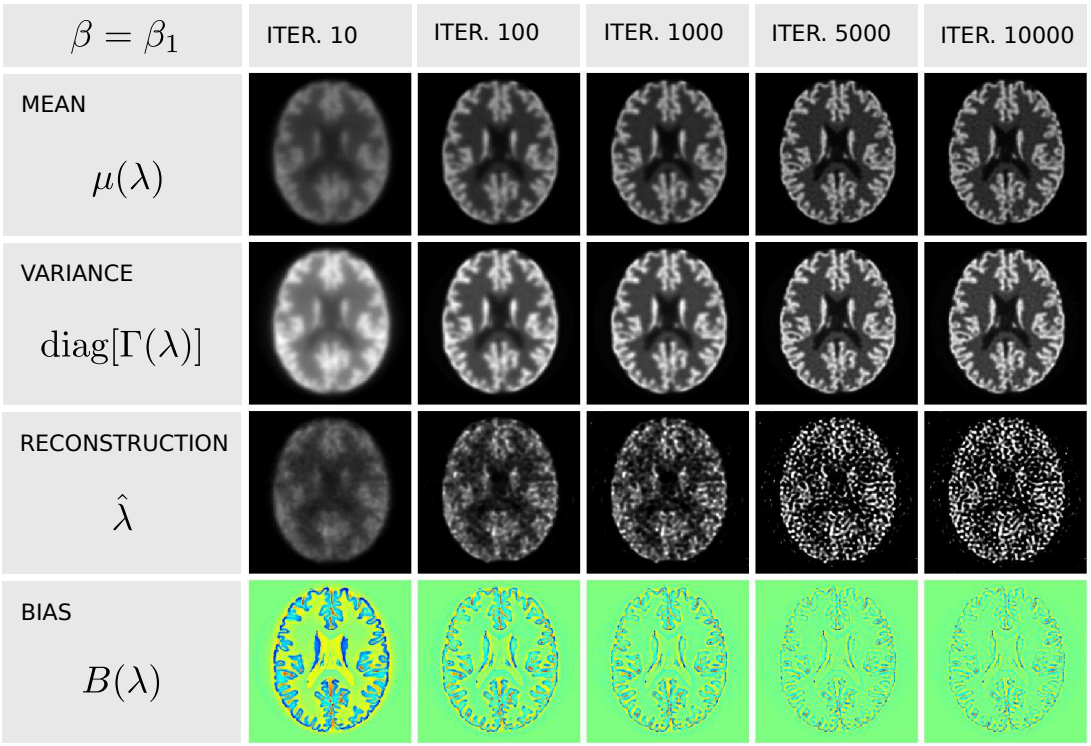


Figure 3.13: MAP-EM smoothing: $\beta = \beta_1$, $N = 10240$ instances, 10 to 10000 iterations.

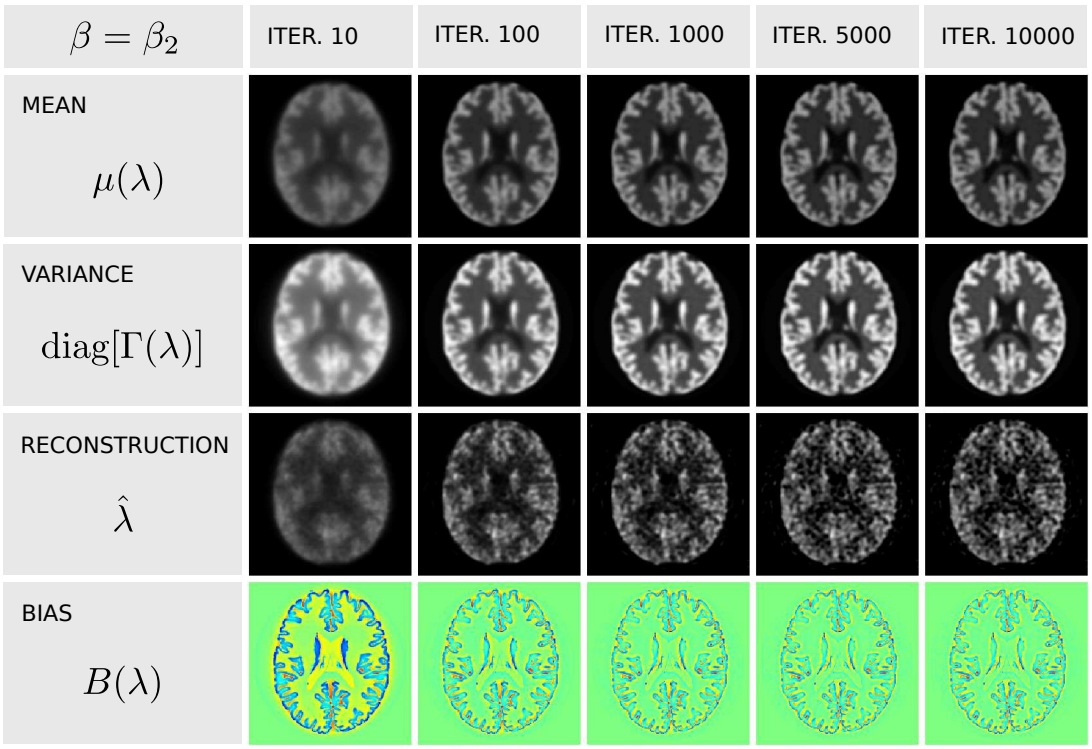


Figure 3.14: MAP-EM smoothing: $\beta = \beta_2$, $N = 10240$ instances, 10 to 10000 iterations.

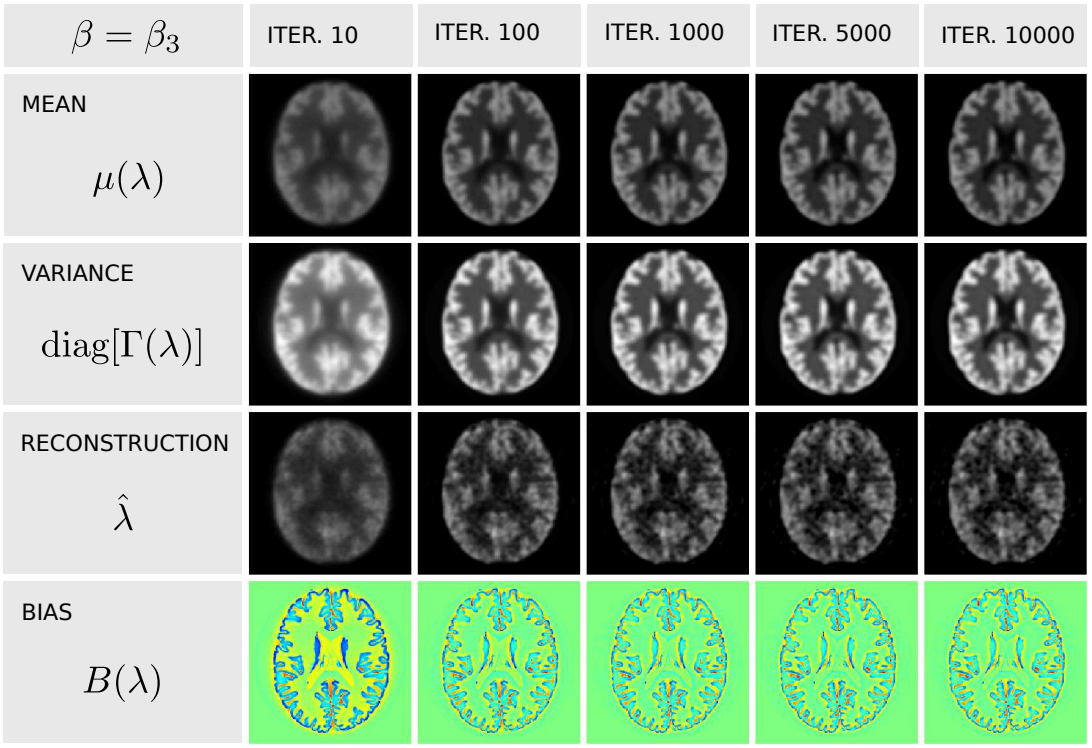


Figure 3.15: MAP-EM smoothing: $\beta = \beta_3$, $N = 10240$ instances, 10 to 10000 iterations.

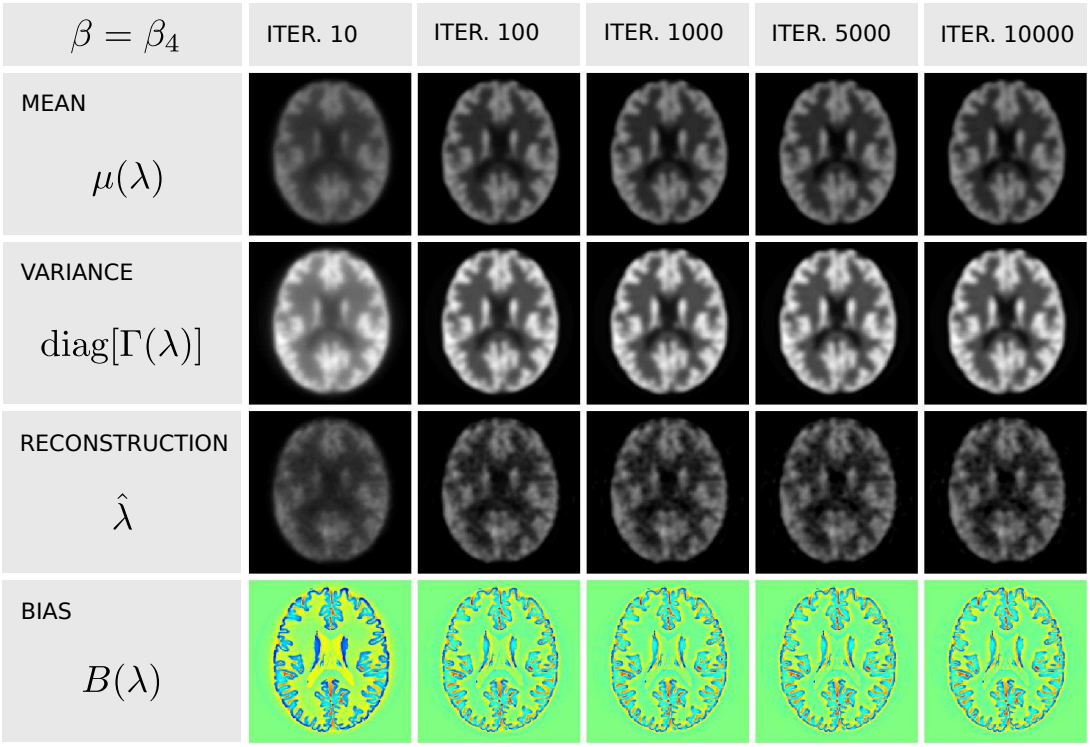


Figure 3.16: MAP-EM smoothing: $\beta = \beta_4$, $N = 10240$ instances, 10 to 10000 iterations.

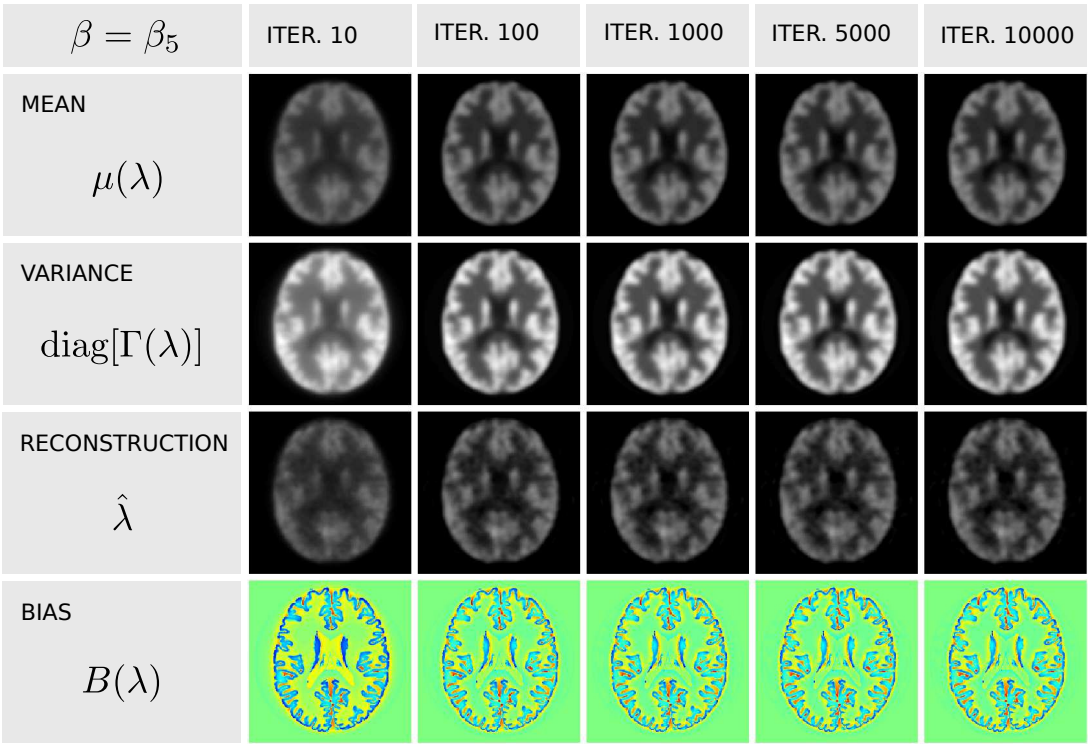


Figure 3.17: MAP-EM smoothing: $\beta = \beta_5$, $N = 10240$ instances, 10 to 10000 iterations.

3.A *The conditional expectation of the hidden variables of the Poisson model*

This appendix describes a derivation of the conditional expectation of the hidden variables z_{vd} introduced in paragraph 3.1.1 in the context of the derivation of the MLME formula for the Poisson model (refer to paragraph 3.1.1 for the definition of $z_{vd}, \boldsymbol{\lambda}, \boldsymbol{q}$). Although the conditional expectation of z_{vd} presents a simple and well known expression (see e.g. Green [30]):

$$\mathbb{E}_{p(z_{vd}|\boldsymbol{\lambda},\boldsymbol{q})}[z_{vd}] = \frac{a_{vd}\,\lambda_v\,q_d}{\sum_{v'=1}^{N_v} a_{v'd}\lambda_{v'}}, \tag{3.66}$$

deriving such expression is somewhat involved. Despite the widespread use of MLEM, this derivation seems to have not been reported in the literature of computed tomography. This appendix is organised as follows: paragraphs 3.A.1 and 3.A.2 define formulas that will be useful for the derivation of the conditional expectation of the hidden variables of the Poisson model; paragraphs 3.A.3 and 3.A.4 derive the expectation of a Poisson-distributed random variable and the sum property of the Poisson distribution, serving as introductions to the procedure for the derivation of the conditional expectation of the hidden variables of the Poisson model, described in the last paragraph (3.A.5).

3.A.1 Taylor expansion of the exponential function

The following equation expresses the Taylor polynomial expansion of the exponential function e^x (with $x \in \mathbb{R}$):

$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!} \tag{3.67}$$

Equation 3.67 will be used in the derivations that follow.

3.A.2 Multinomial theorem

The multinomial formula (see e.g. [62]) expresses, for any positive integer m and any nonnegative integer n, how a sum with m terms expands when raised to an arbitrary power n:

$$\sum_{\substack{k_1\geq 0\\ k_2\geq 0\\ \cdots\\ k_m\geq 0\\ \sum_{i=1}^{m} k_i = n}} \binom{n}{k_1, k_2, \ldots, k_m} \prod_{i=1}^{m} x_i^{k_i} = (x_1 + x_2 + \cdots + x_m)^n, \tag{3.68}$$

where

$$\binom{n}{k_1, k_2, \ldots, k_m} = \frac{n!}{k_1!\,k_2!\,\ldots\,k_m!} \tag{3.69}$$

is a multinomial coefficient (see e.g. [62]). Note that the sum is taken over all combinations of nonnegative integer indices k_1 through k_m such that the sum of all k_i is n. That is, for each term in the expansion, the exponents of the x_i must add up to n.

3.A.3 Expected value of a Poisson-distributed random variable

Let us consider a variable q distributed according to the Poisson distribution with parameters λ:

$$p(q|\lambda) = \mathcal{P}(q;\lambda) = \frac{e^{-\lambda}\lambda^q}{q!} \tag{3.70}$$

The expected value of the variable q under this distribution is:

$$\mathbb{E}_{p(q|\lambda)}[q] = \sum_{q=0}^{\infty} q\frac{e^{-\lambda}\lambda^q}{q!} \tag{3.71}$$

$$= \sum_{q=1}^{\infty} q\frac{e^{-\lambda}\lambda^q}{q!} \tag{3.72}$$

$$= \sum_{q=1}^{\infty} \frac{e^{-\lambda}\lambda^q}{(q-1)!} \tag{3.73}$$

With the change of variable $h = q - 1$, we obtain:

$$\mathbb{E}_{p(q|\lambda)}[q] = e^{-\lambda}\lambda\sum_{h=0}^{\infty}\frac{\lambda^h}{h!}, \tag{3.74}$$

where the infinite sum, as expressed by equation (3.67), converges to e^{λ}; therefore:

$$\mathbb{E}_{p(q|\lambda)}[q] = \lambda \tag{3.75}$$

3.A.4 Sum property of the Poisson distribution

The sum of two independent Poisson-distributed variables is Poisson distributed. Let us verify this property of the Poisson distribution. Let us consider two independent variables q_1 and q_2 distributed according to the Poisson distribution with parameters λ_1 and λ_2 respectively:

$$p(q_1|\lambda_1) = \mathcal{P}(q_1;\lambda_1) = \frac{e^{-\lambda_1}\lambda_1^{q_1}}{q_1!} \tag{3.76}$$

$$p(q_2|\lambda_2) = \mathcal{P}(q_2;\lambda_2) = \frac{e^{-\lambda_2}\lambda_2^{q_2}}{q_2!} \tag{3.77}$$

$$p(q_1, q_2|\lambda_1, \lambda_2) = p(q_1|\lambda_1)p(q_2|\lambda_2) \tag{3.78}$$

(equation (3.78) expresses the independence of the two random variables). The probability distribution associated to the sum of the two random variables

$$q = q_1 + q_2 \tag{3.79}$$

is expressed by the integral sum over the joint probability of q_1 and q_2, with the constraint $q_1 + q_2 = q$:

$$p(q|\lambda_1, \lambda_2) = \sum_{\substack{q_1\geq 0\\ q_2\geq 0\\ q_1+q_2=q}} p(q_1, q_2|\lambda_1, \lambda_2) \tag{3.80}$$

$$= \sum_{\substack{q_1\geq 0\\ q_2\geq 0\\ q_1+q_2=q}} \frac{e^{-\lambda_1}\lambda_1^{q_1}}{q_1!}\frac{e^{-\lambda_2}\lambda_2^{q_2}}{q_2!} \tag{3.81}$$

Let us multiply and divide by $q!$ and re-arrange the terms:

$$p(q|\lambda_1, \lambda_2) = \frac{e^{-(\lambda_1+\lambda_2)}}{q!} \sum_{\substack{q_1 \geq 0 \\ q_2 \geq 0 \\ q_1+q_2=q}} \frac{q!}{q_1!\, q_2!} \lambda_1^{q_1} \lambda_2^{q_2} \tag{3.82}$$

$$= \frac{e^{-(\lambda_1+\lambda_2)}}{q!} \sum_{\substack{q_1 \geq 0 \\ q_2 \geq 0 \\ q_1+q_2=q}} \binom{q}{q_1, q_2} \lambda_1^{q_1} \lambda_2^{q_2}, \tag{3.83}$$

where $\binom{q}{q_1, q_2}$ are the multinomial coefficients reported in equation (3.69). Using the multinomial formula (3.68), the summation in (3.83) is equivalent to $(\lambda_1 + \lambda_2)^q$, therefore we obtain:

$$p(q|\lambda_1, \lambda_2) = \frac{e^{-(\lambda_1+\lambda_2)}(\lambda_1 + \lambda_2)^q}{q!} \tag{3.84}$$

$$= \mathcal{P}\left(q; \lambda_1 + \lambda_2\right) \tag{3.85}$$

3.A.5 Conditional expectation of the hidden variables

Let us consider the model represented by the graph in Fig. 3.18-left, where $\boldsymbol{\lambda} = \{\lambda_v\}$, in the context of emission imaging, is the vector of rates of emission at the locations indexed by v. $\boldsymbol{q} = \{q_d\}$ is a vector of measurements (i.e. number of events detected at each detector location d); and the variables z_{vd} represent the number of events emitted in v and detected in d. Therefore we have:

$$q_d = \sum_v z_{vd} \tag{3.86}$$

Equation (3.86) can be expressed as a conditional probability distribution using Dirac's delta function:

$$p(q_d|z_{11}, z12, \ldots, z_{21}, \ldots) = \delta\left(q_d - \sum_v z_{vd}\right) \tag{3.87}$$

Let a_{vd} be the probability that an event emitted in v is detected in d. The probability distribution associated to z_{vd} conditional to the emission rates $\boldsymbol{\lambda}$ depends, by definition of z_{vd}, only on λ_v and is a Poisson distribution with parameter $a_{vd}\lambda_v$ if the events are emitted independently from one another:

$$p(z_{vd|\boldsymbol{\lambda}}) = \mathcal{P}\left(z_{vd}; a_{vd}\lambda_v\right) \tag{3.88}$$

Therefore, by the sum property of the Poisson distribution (that will be derived here in 3.A.4), the probability distribution associated to each of the measurements q_d is a Poisson distribution:

$$p(q_{d|\boldsymbol{\lambda}}) = \mathcal{P}\left(q_d; \sum_v a_{vd}\lambda_v\right) \tag{3.89}$$

For the probabilistic model defined by equations (3.87) and (3.88), we are interested in the expectation of each of z_{vd} when $\boldsymbol{\lambda}$ and $\boldsymbol{q}$ are known variables (as expressed by the shaded nodes of the graphs in Fig. 3.18). Let us notice, e.g. via the global Markov properties of the moralised graph of Fig. 3.18-right

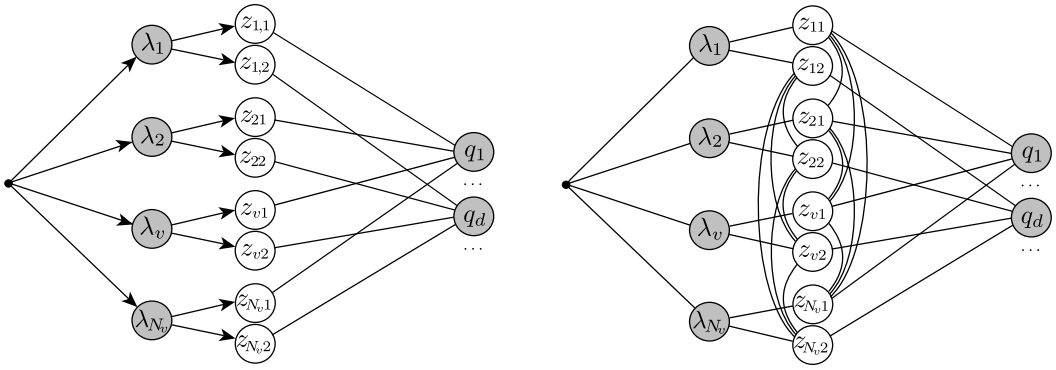


Figure 3.18: Left: graphical representation of the Poisson model explicitly showing the hidden variables $\mathbf{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_v, \ldots, \mathbf{z}_{N_v}\}$ with $\mathbf{z}_v = \{z_{1v}, z_{1v}, \ldots, z_{dv}, \ldots, z_{N_d v}\}$. The discrete variable z_{vd} indicates the number of photons emitted from spatial location v and detected at detector bin d. Right: moralised graph.

(see Paragraph 2.1.5), that the probability distribution of z_{11}, when $\boldsymbol{\lambda}$ and $\boldsymbol{q}$ are given, depends on q_1 and on $z_{21}, z_{31}, \ldots$ but does not depend on $q_2, q_3, \ldots$ nor on $z_{12}, z_{13}, \ldots, z_{22}, z_{23}, \ldots$. Therefore, when considering e.g. the probability distribution of z_{11} and its expectation, one may refer to the subgraph of Fig. 3.19-left.

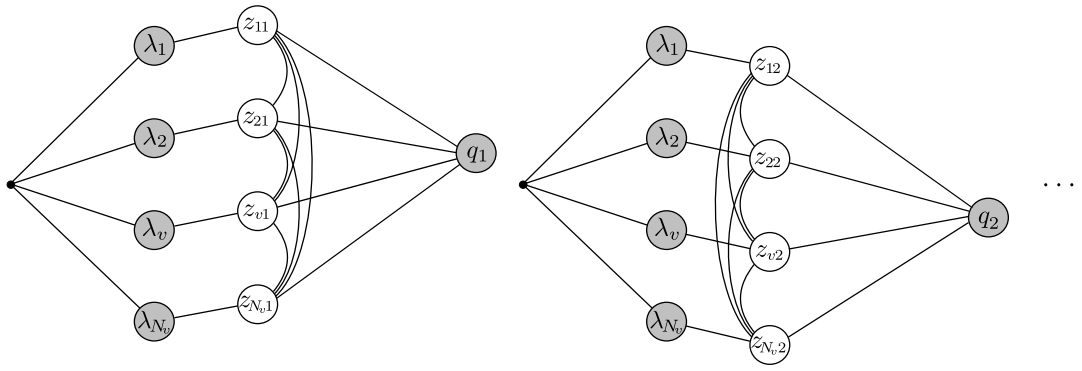


Figure 3.19: The probability distribution of $z_{11}, z_{12}, \ldots, z_{21}, z_{22}, \ldots$ conditional to $\boldsymbol{\lambda}, \boldsymbol{q}$, expressed by the graphs in Fig. 3.19, factorises into N_d independent terms. Cconditionally to $\boldsymbol{\lambda}, \boldsymbol{q}$, variable z_{11} depends on q_1 and on $z_{21}, z_{31}, \ldots$, but does not depend on $q_2, q_3, \ldots$ nor on $z_{12}, z_{13}, \ldots, z_{22}, z_{23}, \ldots$.

In order to keep the formulation uncluttered, rather than using the indexes v and d, let us consider, from here on, the case of $\boldsymbol{\lambda} = \{\lambda_1, \lambda_2, \lambda_3\}$ (i.e. $N_v = 3$) and derive the expectation of z_{11}. The generalisation is straightforward.

$$\mathbb{E}_{p(z_{11}|\lambda_1,\lambda_2,\lambda_3,q_1)}[z_{11}] \tag{3.90}$$

In order to derive the marginal conditional probability distribution of $p(z_{11}|\lambda_1, \lambda_2, \lambda_3, q_1)$, let us start

by considering the joint conditional probability of all the variables of the graph, then we will derive the conditional probability distribution of the hidden variables and finally marginalise with respect of all the hidden variables except for the variable of interest z_{11}. From the graph in Fig. 3.19-left, the joint probability of all the variables of the model has the following expression:

$$p(\lambda_1, \lambda_2, \lambda_3, z_{11}, z_{21}, z_{31}, d_1) = p(\lambda_1)p(\lambda_2)p(\lambda_3)p(z_{11}|\lambda_1)p(z_{21}|\lambda_2)p(z_{31}|\lambda_3)p(q_1|z_{11}, z_{21}, z_{31}) \tag{3.91}$$

Applying the law of total probability, we obtain:

$$\begin{aligned} p(z_{11}, z_{21}, z_{31}|\lambda_1, \lambda_2, \lambda_3, d_1) &= \frac{p(\lambda_1, \lambda_2, \lambda_3, z_{11}, z_{21}, z_{31}, d_1)}{p(\lambda_1, \lambda_2, \lambda_3, q_1)} \\ &= \frac{p(\lambda_1)p(\lambda_2)p(\lambda_3)p(z_{11}|\lambda_1)p(z_{21}|\lambda_2)p(z_{31}|\lambda_3)p(q_1|z_{11}, z_{21}, z_{31})}{p(\lambda_1)p(\lambda_2)p(\lambda_3)p(q_1|\lambda_1, \lambda_2, \lambda_3)} \\ &= \frac{p(z_{11}|\lambda_1)p(z_{21}|\lambda_2)p(z_{31}|\lambda_3)p(q_1|z_{11}, z_{21}, z_{31})}{p(q_1|\lambda_1, \lambda_2, \lambda_3)} \end{aligned} \tag{3.92}$$

The marginal probability distribution of z_{11} (conditional to $\lambda_1, \lambda_2, \lambda_3$ and d_1) is obtained by integrating the joint probability distribution of z_{11}, z_{21}, z_{31} of equation (3.92) over z_{21} and z_{31}, as follows:

$$p(z_{11}|\lambda_1, \lambda_2, \lambda_3, q_1) = \sum_{\substack{z_{21}\geq 0 \\ z_{31}\geq 0}} p(z_{11}, z_{21}, z_{31}|\lambda_1, \lambda_2, \lambda_3, d_1) \tag{3.93}$$

Finally, the expectation of z_{11} (when $\lambda_1, \lambda_2, \lambda_3$ and d_1 are given), is expressed by the following integral:

$$\mathbb{E}_{p(z_{11}|\lambda_1,\lambda_2,\lambda_3,q_1)}[z_{11}] = \sum_{z_{11}\geq 1} z_{11}\; p(z_{11}|\lambda_1, \lambda_2, \lambda_3, d_1) \tag{3.94}$$

(note that in equation (3.94) $\sum_{z_{11}\geq 1}$ is equivalent to $\sum_{z_{11}\geq 0}$; $\sum_{z_{11}\geq 1}$ is a more convenient choice as it simplifies the derivation that follows). Inserting (3.93) and (3.92) in (3.94), we obtain:

$$\mathbb{E}_{p(z_{11}|\lambda_1,\lambda_2,\lambda_3,q_1)}[z_{11}] = \sum_{\substack{z_{11}\geq 1 \\ z_{21}\geq 0 \\ z_{31}\geq 0}} z_{11}\; p(z_{11}, z_{21}, z_{31}|\lambda_1, \lambda_2, \lambda_3, d_1) \tag{3.95}$$

$$= \frac{1}{p(q_1|\lambda_1, \lambda_2, \lambda_3)} \sum_{\substack{z_{11}\geq 1 \\ z_{21}\geq 0 \\ z_{31}\geq 0}} z_{11} p(z_{11}|\lambda_1)p(z_{21}|\lambda_2)p(z_{31}|\lambda_3)p(q_1|z_{11}, z_{21}, z_{31}) \tag{3.96}$$

The last term of equation (3.96) is the delta function (3.87), therefore the summation can be rewritten as:

$$\mathbb{E}_{p(z_{11}|\lambda_1,\lambda_2,\lambda_3,q_1)}[z_{11}] = \frac{1}{p(q_1|\lambda_1, \lambda_2, \lambda_3)} \sum_{\substack{z_{11}\geq 1 \\ z_{21}\geq 0 \\ z_{31}\geq 0 \\ z_{11}+z_{21}+z_{31}=q_1}} z_{11}\; p(z_{11}|\lambda_1)p(z_{21}|\lambda_2)p(z_{31}|\lambda_3) \tag{3.97}$$

Inserting (3.88) into (3.97), then doing the change of variable $k = z_{11} - 1$; we obtain:

$$\mathbb{E}_{p(z_{11}|\lambda_1,\lambda_2,\lambda_3,q_1)}[z_{11}] = \frac{e^{-a_{11}\lambda_1}e^{-a_{21}\lambda_2}e^{-a_{31}\lambda_3}}{p(q_1|\lambda_1,\lambda_2,\lambda_3)} \sum_{\substack{z_{11}\geq 1\\ z_{21}\geq 0\\ z_{31}\geq 0\\ z_{11}+z_{21}+z_{31}=q_1}} z_{11}\, \frac{(a_{11}\lambda_1)^{z_{11}}}{z_{11}!}\frac{(a_{21}\lambda_2)^{z_{21}}}{z_{21}!}\frac{(a_{31}\lambda_3)^{z_{31}}}{z_{31}!} \tag{3.98}$$

$$= \frac{e^{-a_{11}\lambda_1}e^{-a_{21}\lambda_2}e^{-a_{31}\lambda_3}}{p(q_1|\lambda_1,\lambda_2,\lambda_3)} \sum_{\substack{z_{11}\geq 1\\ z_{21}\geq 0\\ z_{31}\geq 0\\ z_{11}+z_{21}+z_{31}=q_1}} \frac{(a_{11}\lambda_1)^{z_{11}}}{(z_{11}-1)!}\frac{(a_{21}\lambda_2)^{z_{21}}}{z_{21}!}\frac{(a_{31}\lambda_3)^{z_{31}}}{z_{31}!} \tag{3.99}$$

$$= \frac{e^{-a_{11}\lambda_1}e^{-a_{21}\lambda_2}e^{-a_{31}\lambda_3}}{p(q_1|\lambda_1,\lambda_2,\lambda_3)} \sum_{\substack{k\geq 0\\ z_{21}\geq 0\\ z_{31}\geq 0\\ k+z_{21}+z_{31}=q_1-1}} \frac{(a_{11}\lambda_1)^{k+1}}{k!}\frac{(a_{21}\lambda_2)^{z_{21}}}{z_{21}!}\frac{(a_{31}\lambda_3)^{z_{31}}}{z_{31}!} \tag{3.100}$$

Let us multiply and divide (3.100) by $(q_1 - 1)!$ and then apply the multinomial formula of equations (3.68)-(3.69):

$$\mathbb{E}_{p(z_{11}|\lambda_1,\lambda_2,\lambda_3,q_1)}[z_{11}] = \frac{e^{-a_{11}\lambda_1}e^{-a_{21}\lambda_2}e^{-a_{31}\lambda_3}}{(q_1-1)!\, p(q_1|\lambda_1,\lambda_2,\lambda_3)} \sum_{\substack{k\geq 0\\ z_{21}\geq 0\\ z_{31}\geq 0\\ k+z_{21}+z_{31}=q_1-1}} \frac{(q_1-1)!}{k!\, z_{21}!\, z_{31}!}(a_{11}\lambda_1)^{k+1}(a_{21}\lambda_2)^{z_{21}}(a_{31}\lambda_3)^{z_{31}} \tag{3.101}$$

$$= \frac{e^{-a_{11}\lambda_1}e^{-a_{21}\lambda_2}e^{-a_{31}\lambda_3}(a_{11}\lambda_1)}{(q_1-1)!\, p(q_1|\lambda_1,\lambda_2,\lambda_3)} \sum_{\substack{k\geq 0\\ z_{21}\geq 0\\ z_{31}\geq 0\\ k+z_{21}+z_{31}=q_1-1}} \binom{q_1-1}{z_{21}, z_{31}} (a_{11}\lambda_1)^{k}(a_{21}\lambda_2)^{z_{21}}(a_{31}\lambda_3)^{z_{31}} \tag{3.102}$$

$$= \frac{e^{-a_{11}\lambda_1}e^{-a_{21}\lambda_2}e^{-a_{31}\lambda_3}(a_{11}\lambda_1)}{(q_1-1)!\, p(q_1|\lambda_1,\lambda_2,\lambda_3)} (a_{11}\lambda_1 + a_{21}\lambda_2 + a_{31}\lambda_3)^{q_1-1} \tag{3.103}$$

Inserting equation (3.89) in (3.103), we obtain:

$$\mathbb{E}_{p(z_{11}|\lambda_1,\lambda_2,\lambda_3,q_1)}[z_{11}] = \frac{a_{11}\ \lambda_1\ q_1}{a_{11}\lambda_1 + a_{21}\lambda_2 + a_{31}\lambda_3} \tag{3.104}$$

Equation (3.104) is valid for every value of v of the N_v-dimensional vector $\boldsymbol{\lambda} = \{\lambda_v\}$ and for every value of d of the N_d-dimensional measurement $\boldsymbol{q} = \{q_d\}$, therefore:

$$\mathbb{E}_{p(z_{vd}|\boldsymbol{\lambda},\boldsymbol{q})}[z_{vd}] = \frac{a_{vd}\ \lambda_v\ q_d}{\sum_{v'=1}^{N_v} a_{v'd}\lambda_{v'}} \tag{3.105}$$

Chapter 4

Physiologically-based modelling

4.1 Physiologically-based modelling paradigm

Uncertainty in Emission Tomography is dominated by photon counting statistics. It is therefore essential to adopt a probabilistic model of the emission and interaction of the Gamma photons in order to use optimally the information at hand for the quantification of the uptake of the radio-pharmaceutical. Given the generative probabilistic model of the emission imaging system outlined in Chapter 2, the spatial density of the radio pharmaceutical can be estimated by Maximum Likelihood (ML), as described in Chapter 3. However the maximum likelihood estimation problem is ill-posed, presenting an infinity of equally likely solutions. Furthermore, the unknowns (pharmaceutical density in each voxel) are strongly correlated due to the measurement of line integrals by the emission imaging system. For these reasons, algorithms for the optimisation of the likelihood have slow convergence rates and the ML solutions present a noisy appearance (Chapter 3). A smoothing prior is typically adopted with the effect of reducing the iterations required to achieve convergence and selecting smooth solutions. The smoothing prior constitutes, from the Bayesian perspective, a source of information to the likes of emission imaging data. Being a partially truthful description of the process that generates the activity, the smoothing prior improves the recovery of the activity, as observed through the bias/variance plots in Chapter 3. PET images obtained under the assumption of smoothness, however, when overlaid on an MR image of the same subject (intra-subject), present the problem of *partial volume* or *spill out* effect: an observer can distinguish certain regions in common between the two images, but the PET image is smoother than the MR image, giving the observer the impression of spill out of the pharmaceutical. The partial volume effect highlights how problematic it is to quantify the uptake of pharmaceutical in a region of interest obtained by segmentation of the MR image.

The assumption of smoothness, described in Chapter 3, while interpreted as a *regularisation* strategy in the context of functional analysis, in the Bayesian framework assumes the meaning of a *mechanistic* model of the uptake of the pharmaceutical. One assumes that configurations of the uptake that vary slowly in space are more likely than configurations that present abrupt changes. Such assumption

may be considered a macroscopic probabilistic model of the *physiological processes* that underlie the spatial configuration of the uptake of pharmaceutical. In this work, the assumption of smoothness is replaced with a parametric model that relates the uptake of pharmaceutical to the anatomy, as imaged by the anatomical imaging modality.

Several publications have focused on improvement of the activity estimate by means of an intra-subject anatomical image, relying on the assumption that activity is related to the underlying anatomy, imaged by the MRI scanner. Methods in the literature fall into three main categories: methods that favor a piecewise uniform reconstruction by segmenting the anatomical image and subsequently applying a smoothing prior within each identified region [63]; methods that explicitly extract boundary information from the anatomical image and relax the effect of a global smoothing prior across the identified edges [64] and methods based on information theoretic similarity functionals [65]. The last category has proven particularly interesting, as the involved functionals do not require either explicit segmentation nor boundary extraction from the anatomical image. The introduction of anatomical prior information via these functionals has been shown, in synthetic experiments, to improve the *a posteriori* estimate of activity, by reducing *bias* and variance [63].

The declared intent of the use of similarity functionals is to produce reconstructions of the functional tomogram that are *similar* to the anatomical image. The approach adopted in this work differs from the aforementioned methodologies in that it is based on the modeling explicitly the joint imaging system. The probabilistic generative modeling approach described in Chapter 2 is adopted. While the similarity functionals hide the reasons of similarity between the images, the generative modelling approach expresses explicitly the models of the imaging systems, therefore exposing a clear model of the underlying physiological processes.

- model of the imaging systems
- model of the underlying physiological processes (macroscopic or microscopic)

Such modelling paradigm, will be referred to as *physiologically-based modelling*. Models that express the relation between observations and hidden variables related to the multiple available imaging modalities are meant to enable the concurrent estimation of quantities related to function and anatomy for automatic diagnosis and synthesis of images. Along this line of thought, recently Venkataraman *et al.* [66] have proposed a generative model for joint estimation of brain connectivity from functional MRI (fMRI) and diffusion weighted imaging (DWI) and Hiltunen *et al.* have proposed a combined reconstruction-classification method for single modality diffuse optical tomography [67]. Earlier related work on joint estimation of function and anatomy was developed by Sastri *et al.*[68] who introduced a tissue composition model based on a finite mixture and Rangarajan *et al.* [69] who introduced an iterative scheme for the same model, based on maximisation of mutual information.

The physiologically-based modeling paradigm and the models described in the remainder of this chapter and in the following chapters are inspired by the Bayesian framework for automated classification

of tissue described in [70] and in the Unified Segmentation framework [5].

4.2 Finite anatomical-functional states

A common approach for the extraction of anatomical information from MR images is to assume that there exist hidden discrete states. Such assumption underlies the models described by Van Leemput [70] and Ashburner [5]. While the assumption that there exist hidden finite states may be considered a mathematical abstraction, the hidden states may be interpreted, in the context of MR imaging, as anatomical states, such as gray matter, white matter and cerebrospinal fluid. As will be clarified by the end of this chapter, the association of the abstract hidden variables with states of anatomy presents certain difficulties that arise from the non-convexity of the optimisation problem associated to the estimation of the model parameters. Such difficulty however, as discussed, is effectively addressed by the use of population-derived prior information. The model that follows extends the models described by Van Leemput [70] and Ashburner [5] in the context of MR, assuming, in the context of MR-PET imaging, finite states of anatomy and function. It is assumed that there exist hidden discrete states of anatomy and function, such as healthy gray matter, healthy white matter, cerebrospinal fluid, hyper-active gray matter and hyper-active white matter.

The model described in the following is an expression of the physiologically-based paradigm, integrating models of the MRI and PET acquisition systems and a model of the uptake of the pharmaceutical in a unified framework. The model addresses fully automatic estimation of the parameters by integrating population-based prior information and a contextual model.

This paragraph first introduces the model of the PET acquisition system (4.2.1), followed by the hidden-state model of the pharmaceutical uptake (4.2.2) and the model of the MR acquisition system (4.2.3). Finally 4.2.4 describes the joint model. Paragraph (4.3) is dedicated to the description of an algorithm for the joint estimation of the model parameters.

4.2.1 Model of the PET acquisition system

The model of the emission imaging system, described in detail in Chapter 2.3, is expressed by the following conditional probability distribution of the photon counts $\boldsymbol{q}$ given the rate of emission of gamma radiation $\boldsymbol{\lambda}$, proportional to the spatial concentration of the radio-pharmaceutical:

$$p(\boldsymbol{q}|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} \mathcal{P}\left(q_d; \textstyle\sum_{v=1}^{N_v} a_{vd}\lambda_v\right) \tag{4.1}$$

The system matrix a_{vd} encompasses the characteristics of the imaging system (location, size, sensitivity, spatial resolution of the Gamma detectors, eventual collimators) and the attenuation of Gamma radiation through the patient. The model of the PET acquisition system is represented by the graphical model in Fig. 4.1-A.

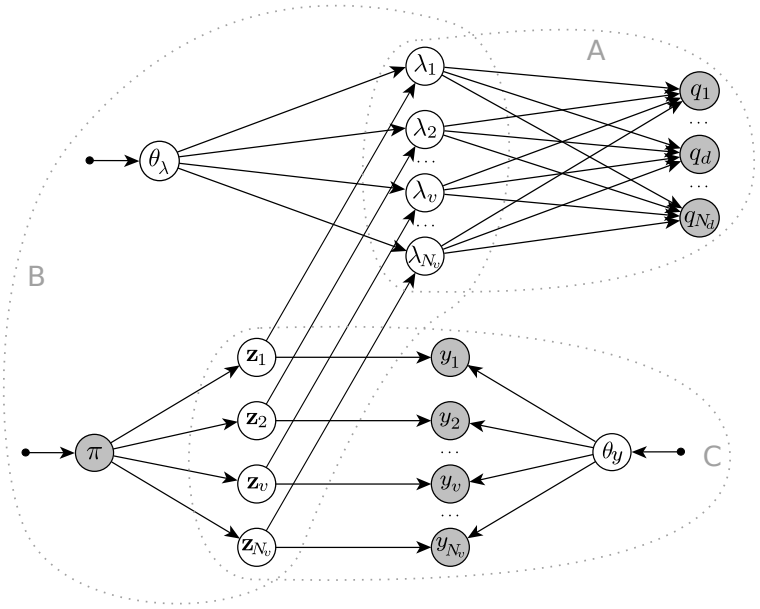


Figure 4.1: Finite anatomical-functional states model for PET-MR. A: model of the PET acquisition system that relates the pharmaceutical concentration λ_v in voxels $v = 1, \ldots, N_v$ to the photon counts in detector bins $d = 1, \ldots, N_d$; B: finite anatomical-functional states model; C: model of the MR acquisition system. The joint model expresses the assumption that there exists a finite discrete variable $\mathbf{z}_v$ for each voxel. If the state of the discrete variable in a voxel v were known, then one would associate a normal probability distribution to the outcome of the MR acquisition y_v. The set of normal distributions (one for each possible state) is parametrised by θ_y. If the hidden state in v were known, one would also associate a normal probability distribution to the concentration of the radio pharmaceutical λ_v, parametrised by the set of parameters θ_λ. π parametrises the spatially varying prior probability distribution of the hidden states.

4.2.2 Finite anatomical-functional states

It is assumed that there exists a finite number of tissue types and that, in a given type of tissue, the pharmaceutical uptake is considered uncertain, but to some extent predictable. The expectation and the extent of variation of the uptake within each type of tissue are captured by a parametric finite mixture model. What is intended exactly by *type of tissue* will be clarified once we introduce the use of population-derived information in the following.

Assuming that the hidden state may have one of N_k possible values, let us use a 1-of-N_k vectorial representation for the hidden states. The discrete state at location v is indicated by

$$\mathbf{z}_v = [z_{v1}, z_{v2}, \ldots, z_{vk}, \ldots, z_{vN_k}] \tag{4.2}$$

$$z_{vk} \in \{0, 1\} \qquad \sum_{k=1}^{N_k} z_{vk} = 1 \tag{4.3}$$

Let us indicate with $\mathbf{Z}$ the hidden states at all voxel locations:

$$\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_v, \ldots, \mathbf{z}_{N_v}] \tag{4.4}$$

The hidden variables are considered to be a priori independent from one another and their probability distribution is multinomial. This choice will allow us to introduce population-derived information. Note that the multinomial probability distribution specifies the probability of all possible states, allowing maximum flexibility in the definition of the prior probability distribution. The prior probability of the hidden states, function of the voxel location, will be derived, in the experiments in paragraph 4.5, from a population of images. Let us denote the spatially dependent prior probability of the hidden states with π_{vk}:

$$p(z_{vk} = 1|\pi) = \pi_{vk} \tag{4.5}$$

Let us indicate with π_v the prior probability of each of the allowed states in voxel v:

$$\pi_v = [\pi_{v1}, \pi_{v2}, \ldots, \pi_{vk}, \ldots, \pi_{vN_k}] \tag{4.6}$$

$$0 \leq \pi_{vk} \leq 1 \qquad \sum_{k=1}^{N_k} \pi_{vk} = 1 \tag{4.7}$$

and with π the set of all the parameters of the (spatially varying) multinomial prior probability distributions of the hidden states:

$$\pi = [\pi_1, \pi_2, \ldots, \pi_v, \ldots, \pi_{N_v}] \tag{4.8}$$

Because the hidden state $\mathbf{z}_v$ in v uses a 1-of-N_k representation, the prior probability distribution associated to the state in v can also be written as:

$$p(\mathbf{z}_v|\pi) = \prod_{k=1}^{N_k} \pi_{vk}^{z_{vk}} \tag{4.9}$$

This second notation turns out to be often more convenient. The prior probability associated to the set of all the hidden variables is:

$$p(\mathbf{Z}|\pi) = \prod_{v=1}^{N_v} \prod_{k=1}^{N_k} \pi_{vk}^{z_{vk}} \tag{4.10}$$

The concentration of the radio-pharmaceutical λ_v in a voxel v belonging to class k is assumed to be normally distributed around a certain mean $\mu_{\lambda,k}$, with variance $\sigma^2_{\lambda,k}$, grouped in $\theta_{\lambda,k} = \{\mu_{\lambda,k}, \sigma^2_{\lambda,k}\}$ and $\theta_\lambda = \{\theta_{\lambda_1}, \theta_{\lambda_2}, \ldots, \theta_{\lambda_{N_k}}\}$:

$$p(\lambda_v|z_{vk} = 1, \theta_\lambda) = \mathcal{N}(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}) \tag{4.11}$$

Because the hidden state $\mathbf{z}_v$ in v is expressed with a 1-of-N_k representation, this last expression can be written as:

$$p(\lambda_v|\mathbf{z}_v, \theta_\lambda) = \prod_{k=1}^{N_k} \mathcal{N}(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k})^{z_{vk}} \tag{4.12}$$

which entails that $\lambda_v \perp \lambda_{v'}|\mathbf{z}, \theta_\lambda$:

$$p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_\lambda) = \prod_{v=1}^{N_v} \prod_{k=1}^{N_k} \mathcal{N}\left(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}\right)^{z_{vk}} \tag{4.13}$$

Here, $\theta_\lambda = \{\theta_{\lambda_1}, \theta_{\lambda_2}, \ldots, \theta_{\lambda_{N_k}}\}$ are the parameters of the pharmaceutical uptake model, describing the expectation and the extent of variation of the uptake in each type of tissue. The generative model that relates the spatial concentration of the radio-pharmaceutical to the anatomical-functional states is represented by the graphical model in Fig. 4.1-B.

4.2.3 Model of the MR acquisition system

The MR imaging system is described by a parametric voxel-based finite mixture model commonly employed for the classification of tissue types by means of MR images [71]. It relates the observed image intensities to the underlying finite hidden states. The intensity y_v of a voxel v that belongs to class k is assumed to be normally distributed around a certain mean $\mu_{y,k}$, with variance $\sigma^2_{y,k}$. Let us indicate the parameters compactly with $\theta_{y,k} = \{\mu_{y,k}, \sigma_{y,k}\}$ and $\theta_y = \{\theta_{y_1}, \theta_{y_2}, \ldots, \theta_{y_{N_k}}\}$, then, we get:

$$p(y_v|z_{vk} = 1, \theta_y) = \mathcal{N}\left(y_v; \mu_{y,k}, \sigma_{y,k}\right) \tag{4.14}$$

Again, because the hidden state $\mathbf{z}_v$ in v uses a 1-of-N_k representation, this expression can be written as:

$$p(y_v|\mathbf{z}_v, \theta_y) = \prod_{k=1}^{N_k} \mathcal{N}\left(y_v; \mu_{y,k}, \sigma_{y,k}\right)^{z_{vk}} \tag{4.15}$$

which entails that $y_v \perp y_{v'}|\mathbf{Z}, \theta_y$:

$$p(\boldsymbol{y}|\mathbf{Z}, \theta_y) = \prod_{v=1}^{N_v} \prod_{k=1}^{N_k} \mathcal{N}\left(y_v; \mu_{y,k}, \sigma_{y,k}\right)^{z_{vk}} \tag{4.16}$$

The model of the MR imaging system (4.15) is represented by the Bayesian Network in Fig. 4.1-C. The set of parameter are the mean and spread of the intensity for each class $\theta_y = \{\theta_{y_1}, \theta_{y_2}, \ldots, \theta_{y_{N_k}}\}$.

4.2.4 The joint model

The hidden tissue states are considered the unique underlying cause of mutual dependence of the two images, as expressed by the arrows of the graph in Fig. 4.1. The hidden state in each voxel is related to the intensity of the MR image and to the pharmaceutical density (which is also a hidden variable) by the parametric models of Sec. 4.2.2 and Sec. 4.2.3.

The probability of voxel v being in state k depends a priori on the location of v. Such prior distribution is expressed by the multinomial probability distribution π_{vk}, $\sum_k^{N_k} \pi_{vk} = 1$. The use of the spatially varying multinomial prior probability of the hidden states is key to automating the algorithm, as otherwise the parameters of the mixture models converge to different tissue classes non predictably.

The joint model is expressed graphically by the DAG in Fig. 4.1 and by the compact representation in Fig. 4.2.

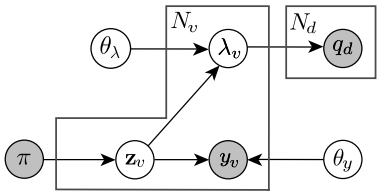


Figure 4.2: Compact representation of the finite anatomical-functional states model of figure 4.1.

4.3 Joint estimation of the unobserved variables

Let us consider the MAP estimate of $\boldsymbol{\lambda}$ and θ:

$$\hat{\boldsymbol{\lambda}}, \hat{\theta} = \underset{\boldsymbol{\lambda},\theta}{\operatorname{argmax}} \log p(\boldsymbol{\lambda}, \theta | \mathbf{q}, \mathbf{y}, \pi) \tag{4.17}$$

In order to derive the expression of the joint posterior, let us start from the joint probability of all variables, easily obtained from the DAG of Fig. 4.2. One methodology to obtain the joint probability distribution from the DAG is the following: one assigns an ordering to the variables such that there are no links from any node to any lower numbered node; the joint probability distribution can be written as a sequence of products, starting from the lowest numbered node and following increasing node indexes, of the probability distribution of the variables at each node conditional to the parent nodes. In the case of Fig. 4.2:

$$p(\pi, \theta, \mathbf{Z}, \boldsymbol{y}, \boldsymbol{\lambda}, \boldsymbol{q}) = p(\pi)p(\theta_\lambda)p(\theta_y)p(\mathbf{Z}|\pi)p(\boldsymbol{y}|\mathbf{Z}, \theta_{\boldsymbol{y}})p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_{\boldsymbol{\lambda}})p(\boldsymbol{q}|\boldsymbol{\lambda}) \tag{4.18}$$

From the chain rule of conditional probabilities:

$$p(\boldsymbol{\lambda}, \theta, \mathbf{Z}|\mathbf{q}, \mathbf{y}, \pi) = \frac{p(\pi, \theta, \mathbf{Z}, \boldsymbol{y}, \boldsymbol{\lambda}, \boldsymbol{q})}{p(\mathbf{q}, \mathbf{y}, \pi)} \tag{4.19}$$

Marginalising over $\mathbf{Z}$:

$$p(\boldsymbol{\lambda}, \theta|\mathbf{q}, \mathbf{y}, \pi) = \frac{1}{p(\mathbf{q}, \mathbf{y}, \pi)} \sum_{\mathbf{Z}} p(\pi, \theta, \mathbf{Z}, \boldsymbol{y}, \boldsymbol{\lambda}, \boldsymbol{q}) \tag{4.20}$$

Since we are interested in maximising this expression with respect to $\boldsymbol{\lambda}$ and θ, we are only interested in the term that appears in the summation, i.e. the marginal of the joint posterior of all the variables with respect to $\mathbf{Z}$. The summation over all possible values of $\mathbf{Z}$ seems to make the expression of the posterior marginalised with respect to $\mathbf{Z}$ complicated. However it does have a simple expression, which can be obtained by noting that the model implies the following factorisation (this can be easily verified e.g. by moralising the graph of Fig. 4.1):

$$p(\boldsymbol{y}, \boldsymbol{\lambda}|\pi, \theta_y, \theta_\lambda) = \prod_{v=1}^{N_v} p(y_v, \lambda_v|\pi, \theta_y, \theta_\lambda), \tag{4.21}$$

where:

$$p(y_v, \lambda_v|\pi, \theta_y, \theta_\lambda) = \sum_{\mathbf{z}_v} p(\mathbf{z}_v|\pi)p(y_v|\mathbf{z}_v, \theta_y)p(\lambda_v|\mathbf{z}_v, \theta_\lambda) \quad (4.22)$$

$$= \sum_{\mathbf{z}_v} \pi_{vk}^{z_{vk}} p(y_v|\mathbf{z}_v, \theta_y)p(\lambda_v|\mathbf{z}_v, \theta_\lambda) \quad (4.23)$$

$$= \sum_{\mathbf{z}_v} \prod_{k=1}^{N_k} \pi_{vk}^{z_{vk}} \prod_{k=1}^{N_k} \mathcal{N}\left(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}\right)^{z_{vk}} \prod_{k=1}^{N_k} \mathcal{N}\left(y_v; \mu_{y,k}, \sigma_{y,k}\right)^{z_{vk}} \quad (4.24)$$

$$= \sum_{k=1}^{N_k} \pi_{vk} \mathcal{N}\left(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \mathcal{N}\left(y_v; \mu_{y,k}, \sigma_{y,k}\right) \quad (4.25)$$

The last equality is a consequence of the 1-of-N_k representation of $\mathbf{z}_v$ (the possible values of $\mathbf{z}_v$ are $\{[0, 0, \ldots, 0, 0, 1], [0, 0, \ldots, 0, 1, 0], \ldots\}$). From (4.21) and (4.25), we get:

$$p(\boldsymbol{y}, \boldsymbol{\lambda}|\pi, \theta_y, \theta_\lambda) = \prod_{v=1}^{N_v} \sum_{k=1}^{N_k} \pi_{vk} \mathcal{N}\left(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \mathcal{N}\left(y_v; \mu_{y,k}, \sigma_{y,k}\right) \quad (4.26)$$

Substituting (4.18) in (4.20), and given equation (4.26) and the following equality (marginalisation over variable $\mathbf{Z}$):

$$p(\mathbf{Z}|\pi)p(\boldsymbol{y}|\mathbf{Z}, \theta_{\boldsymbol{y}})p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_{\boldsymbol{\lambda}}) = p(\boldsymbol{y}, \boldsymbol{\lambda}|\pi, \theta_y, \theta_\lambda), \quad (4.27)$$

the log of the posterior distribution of $\boldsymbol{\lambda}, \theta$ has the following expression:

$$\log p(\boldsymbol{\lambda}, \theta|\boldsymbol{q}, \boldsymbol{y}, \pi) \propto \log p(\theta_y) + \log p(\theta_\lambda) + \log p(\boldsymbol{q}|\boldsymbol{\lambda}) + \sum_{v=1}^{N_v} \log \sum_{k=1}^{N_k} \pi_{vk} \mathcal{N}\left(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \mathcal{N}\left(y_v; \mu_{y,k}, \sigma_{y,k}\right) \quad (4.28)$$

The equations above express the conditional probability distributions of all the subsets. The section that follows describes the use of the ICM algorithm for the joint optimisation of the parameters, based on the iterative maximisation of the conditional probability distributions of the subsets.

Iterated conditional modes

If the prior probability distributions $p(\theta_y)$ and $p(\theta_\lambda)$ are differentiable, (4.28) is differentiable with respect to $\boldsymbol{\lambda}$ and θ (the expression of the first derivatives of the term $\log p(\boldsymbol{q}|\boldsymbol{\lambda})$ has been reported in (2.3.3)). In order to optimise the unknown variables jointly, one can proceed with a projected gradient ascent algorithm, with a suitable projection operator to guarantee positivity of the solution for $\boldsymbol{\lambda}$, μ_λ and μ_y. However, as discussed in 2.1.10, we wish to find a partitioning of the variables in subsets such that the optimisation of the conditional probability distribution of each subset is simpler than the joint optimisation. The joint optimisation is then performed by alternating the optimisation of the marginal posterior of the variables of each subset in turn, conditional to the provisional estimates of the variables in all other subsets (iterated conditional modes algorithm (ICM) - see paragraph 2.1.10). Partitioning the unknown variables in the following two subsets:

- $\mathcal{S}_1 : \theta = \{\theta_\lambda, \theta_y\}$
- $\mathcal{S}_2 : \boldsymbol{\lambda}$.

The ICM algorithm consists in alternating the following optimisations:

$$\theta^{(n+1)} = \underset{\theta}{\operatorname{argmax}} \log p\left(\theta | \boldsymbol{q}, \boldsymbol{y}, \boldsymbol{\lambda}, \pi\right) |_{\boldsymbol{\lambda}^{(n)}} \tag{4.29}$$

$$\boldsymbol{\lambda}^{(n+1)} = \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \log p\left(\boldsymbol{\lambda} | \boldsymbol{q}, \boldsymbol{y}, \theta, \pi\right) |_{\theta^{(n)}} \tag{4.30}$$

This particular choice of subsets is convenient as the optimisation problem (4.29) can be formulated in the expectation maximisation framework, obtaining a parameter-free update formula. For the second subset we can apply the parameter-free OSL algorithm described in 3.2.1. The two sections that follow derive update formulae for (4.29) and (4.30).

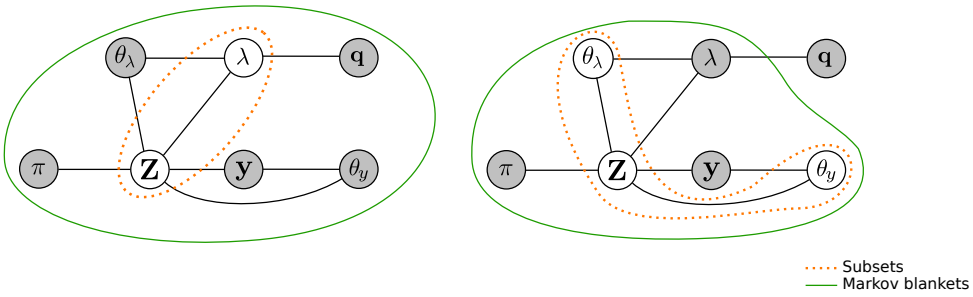


Figure 4.3: Moralised MRF of the model of Fig. 4.1 and 4.2 and visualisation of the Iterated Conditional Modes algorithm for the joint optimisation of the model parameters $\boldsymbol{\lambda}, \theta_{\boldsymbol{\lambda}}, \theta_y$. The ICM algorithm alternates the optimisation of $p\left(\theta | \boldsymbol{q}, \boldsymbol{y}, \boldsymbol{\lambda}, \pi\right) |_{\boldsymbol{\lambda}^{(n)}}$ and $p\left(\boldsymbol{\lambda} | \boldsymbol{q}, \boldsymbol{y}, \theta, \pi\right) |_{\theta^{(n)}}$. The two conditional distributions are obtained by marginalising $p\left(\theta, \mathbf{Z} | \boldsymbol{q}, \boldsymbol{y}, \boldsymbol{\lambda}, \pi\right) |_{\boldsymbol{\lambda}^{(n)}}$ and $p\left(\boldsymbol{\lambda}, \mathbf{Z} | \boldsymbol{q}, \boldsymbol{y}, \theta, \pi\right) |_{\theta^{(n)}}$ over $\mathbf{Z}$. The graph displays the Markov blankets of $\boldsymbol{\lambda}, \mathbf{Z}$ and $\boldsymbol{y}, \mathbf{Z}$, i.e. the set of variables involved in the two conditional distributions.

Model parameters

Let us consider the maximisation of the conditional probability of the model parameters given the provisional estimate of $\boldsymbol{\lambda}$:

$$\theta^{(n+1)} = \underset{\theta}{\operatorname{argmax}} \log p\left(\theta | \mathbf{q}, \boldsymbol{y}, \boldsymbol{\lambda}, \pi\right) |_{\boldsymbol{\lambda}^{(n)}} \tag{4.31}$$

From the moralised graph (Fig. 4.3): $\theta \perp \boldsymbol{q} | \boldsymbol{\lambda}$:

$$p(\theta | \boldsymbol{q}, \boldsymbol{y}, \boldsymbol{\lambda}, \pi) = p(\theta | \boldsymbol{y}, \boldsymbol{\lambda}, \pi). \tag{4.32}$$

The expression $p(\theta | \boldsymbol{y}, \boldsymbol{\lambda}, \pi)$ represents the likelihood of a bi-variate mixture model, studied in detail in paragraph 4.A. Paragraph 4.A clarifies why this expression is the likelihood of a bi-variate mixture and

describes the derivation of the EM update formula for the maximisation of $p(\theta|\boldsymbol{y}, \boldsymbol{\lambda}, \pi)$ with respect to θ. Let us report here the final result, the EM update formula:

$$l_{vk} \triangleq \frac{\pi_{vk}\mathcal{N}\left(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}\right)\mathcal{N}\left(y_v; \mu_{y,k}, \sigma_{y,k}\right)}{\sum_{k=1}^{N_k}\pi_{vk}\mathcal{N}\left(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}\right)\mathcal{N}\left(y_v; \mu_{y,k}, \sigma_{y,k}\right)} \tag{4.33}$$

$$\mu_{y,k}^{(n+1)} = \frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk} y_v \tag{4.34}$$

$$\sigma_{y,k}^{(n+1)} = \left[\frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk}\left(\mu_{y,k}^{(n+1)} - y_v\right)^2\right]^{\frac{1}{2}} \tag{4.35}$$

$$\mu_{\lambda,k}^{(n+1)} = \frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk} \lambda_v \tag{4.36}$$

$$\sigma_{\lambda,k}^{(n+1)} = \left[\frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk}\left(\mu_{\lambda,k}^{(n+1)} - \lambda_v\right)^2\right]^{\frac{1}{2}}, \tag{4.37}$$

where $\theta = \{\theta_\lambda, \theta_y\}$, $\theta_\lambda = \{\theta_{\lambda_1}, .., \theta_{\lambda_N}\}$, $\theta_y = \{\theta_{y_1}, .., \theta_{y_N}\}$, $\theta_{y,k} = \{\mu_{\lambda,k}, \sigma_{y,k}\}$ and $\theta_{\lambda,k} = \{\mu_{\lambda,k}, \sigma_{\lambda,k}\}$.

Radio-Pharmaceutical concentration

Let us now consider the maximisation of the conditional probability of the activity $\boldsymbol{\lambda}$ given the provisional estimate of the model parameters θ:

$$\boldsymbol{\lambda}^{(n+1)} = \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \log p\left(\boldsymbol{\lambda}|\boldsymbol{q}, \boldsymbol{y}, \theta, \pi\right)|_{\theta^{(n)}} \tag{4.38}$$

This is the *log* of the posterior probability distribution of $\boldsymbol{\lambda}$, conditioned on the provisional estimates of the model parameters θ. Let us consider its first derivatives with respect to vector valued λ:

$$\frac{\partial}{\partial\lambda_r} \log p\left(\boldsymbol{\lambda}|\boldsymbol{q}, \boldsymbol{y}, \theta, \pi\right) = \tag{4.39}$$

$$= \frac{\partial}{\partial\lambda_r} \log \sum_{\mathbf{Z}} p\left(\boldsymbol{\lambda}, \mathbf{Z}|\boldsymbol{q}, \boldsymbol{y}, \theta, \pi\right) \tag{4.40}$$

$$= \frac{\partial}{\partial\lambda_r} \log \sum_{\mathbf{Z}} \frac{p\left(\boldsymbol{\lambda}, \mathbf{Z}, \boldsymbol{q}, \boldsymbol{y}, \theta, \pi\right)}{p\left(\boldsymbol{q}, \boldsymbol{y}, \theta, \pi\right)} \tag{4.41}$$

$$= \frac{\partial}{\partial\lambda_r} \log \left[p\left(\boldsymbol{q}|\boldsymbol{\lambda}\right) \sum_{\mathbf{Z}} p(\mathbf{Z}|\pi) p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_\lambda) p(\boldsymbol{y}|\mathbf{Z}, \theta_y)\right] \tag{4.42}$$

$$= \underbrace{\frac{\partial}{\partial\lambda_r} \log \prod_d p(q_d|\boldsymbol{\lambda})}_{\text{Emission term}} + \underbrace{\frac{\partial}{\partial\lambda_r} \log \sum_{\mathbf{Z}} p(\mathbf{Z}|\pi) p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_\lambda) p(\boldsymbol{y}|\mathbf{Z}, \theta_y)}_{\text{Finite mixture term}} \tag{4.43}$$

The gradient of the conditional probability distribution of $\boldsymbol{\lambda}$ (4.43) is composed of two terms; the first term is the likelihood of the Poisson model associated to the photon emission and is therefore indicated, from here on, as *emission term*; the second term arises from the finite mixture model associated to the

discrete hidden states, therefore it is indicated as *finite mixture term*. The emission term has been derived in Chapter 2 (equation (2.3.3)). Expanding the finite mixture term:

$$\underbrace{\frac{\partial}{\partial\lambda_r}\log\sum_{\mathbf{Z}}p(\mathbf{Z}|\pi)p(\boldsymbol{\lambda}|\mathbf{Z},\theta_\lambda)p(\boldsymbol{y}|\mathbf{Z},\theta_y)}_{\text{Finite mixture term}}= \tag{4.44}$$

$$= \frac{\partial}{\partial\lambda_r}\log\prod_{v}^{N_v}\sum_{\mathbf{z}_v}\prod_{k=1}^{N_k}\pi_{vk}^{z_{vk}}\mathcal{N}\left(\lambda_v;\mu_{\lambda,k},\sigma_{\lambda,k}\right)^{z_{vk}}\mathcal{N}\left(y_v;\mu_{y,k},\sigma_{y,k}\right)^{z_{vk}} \tag{4.45}$$

$$= \frac{\partial}{\partial\lambda_r}\log\prod_{v}^{N_v}\sum_{k=1}^{N_k}\pi_{vk}\mathcal{N}\left(\lambda_v;\mu_{\lambda,k},\sigma_{\lambda,k}\right)\mathcal{N}\left(y_v;\mu_{y,k},\sigma_{y,k}\right) \tag{4.46}$$

$$= \frac{\partial}{\partial\lambda_r}\sum_{v=1}^{N_v}\log\sum_{k=1}^{N_k}\pi_{vk}\mathcal{N}\left(\lambda_v;\mu_{\lambda,k},\sigma_{\lambda,k}\right)\mathcal{N}\left(y_v;\mu_{y,k},\sigma_{y,k}\right) \tag{4.47}$$

$$= \frac{\partial}{\partial\lambda_r}\log\sum_{k=1}^{N_k}\pi_{rk}\mathcal{N}\left(\lambda_r;\mu_{\lambda,k},\sigma_{\lambda,k}\right)\mathcal{N}\left(y_r;\mu_{y,k},\sigma_{y,k}\right) \tag{4.48}$$

$$= \frac{\sum_{k=1}^{N_k}\frac{\mu_{\lambda,k}-\lambda_r}{\sigma^2_{\lambda,k}}\pi_{rk}\mathcal{N}\left(\lambda_r;\mu_{\lambda,k},\sigma_{\lambda,k}\right)\mathcal{N}\left(y_r;\mu_{y,k},\sigma_{y,k}\right)}{\sum_{k=1}^{N_k}\pi_{rk}\mathcal{N}\left(\lambda_r;\mu_{\lambda,k},\sigma_{\lambda,k}\right)\mathcal{N}\left(y_r;\mu_{y,k},\sigma_{y,k}\right)} \tag{4.49}$$

$$= \sum_{k=1}^{N_k}l_{rk}\frac{\mu_{\lambda,k}-\lambda_r}{{\sigma_{\lambda,k}}^2} \tag{4.50}$$

(note that of the N_v terms in (4.47), only the r-th term is non-zero, therefore the change of index from v to r between equations (4.47) and (4.48), where the sum disappears). In order to optimise with respect to λ one may use again a projected gradient ascent algorithm. It would be preferred, in order to maintain the positivity of the solution and for its simplicity and automatic line search feature, to adopt the EM algorithm for this local maximisation with respect to λ. Unfortunately, as discussed in 3.1.1, there is no closed form solution for the maximisation (M) step of the EM algorithm. In order to benefit from the advantages of the Expectation Maximisation algorithm, let us adopt, for the maximisation of (4.38), the One Step Late approximation, described in 3.2.1:

$$\lambda_v^{(n+1)} = \lambda_v^{(n)}\frac{1}{\sum_{d=1}^{N_d}a_{vd}-\sum_{k=1}^{N_k}l_{vk}\frac{\mu_{\lambda,k}-\lambda_v^{(n)}}{{\sigma_{\lambda,k}}^2}}\sum_{d=1}^{N_d}a_{vd}\frac{q_d}{\sum_{v'=1}^{N_v}a_{v'd}\lambda_{v'}^{(n)}} \tag{4.51}$$

Summary of the algorithm

The ICM algorithm consists of alternating two steps:

i Optimisation of the parameters of the mixture model (Expectation Maximisation):

$$l_{vk} \triangleq \frac{\pi_{vk}\mathcal{N}\left(\lambda_v;\mu_{\lambda,k},\sigma_{\lambda,k}\right)\mathcal{N}\left(y_v;\mu_{y,k},\sigma_{y,k}\right)}{\sum_{k=1}^{N_k}\pi_{vk}\mathcal{N}\left(\lambda_v;\mu_{\lambda,k},\sigma_{\lambda,k}\right)\mathcal{N}\left(y_v;\mu_{y,k},\sigma_{y,k}\right)} \tag{4.52}$$

$$\mu_{y,k}^{(n+1)} = \frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk} y_v \tag{4.53}$$

$$\sigma_{y,k}^{(n+1)} = \frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk} \left(\mu_{y,k}^{(n+1)} - y_v\right) \tag{4.54}$$

$$\mu_{\lambda,k}^{(n+1)} = \frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk} \lambda_v \tag{4.55}$$

$$\sigma_{\lambda,k}^{(n+1)} = \frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk} \left(\mu_{\lambda,k}^{(n+1)} - \lambda_v\right) \tag{4.56}$$

ii Optimisation of the spatial concentration of the radio-pharmaceutical (One-step-late Expectation Maximisation):

$$\lambda_v^{(n+1)} = \lambda_v^{(n)} \frac{1}{\sum_{d=1}^{N_d} a_{vd} - \frac{\partial}{\partial\lambda_v}\log p(\lambda|\mathbf{y},\theta,\pi)|_{\lambda=\lambda^{(n)}}} \sum_{d=1}^{N_d} a_{vd} \frac{q_d}{\sum_{v'=1}^{N_v} a_{v'd}\lambda_{v'}^{(n)}}, \tag{4.57}$$

$$\text{where: } \frac{\partial}{\partial\lambda_v}\log p(\lambda|\mathbf{y},\theta,\pi) = \sum_{k=1}^{N_k} l_{vk}\frac{\mu_{\lambda,k}-\lambda_v}{\sigma^2} \tag{4.58}$$

4.4 Context and magnetic field distortion

The model described in the previous paragraphs expresses the probability of the hidden state in voxel v only as a function of the MR intensity and pharmaceutical uptake in the same voxel. The model is made more robust by adding contextual information in the form of spatial dependence of the hidden labels. Not only $\mathbf{z}_v$ depends on y_v and λ_v, but on $\mathbf{z}_{v'}, v' \neq v$ elsewhere. For computational convenience, the spatial dependence of k is modelled with a joint distribution of the variables $\mathbf{z}_v$ which factorises on the first order neighbourhood structure of the image lattice (Markov Random Field - MRF): $\mathbf{z}_v$ is assumed to be conditionally independent from all other states if its 6 nearest neighbours are known. The dependence of state $\mathbf{z}_v$ upon its neighbours $\mathcal{N}_v$ is expressed by the following Potts model $p(\mathbf{z}_v|\mathcal{N}_v,\theta_\mathbf{z})$, parametrised by $\theta_\mathbf{z}$, which corresponds (due to the Markov-Gibbs equivalence [32]) to the joint density of the hidden states $p(k|\theta_\mathbf{z})$ (see also [71]):

$$p(\mathbf{z}_v|\mathcal{N}_v,\theta_\mathbf{z}) = \frac{e^{-\mathbf{z}_v^T\, G_{\theta_\mathbf{z}}\, \mathbf{g}_v}}{\sum_n^N e^{-\mathbf{z}_v^T\, G_{\theta_\mathbf{z}}\, \mathbf{g}_v}} \tag{4.59}$$

$$p(\mathbf{Z}|\theta_\mathbf{z}) = \frac{e^{-\sum_v^{N_v} -\mathbf{z}_v^T\, G_{\theta_\mathbf{z}}\, \mathbf{g}_v}}{R} \tag{4.60}$$

where $\mathbf{g}_v$ counts the labels of each class between the neighbours, $G_{\theta_\mathbf{z}}$ is a $[N \times N]$ matrix of parameters of the Potts model that expresses the affinity of all pairs of states and R is the partition function, not involved in the maximisation of the joint probability.

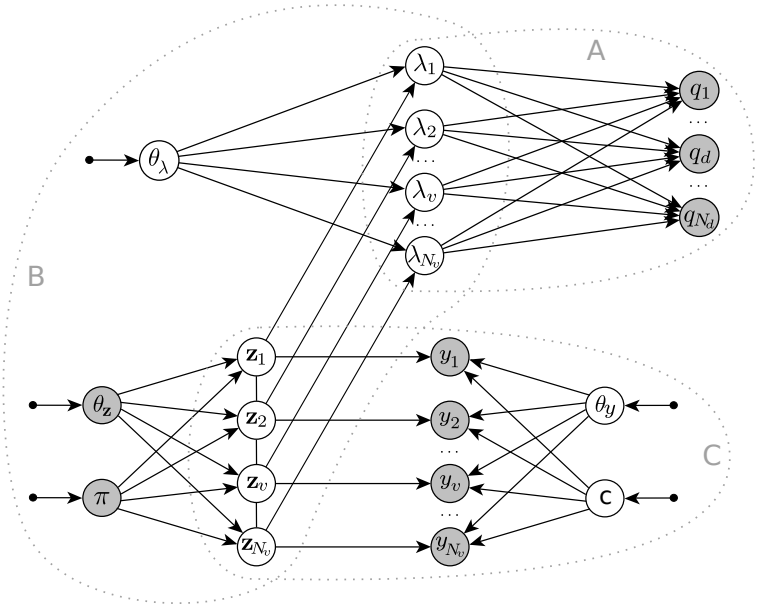


Figure 4.4: Contextual model and model of the magnetic field distortion: $\theta_\mathbf{z}$ parametrises the contextual model that relates spatially the discrete hidden variables and c parametrises a polynomial model of bias field inhomogeneity of the MR imaging system.

The intensity of the MR image is distorted due to the non-uniformity of the magnetic field. The smoothly varying bias due to the non-uniformity of the magnetic field is modelled by a linear combination of J polynomial basis functions $\phi_j(X)$, where X denotes the 3D coordinates. Intensity is log transformed, as suggested in [71], in order to treat the bias field, which multiplicates the intensity, as an additive term. From now on, therefore $\boldsymbol{y}$ indicates the log of the MR image intensity. The set of the J parameters of the polynomial model of the bias field is denoted by $c = \{c_1, \ldots, c_J\}$. Equation (4.14) is then substituted by:

$$p(y_v|z_{vk} = 1, \theta_y, c) = \mathcal{N}\left(y_v - \sum_{j=1}^{J} c_j \phi_j(X_v); \mu_{y,k}, \sigma_{y,k}\right) \tag{4.61}$$

where X_v are the coordinates of voxel v.

The joint optimisation is performed by alternating the optimisation of three subsets:

- $\mathcal{S}_1 : \theta = \{\theta_\lambda, \theta_y\}$
- $\mathcal{S}_2 : \boldsymbol{\lambda}$
- $\mathcal{S}_3 : c.$

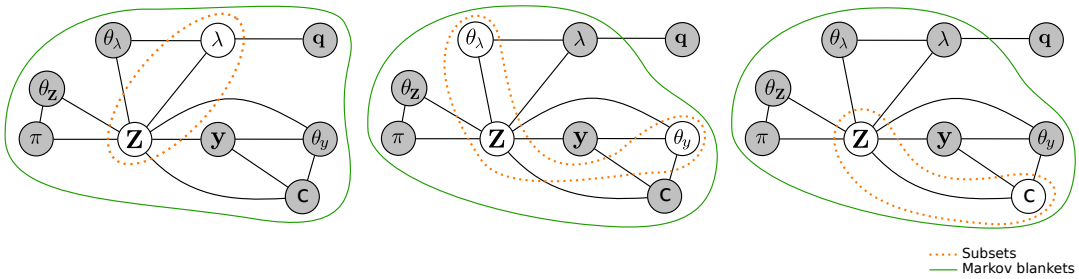


Figure 4.5: Moralised MRF for the joint model with the contextual model and the model of the magnetic field distortion.

The optimisation problems associated to $\mathcal{S}_1$ and $\mathcal{S}_3$ have been described and investigated in the context of image segmentation in [71]. The optimisation with respect to parameters c does not involve, as expressed by the Markov blanket of Fig. 4.5-right, the photon counts $\boldsymbol{q}$, therefore it only involves the bi-variate mixture. Such optimisation problem has been described, in the context of image segmentation, in [71], which proposes to apply the Mean Field Approximation (MFA), yielding the following update formulae:

i Update the parameters θ:

$$l_{vk} \triangleq \frac{\pi_{vk} p(y_v|z_{vk};\theta_{y,k}^{(n)}) p(\lambda_v|z_{vk};\theta_{\lambda,k}^{(n)}) p(z_{vk}=1|l_{\mathcal{N}_v}^{(n)};\theta_{\mathbf{z}})}{\sum_{k=1}^{N_k} \pi_{vk} p(y_v|z_{vk};\theta_{y,k}^{(n)}) p(\lambda_v|z_{vk};\theta_{\lambda,k}^{(n)}) p(z_{vk}=1|l_{\mathcal{N}_v}^{(n)};\theta_{\mathbf{z}})} \tag{4.62}$$

where the spatial dependence term $p(z_{vk}|l_{\mathcal{N}_v}^{(n)};\theta_{\mathbf{z}})$ is approximated from the previous estimate of l_{vk} by the MFA [71]:

$$p(z_{vk}=1|l_{\mathcal{N}_v}^{(n)};\theta_{\mathbf{z}}) = \frac{e^{-U_{\theta_{\mathbf{z}}}(e_k|l_{\mathcal{N}_v}^{(n)})}}{\sum_{k'=1}^{N_k} e^{-U\theta_{\mathbf{z}}(e_{n'}|l_{\mathcal{N}_v}^{(n)})}} \qquad U_{\theta_{\mathbf{z}}}(e_k|l_{\mathcal{N}_v}^{(n)}) = e_k^T \, G_{\theta_{\mathbf{z}}} \, \mathbf{g}_v^{(n)} \tag{4.63}$$

where e_k is the unit vector with k-th element equal to 1 and $\mathbf{g}_v^{(n)}$ is the vector of length N with elements given by:

$$g_{vn}^{(n)} = \sum_{\mathcal{N}_v} l_{vk}^{(n)} \tag{4.64}$$

$$\mu_{y,k}^{(n+1)} = \frac{1}{N_v} \frac{\sum_{v=1}^{N_v} p_{vk}^{(n+1)} y_v - \sum_{j=1}^{J} c_j \phi_j(X_v)}{\pi_{vk}} \tag{4.65}$$

$$\sigma_{y,k}^{2^{(n+1)}} = \frac{1}{N_v} \frac{\sum_{v=1}^{N_v} p_{vk}^{(n+1)} (\mu_{y,k}^{(n+1)} - y_v + \sum_{j=1}^{J} c_j \phi_j(X_v))^2}{\pi_{vk}} \tag{4.66}$$

$$\mu_{\lambda,k}^{(n+1)} = \frac{1}{N_v} \frac{\sum_{v=1}^{N_v} p_{vk}^{(n+1)} \lambda_v}{\pi_{vk}} \tag{4.67}$$

$$\sigma_{\lambda,k}^{2^{(n+1)}} = \frac{1}{N_v} \frac{\sum_{v=1}^{N_v} p_{vk}^{(n+1)} (\mu_{\lambda,k}^{(n+1)} - \lambda_v)^2}{\pi_{vk}}. \tag{4.68}$$

ii Update the bias field parameters:

$$\begin{bmatrix} c_1^{(n+1)} \\ c_2^{(n+1)} \\ \dots \end{bmatrix} = \left(A^T \, diag \, w_v^{(n+1)} \, A \right)^{-1} A^T \, diag \, w_v^{(n+1)} \begin{bmatrix} y_1 - \tilde{y}_1^{(n+1)} \\ y_2 - \tilde{y}_2^{(n+1)} \\ \dots \end{bmatrix}, \tag{4.69}$$

where A is the geometrical matrix of the bias field model, each of its columns evaluating the polynomial basis function ϕ_j at voxel coordinates X_v and

$$w_v^{(n+1)} = \sum_{k=1}^{N_k} w_{vk}^{(n+1)} \qquad w_{vk}^{(n+1)} = \frac{p_{vk}^{(n+1)}}{\sigma_{y,k}^{2^{(n+1)}}} \qquad \tilde{y}_v^{(n+1)} = \frac{\sum_{k=1}^{N_k} w_{vk}^{(n+1)} \mu_{y,k}^{(n+1)}}{\sum_{k=1}^{N_k} w_{vk}^{(n+1)}}.$$

iii Update the estimate of the pharmaceutical density:

Under the MFA, the expectation of the hidden variables at step n is expressed by (4.62). Therefore, under the MFA, the update formula for the subset $\mathcal{S}_2$ described in the previous paragraph is affected in that (4.33) is replaced by (4.62):

$$\lambda_v^{(n+1)} = \lambda_v^{(n)} \frac{1}{\sum_{d=1}^{N_d} a_{vd} - \frac{\partial}{\partial \lambda_v} \log p(\lambda|\mathbf{y}, \theta, \pi)|_{\lambda=\lambda^{(n)}}} \sum_{d=1}^{N_d} a_{vd} \frac{q_d}{\sum_{v'=1}^{N_v} a_{v'd} \lambda_{v'}^{(n)}} \tag{4.70}$$

$$\frac{\partial}{\partial \lambda_v} \log p(\lambda|\mathbf{y}, \theta, \pi) = \sum_{k=1}^{N_k} l_{vk} \frac{\mu_{\lambda,k} - \lambda_v}{\sigma^2} \tag{4.71}$$

4.5 Experiments

Uptake of FDG was simulated by assigning typical average values observed in PET FDG scans to the BrainWeb (http://mouldy. bic.mni.mcgill.ca/brainweb/) ground truth tissue model: 4 parts uptake in the gray matter (GM), 1 part in white matter (WM), 0 part in the cerebral-spinal fluid (CSF). One hot and one cold spherical lesion of $14\ mm$ diameter were simulated by augmenting the activity by 30% in the hot lesion and reducing it by 50% in the cold lesion. The PET scan was simulated with the PET-Sorteo [59] simulator.

For reconstruction, the number of classes N_k was set to 6: White Matter, Internal Gray Matter, External Gray Matter, External CSF, Internal CSF, everything else. The statistical atlas for the 6 classes was obtained from the publicly available MNI-152 brain atlas [72] by splitting manually internal and external regions for the Gray Matter and CSF classes. The off-diagonal elements of the $N \times N$ parameter matrix $\theta_{\mathbf{z}}$ were set to 0.1 and the diagonal elements to 0, penalising equally all transitions except for the transition to the identical class (more sophisticated parameter selection criteria are reported in [70]). The order of the polynomial for bias field correction J was set to 4. Scatter was accounted for simply by discarding the photons with detected energy smaller than $400\ KeV$.

Ancestral sampling

A visualisation of the type of assumptions encoded by a probabilistic model can be obtained by sampling from the joint probability distribution. In order to draw random samples from the joint probability distribution of $\mathbf{Z}, \boldsymbol{\lambda}, \boldsymbol{y}, \boldsymbol{q}$, we have first obtained a reasonable value of the parameters $\theta_\lambda, \theta_y, c$ by performing MAP estimation on a single realisation of the MR image and of the emission data, running the algorithm for 10000 iterations. Given the parameters, samples of $\mathbf{Z}, \boldsymbol{\lambda}, \boldsymbol{y}, \boldsymbol{q}$ were drawn with the *ancestral sampling* technique. As explained in [13]- p.365, ancestral sampling is a simple technique to sample from the joint probability distribution encoded by a directed acyclic graph. Ancestral sampling consists in assigning an ordering to the variables such that there are no links from any node to any lower numbered node. In order to sample from the joint probability distribution of all the nodes, one starts by sampling from the lowered numbered node, then draws a sample from all other nodes, following the increasing nodes index. Each sample is drawn according to the conditional probability distribution of the variables associated to the node, conditional to the value obtained by sampling the lower numbered node (parent node). Referring to Fig. 4.2, the indexes assigned to the nodes are $\mathbf{Z} : 1, \boldsymbol{y} : 2, \boldsymbol{\lambda} : 3, \boldsymbol{q} : 4$. In order to simplify the sampling algorithm, the spatial dependence of the hidden variables was disregarded. Consequently, the set of the hidden discrete variables in all voxels is obtained by sampling each state independently, for every voxel v, from the spatially varying multinomial distribution π_v (the statistical atlas).

10000 samples were drawn from the joint probability distribution. Fig. 4.6 reports a visualisation of the samples drawn from the joint distribution of $\mathbf{Z}, \boldsymbol{\lambda}, \boldsymbol{y}, \boldsymbol{q}$. The first row visualises a single sample of, from left to right, $\mathbf{Z}, \boldsymbol{\lambda}, \boldsymbol{y}, \boldsymbol{q}$. The second row visualises the mean over all samples and the third row

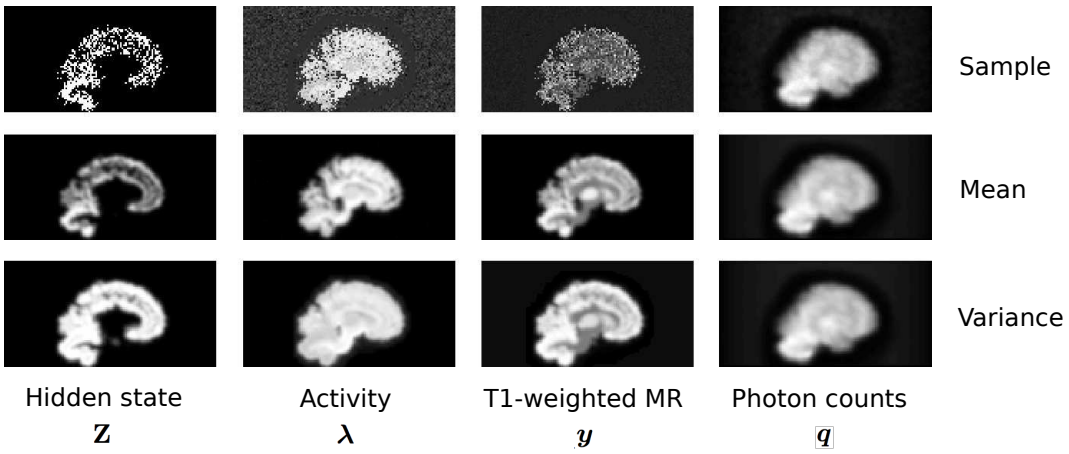


Figure 4.6: Ancestral sampling, 10000 samples. From top to bottom: random sample, mean, variance. From left to right: hidden discrete label (white if state is 1, black for all other states), activity, T1-weighted MR image, photon counts. The mean and variance images express the type of measurements that are expected according to the model.

the variance calculated over all samples. The images of $\boldsymbol{\lambda}, \boldsymbol{y}$ report simply a slice of the imaging volume (central sagittal slice) with grey scale colour coding, while the images of $\boldsymbol{q}$ on the right are grey scale viewgrams of the emission data. Since grey scale is not effective at representing the discrete states, only the first state is visualised in the images on the left. Specifically, the image of the single sample reports $\sum_v^{N_v} \mathbf{z_v} e_1$, with e_1 being the unit vector of length N with 1 in the first position. The mean is obtained by averaging such images over all samples and the variance accordingly. The mean image of the photon counts (right column, central line of Fig. 4.6) expresses the photon counts that one expects to observe on average. Notice that this becomes a uniform distribution if one discards the finite mixture model.

Bias variance analysis

The bias/variance plot provides some insight into the properties of the model. 10240 instances of the measurement were simulated by applying Poisson noise to the noiseless measurement obtained with the PET Sorteo Monte Carlo simulator. The total number of counts was set to 150×10^6 photons (although the exact number of counts of each measurement is a realisation of the Poisson distribution centred in 150×10^6). The reconstruction algorithm was executed for 5000 iterations for each measurement. Fig. 4.7-green displays the bias/variance norm as a function of the number of iterations. The curve is obtained by alternating the optimisation steps starting with a uniform estimate of the activity.

Fig. 4.10 displays the mean of the 10240 reconstructions, the variance (diagonal of the covariance matrix, reshaped as an image), the bias (3.35) and a single reconstruction. Consistently with the observations of the previous chapter, the average of a large number of reconstructions accurately represents the original image, if one iterates until convergence. Fig. 4.11 reports, for comparison, the images obtained

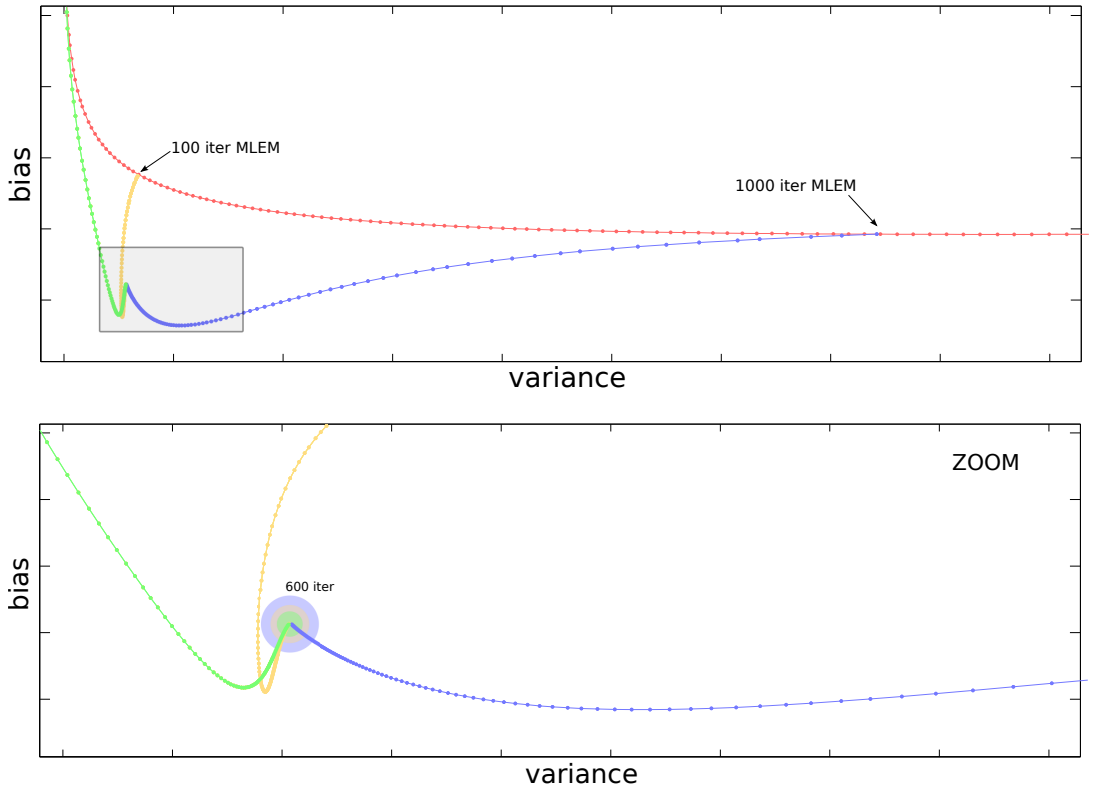


Figure 4.7: Bias/variance norm (for the whole image) as a function of the number of iterations (see equations (3.59) and (3.60)). Red: MLEM; green, yellow and blue: finite mixture model with different ICM schemes (see Paragraph 4.5 - 'Effect of the ICM iteration scheme'). The three ICM chemes converge to the same point in the Bias/Variance plane, indicating that different ICM schemes provide the same results. However, the zoom-in (bottom) shows that the green scheme (see Paragraph 4.5 - 'Effect of the ICM iteration scheme') requires less iterations to approach the point of convergence on the Bias/Variance plane.

with the smoothing prior with smoothing parameter β_3, as described in the experiments of chapter 3 (Fig. 3.15). The single reconstruction has a somewhat similar appearance to the reconstruction with the smoothing prior, though the mean of the estimator closely resembles the phantom. After 1000 iterations the reconstruction does not present visible differences.

Effect of the ICM iteration scheme

The ICM algorithm suggests to iterate until convergence each sub-optimisation step. In order to reduce the computational complexity, one may consider to perform only one or a few steps of each sub-optimisation. Different iteration schemes may be devised, by selecting different number of iterations for each sub-optimisation problem. Does the algorithm converge to the same solution with different iteration schemes? The green, yellow and blue curves of Fig. 4.7 report three experiments that point out that the algorithm converges to the same solution with different iteration schemes. The green curve of

Fig. 4.7 reports the bias variance norm as a function of the number of iterations obtained by alternating one step of OSL-MAPEM for the activity, one step of EM for the parameters and one step of the gradient ascent algorithm for the bias field correction algorithm. The yellow and blue curves were obtained by performing initially 100 and 1000 iterations of OSL-MAPEM and then starting to alternate one iteration of OSL-MAPEM and, one of EM and one of the gradient ascent algorithm for the bias field model. The convergence to a common point suggests that different choices of iteration scheme produce the same results at convergence. The zoom-in in Fig. 4.7-bottom reports a circle, for each of the three iteration schemes, centered at the common point of convergence on the bias/variance plane and whose circumference intersects the bias/variance curve at 600 iterations. The algorithm that alternates a single iteration of each optimisation since the start presents the smallest radius, requires less iterations to approach the solution. Convergence is not guaranteed due to the non-convexity of the model, however it is reassuring to observe that different alternation schemes provide the same solution, indicating that in the algorithms in the experiment select the same local maximum.

Lesion delineation

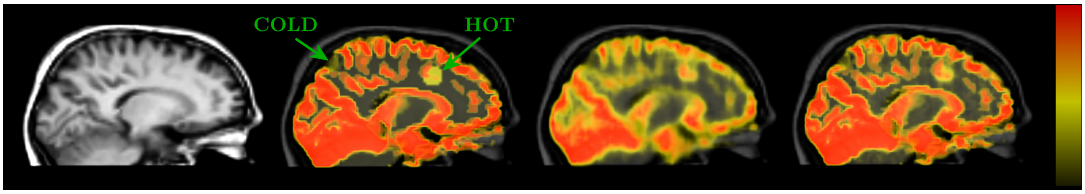
Fig. 4.8 reports a sagittal slice of the reconstruction obtained for one of the noise instances. The table in Fig. 4.8 reports the coefficient of recovery (COR) and signal to noise ratio (SNR) of the uptake in 7 regions of interest including the two lesions. The COR is the ratio of the estimated to true uptake in each region-of-interest (ROI):

$$\text{COR}_{\text{ROI}} = \frac{\sum_{v \in \text{ROI}} \hat{\lambda}_v}{\sum_{v \in \text{ROI}} \lambda_v}, \tag{4.72}$$

where $\hat{\lambda}_v$ is the estimated activity in voxel v and λ_v the true activity in voxel v. The SNR is the ratio between the standard deviation of the activity values in the voxels in a region of interest of the reconstructed image and the mean of the reconstructed activity in the region of interest:

$$\text{SNR}_{\text{ROI}} = \frac{\sqrt{\frac{1}{N}\sum_{v \in \text{ROI}} \left(\hat{\lambda}_v - \frac{1}{N}\sum_{v' \in \text{ROI}} \hat{\lambda}_{v'}\right)^2}}{\frac{1}{N}\sum_{v \in \text{ROI}} \hat{\lambda}_v} \tag{4.73}$$

The results highlight that the pharmaceutical uptake model yields measurements with increased recovery and less noise when the uptake is consistent with the model. It is also remarkable that the lesions are well reconstructed even though they are outliers of the model (there is not a class that captures the lesions), presenting overall improved SNR. This fact can be explained with the strong inter-correlation of the unknowns in emission imaging due to the line-integral measurements: improving the estimate everywhere outside of the lesion already improves the estimate in the lesion.



		White Matter	Internal Gray Matter	External Gray Matter	External CSF	Internal CSF	Hot Lesion	Cold Lesion
MLEM	COR	0.53	0.84	0.87	0.42	0.50	0.14	0.21
	SNR	10.21	7.78	17.67	7.59	3.65	1.13	1.01
UPTAKE MODEL	COR	0.86	0.95	0.95	0.81	0.84	0.14	0.20
	SNR	11.98	10.85	30.00	11.65	2.10	1.58	1.01

Figure 4.8: Top: Synthetic PET-MR FDG imaging data (see-through volume-rendering of 5 mm thick sagittal slices); from left to right: T1-weighted MR image, activity phantom, MLEM reconstruction, reconstruction with the proposed pharmaceutical uptake model using 6 classes from the MNI-152 brain statistical atlas [72]. Bottom: coefficient-of-recovery (COR) and signal-to-noise ratio (SNR) for 7 regions of interest (see equations (4.72) and (4.73)).

Clinical data

The finite mixture model was applied to clinical PET FDG using the same 6 classes that were employed in the synthetic PET FDG study. The emission data and the MR images were acquired on separate machines: the pharmaceutical density was estimated initially with MLEM, aligned with the MR image by rigid registration with Normalised Mutual Information cost function and reconstructed again with the uptake model. Reconstructions are reported in Fig. 4.9.

Conclusion

This chapter has presented a model for the integration of information from multiple imaging modalities in the context of PET-MR, based on the assumption of finite discrete states. Robust estimation is achieved by 1) adopting a population-based statistical atlas to initialise and drive the optimisation of the parameters of the pharmaceutical uptake model and of the MR acquisition system model; 2) adding contextual information in the form of a Markov Random Field over the hidden tissue labels; 3) capturing the MR image bias field.

The method has been evaluated with a synthetic study, showing that it improves the quantification of the pharmaceutical uptake when the simulated data reflects the assumptions of the model and, remarkably, that the uptake estimate may improve also in regions that do not obey to the model. Validation with real data remains an open problem as it would require large sets of imaging data, possibly labelled with long term clinical outcome for specific imaging tasks. To this extent, automation of the reconstruction algorithm is crucial.

The population-derived information assigns a meaning to the discrete hidden variables. The con-

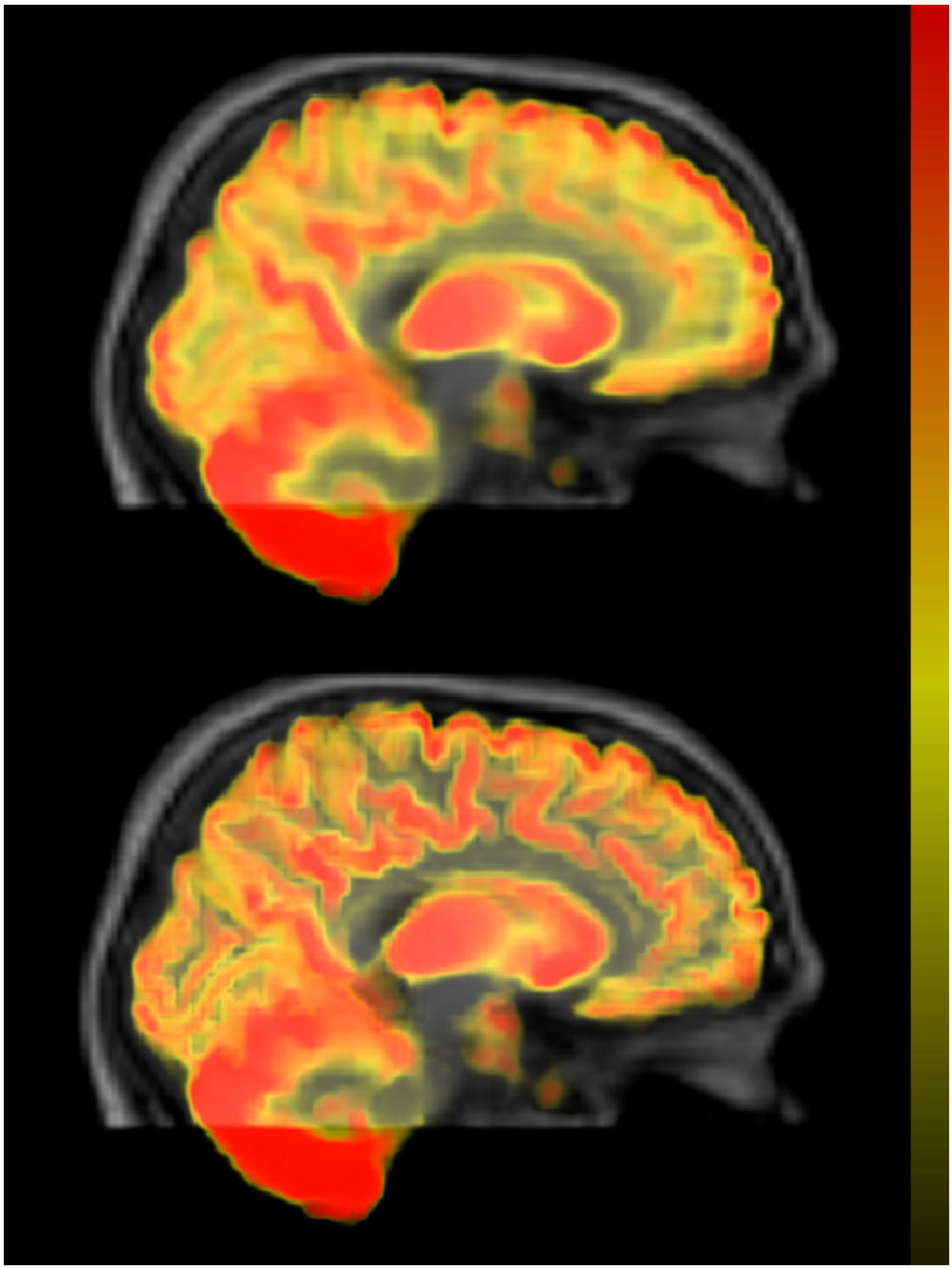


Figure 4.9: PET FDG reconstructions obtained with MLEM (top) and with the proposed pharmaceutical uptake model (bottom) using 6 classes from the MNI-152 brain statistical atlas [72]. See-through volume-rendering of 5 mm-thick sagittal slices.

struction of the population-derived statistical atlas remains an open problem. Perhaps the same model can be used to estimate the probability of the hidden states from a large number of observations)

Given the general acceptance of probabilistic atlas based segmentation algorithm [5, 73], we believe that a joint generative model for PET/MRI might prove useful when coupled with population based

spatially varying priors of the hidden states and eventually with hyperpriors for the parameters of the mixture model in order to include experience in the reconstruction process. The statistical atlas plays an essential role in the algorithm that has been described in this chapter, assigning a meaning to the abstract hidden states of the model. Due to the non-convexity of the optimisation problem, the atlas functions as initialisation, therefore assigning the meaning of the hidden states. In the experiments presented in paragraph 4.5 and 4.5, the statistical atlas was the MNI-152 brain atlas [72], created in the context of brain segmentation. Such atlas encodes the probability of white matter, internal gray matter, external gray matter, external CSF and internal CSF in each location of the image and is therefore suitable for perfusion studies. While the model is unlikely to be effective, in combination with the MNI-152 atlas, in case of receptor/radiolegand studies; we believe that the construction of drug-specific statistical atlases may improve the quantification in receptor/radiolegand studies.

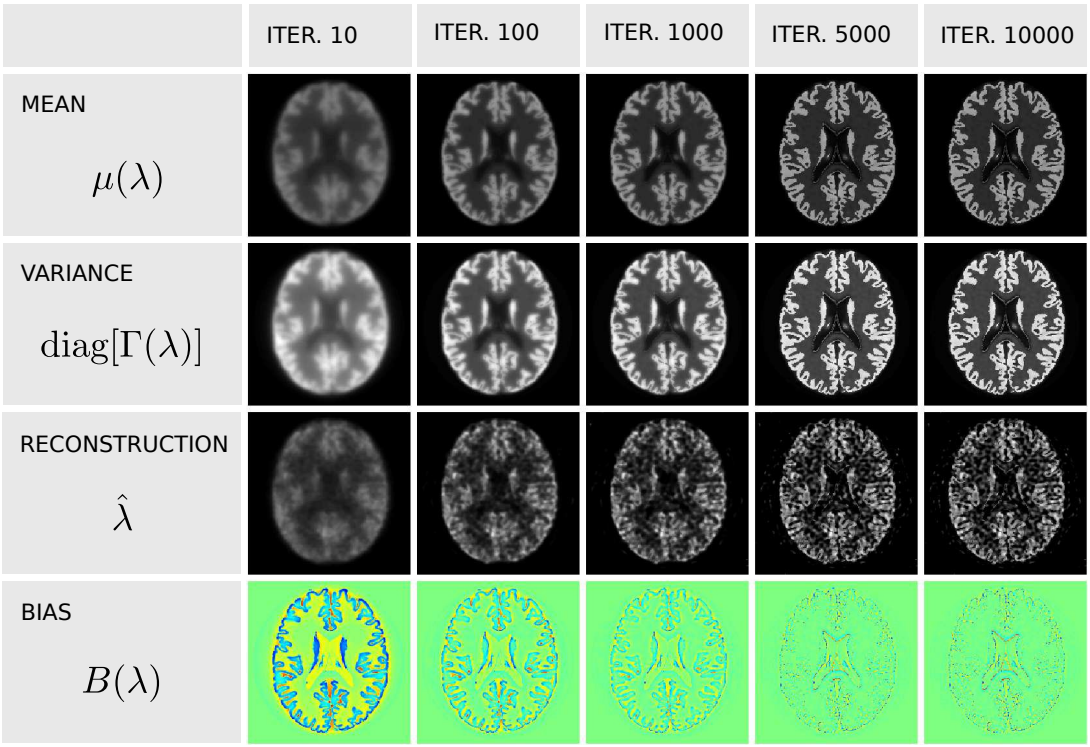


Figure 4.10: Finite anatomical-functional states: $N = 10240$ instances, 10 to 10000 iterations.

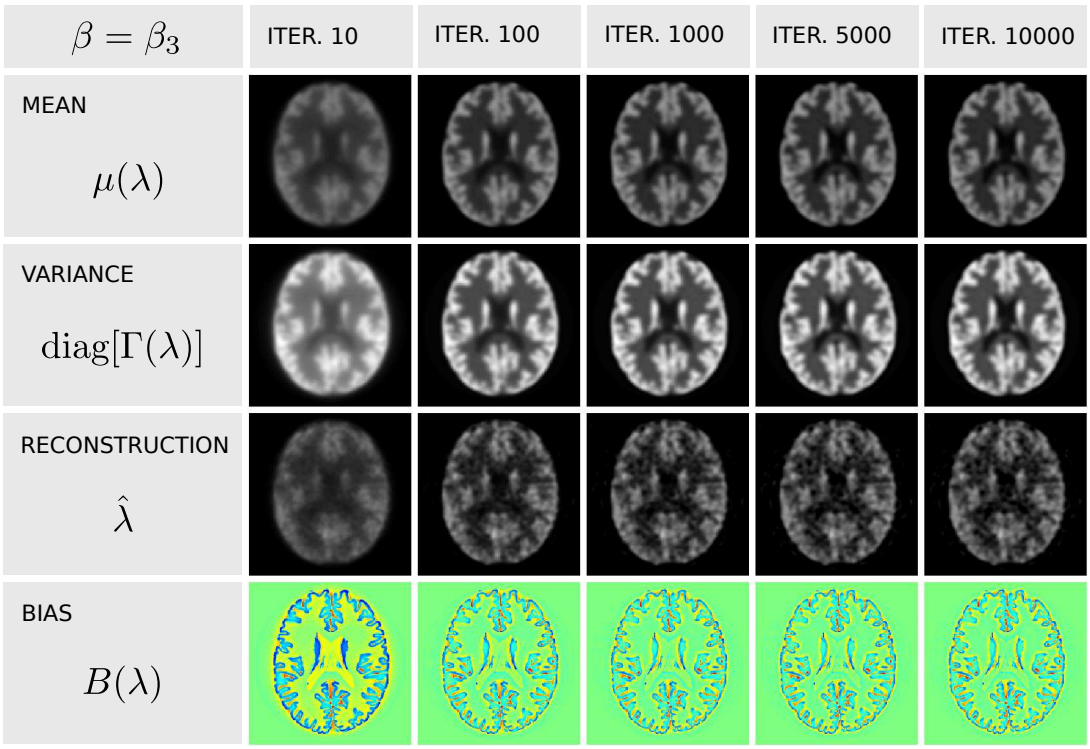


Figure 4.11: MAP-EM smoothing: $\beta = \beta_3$, $N = 10240$ instances, 10 to 10000 iterations.

4.A *The expectation maximisation algorithm for finite mixtures*

4.A.1 Finite mixture as a hidden variable model

A common imaging task is that of determining the state of a set of discrete variables related to continuous measurements. The assumption that there exist hidden discrete states may have, in certain applications, a precise physical meaning, or it may express a more vague understanding that the process that generates the imaging data is eventually characterised by the existence of discrete states. Under the generative approach, one expresses the conditional probability distribution of the imaging data given the hidden variables. Such conditional probability distribution expresses the characteristics of the imaging system. A common approach is to adopt parametric functions for such conditional probability distributions, in order to learn the characteristics of the imaging system from the imaging data. Estimation of the most likely parameters of the imaging system entails the marginalisation over the discrete hidden variables, calling for the EM algorithm, as discussed in 2.1.9.

In MR imaging, finite mixture models are widely used for the purpose of dividing the imaging domain in regions characterised by tissue with similar magnetic properties. In the context of MR imaging, one assumes that the image intensity y_v in voxel v is the realisation of a random process characterised by the existence of a (hidden) discrete label $\mathbf{z}_v$. Let us, once again, describe the discrete label with the 1-of-N_k representation:

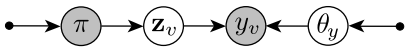


Figure 4.12: Finite mixture as a hidden variable model. The probability distribution associated to variable y_v depends on the hidden variable $\mathbf{z}_v$ and on the model parameters θ. The hidden variable $\mathbf{z}_v$ has prior probability parametrised by π. If one knew the state of $\mathbf{z}_v$, the state of knowledge about y_v would not be affected by π. Each pixel of an image may be considered a realisation of y_v (Fig. 4.13).

$$\mathbf{z}_v = [z_{v1}, z_{v2}, \ldots, z_{vk}, \ldots, z_{vN_k}] \tag{4.74}$$

$$z_{vk} \in \{0, 1\} \tag{4.75}$$

$$\sum_{k=1}^{N_k} z_{vk} = 1 \tag{4.76}$$

Assuming a prior probability distribution over the hidden state in v, the natural parametrisation for the discrete variable is the multinomial distribution.

$$\pi_v = [\pi_{v1}, \pi_{v2}, \ldots, \pi_{vk}, \ldots, \pi_{vN_k}] \tag{4.77}$$

$$\sum_{k=1}^{N_k} \pi_{vk} = 1 \tag{4.78}$$

$$p(\mathbf{z}_v|\pi) = \prod_{k=1}^{N_k} \pi_{vk}^{z_{vk}} \tag{4.79}$$

Let the conditional probability distribution $p(y_v|z_{vk}=1,\theta_y)$ be a parametric function Φ_k parametrised by θ_y:

$$p(y_v|z_{vk}=1,\theta_y)=\Phi_k\left(y_v;\theta_y\right) \tag{4.80}$$

Due to the 1-of-N_k representation of $\mathbf{z}_v$, this expression can be equivalently written as:

$$p(y_v|\mathbf{z}_v,\theta_y)=\prod_{k=1}^{N_k}\Phi_k\left(y_v;\theta_y\right)^{z_{vk}} \tag{4.81}$$

Fig. 4.12 expresses such a model. Applying Bayes rule:

$$p(y_v,\mathbf{z}_v|\pi_v,\theta_y)=p(\mathbf{z}_v|\pi_v,\theta_y)p(y_v|\mathbf{z}_v,\pi_v,\theta_y)=p(\mathbf{z}_v|\pi_v)p(y_v|\mathbf{z}_v,\theta_y) \tag{4.82}$$

and summing over all possible values of $\mathbf{z}_v$ gives:

$$p(y_v|\pi_v,\theta_y)=\sum_{\mathbf{z}_v}p(\mathbf{z}_v|\pi_v)p(y_v|\mathbf{z}_v,\theta_y)=\sum_{k=1}^{N_k}\pi_{vk}\Phi_k\left(y_v;\theta_y\right) \tag{4.83}$$

The probability distribution of y_v is a finite sum of (weighted) distributions, from which the name finite mixture model. For $\Phi_k(y_v,\theta)=\mathcal{N}\left(y_v,\mu_{y,k},\sigma_{y,k}\right)$ (normal distribution), the model is a finite mixture of Gaussians.

4.A.2 Estimation of the model parameters from multiple observations

Assuming that there exists one hidden variable for every voxel of the image, the graph in Figure 4.13 represents the relation between the observed image, the hidden variables and the parameters of the imaging system, with

$$\mathbf{Z}=[\mathbf{z}_1,\mathbf{z}_2,\ldots,\mathbf{z}_v,\ldots,\mathbf{z}_{N_v}] \qquad [N_v\times N_k] \tag{4.84}$$

$$\mathbf{y}=[y_1,y_2,\ldots,y_v,\ldots,y_{N_v}] \qquad [N_v\times 1] \tag{4.85}$$

$$\pi=[\pi_1,\pi_2,\ldots,\pi_v,\ldots,\pi_{N_v}] \qquad [N_v\times N_k] \tag{4.86}$$

One may pose the problem of inferring the most probable value of the model parameters given the

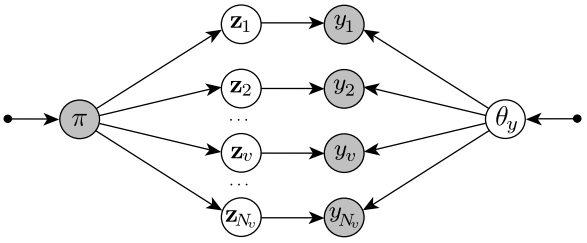


Figure 4.13: Each pixel of an image may be considered a realisation of the model of Fig. 4.12. The problem of estimating the parameters of the imaging model θ_y has the form of an incomplete data problem (see 2.1.9).

observed image $\mathbf{y}$:

$$\hat{\theta_y} = \underset{\theta_y}{\operatorname{argmax}}\, p(\theta_y|\mathbf{y}, \pi) = \underset{\theta_y}{\operatorname{argmax}}\, p(\mathbf{y}|\pi, \theta_y)p(\theta_y) \tag{4.87}$$

From the graph in Fig. 4.13 we get:

$$p(\mathbf{y}|\pi, \theta_y) = \prod_{v=1}^{N_v} p(y_v|\pi_v, \theta_y) \tag{4.88}$$

The $\log$ posterior probability distribution of the model parameters is differentiable with respect to the model parameters as long as the prior probability distribution $p(\theta_y)$ and the observation model $\Phi_k\left(y_v, \theta_y\right)$ are differentiable. This gives:

$$\log p(\theta_y|\mathbf{y}, \pi) = \log p(\theta_y) + \sum_{v=1}^{N_v} \log \left[\sum_{k=1}^{N_k} \pi_k \Phi\left(y_v; \theta_y\right)\right] \tag{4.89}$$

However, the hidden variables suggest to apply the EM algorithm to device an optimisation algorithm for (4.87).

4.A.3 The expectation maximisation algorithm

Considering $\mathbf{Z}$ the *hidden variables* and $\mathbf{y}$ the *observed variables*, the problem of estimating θ may be considered as an *incomplete data* problem (see 2.1.9), regarding the set $\{\mathbf{Z}, \mathbf{y}\}$ as the *complete data*:

$$p(\mathbf{y}, \mathbf{Z}|\pi, \theta_y) = \prod_{v=1}^{N_v} \prod_{k=1}^{N_k} {\pi_{vk}}^{z_{vk}} \mathcal{N}\left(y_v; \mu_{y,k}, \sigma_{y,k}\right)^{z_{vk}} \tag{4.90}$$

The $\log$ posterior of the complete data is:

$$\log p(\mathbf{y}, \mathbf{Z}|\pi, \theta_y) = \sum_{v=1}^{N_v} \sum_{k=1}^{N_k} z_{vk} \left[\log \pi_k + \log \mathcal{N}\left(y_v; \mu_{y,k}, \sigma_{y,k}\right)\right] \tag{4.91}$$

From the graph in Figure 4.13, the posterior probability distribution of the hidden variables conditional to the model parameters is:

$$p(\mathbf{Z}|\mathbf{y}, \pi, \theta_y) \propto \prod_{v=1}^{N_v} \prod_{k=1}^{N_k} \left[\pi_k \mathcal{N}\left(y_v, \mu_{y,k}, \sigma_{y,k}\right)\right]^{z_{vk}} \tag{4.92}$$

The expected value of the indicator variable z_{vk} under this probability distribution is:

$$\mathbb{E}_{\mathbf{Z}}\left[z_{vk}\right] = \frac{\sum_{z_{vk}} z_{vk} \left[\pi_k \Phi_k(y_v; \theta_y)\right]^{z_{vk}}}{\sum_{z_{vk'}} \left[\pi_{k'} \Phi_{k'}(y_v; \theta_y)\right]^{z_{vk'}}} = \frac{\pi_k \Phi_k(y_v; \theta_y)}{\sum_{k'=1}^{N_k} \pi_{k'} \Phi_{k'}(y_v; \theta_y)} \triangleq l_{vk} \tag{4.93}$$

All ingredients to apply the EM recipe of 2.1.9 are ready. The expectation of the $\log$ posterior of the complete data (4.91) with respect to the posterior distribution of the hidden variables conditional on the provisional estimate of the model parameters (4.92) is obtained by simply replacing z_{vk} in (4.91) with its expectation, as the expression in the square brackets does not depend on z_{vk}. This gives:

$$\mathbb{E}_{\mathbf{Z}}\left[\log p(\mathbf{y}, \mathbf{Z}|\pi, \theta_y)\right] = \sum_{v=1}^{N_v} \sum_{k=1}^{N_k} l_{vk} \left[\log \pi_k + \log \Phi_k\left(y_v; \theta_y\right)\right] \tag{4.94}$$

The maximum occurs when the derivatives are zero. With Gaussian functions $\Phi_k(y_v, \theta) = \mathcal{N}(y_v, \mu_{y,k}, \sigma_{y,k})$, differentiating with respect to $\mu_{y,k}$ and $\sigma_{y,k}$ in turn, and setting the derivatives to zero, the maximisers are expressed in closed form:

$$\mu_{y,k}^{(n+1)} = \frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk} y_v \tag{4.95}$$

$$\sigma_{y,k}^{(n+1)} = \left[\frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk} \left(\mu_{y,k}^{(n+1)} - y_v \right) \right]^{\frac{1}{2}} \tag{4.96}$$

$$l_{vk} = \frac{\pi_k \mathcal{N}(y_v; \mu_{y,k}, \sigma_{y,k})}{\sum_{k'=1}^{N_k} \pi_{k'} \mathcal{N}(y_v; \mu_{y,k'}, \sigma_{y,k'})} \tag{4.97}$$

4.A.4 Naive multi-variate mixture

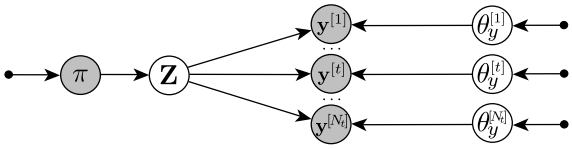


Figure 4.14: Naive multi-variate mixture model: the multiple observations would be independent if the hidden variables were known. Notice that the naive mixture model corresponds to the assumption that the hidden discrete variables $\mathbf{Z}$ are the only common cause underlying the multiple observations.

Let us consider the case of Fig. 4.14, where multiple images $\mathbf{y}^{[1]}, \ldots, \mathbf{y}^{[t]}, \ldots, \mathbf{y}^{[N_t]}$ are considered to be the related to common hidden discrete states. If each image is considered to be the realisation of a hidden process with its own parameters $\theta_{y^{[t]}}$ (as is the case for $\boldsymbol{\lambda}$ and $\boldsymbol{y}$ in the graph of Fig. 4.1), then, as one may easily verify, the EM update equations have the following expression for all values of t:

$$\mu_{y,k}^{[t](n+1)} = \frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk} y_v^{[t]} \tag{4.98}$$

$$\sigma_{y,k}^{[t](n+1)} = \left[\frac{1}{\sum_{v=1}^{N_v} l_{vk}} \sum_{v=1}^{N_v} l_{vk} \left(\mu_{y,k}^{[t](n+1)} - y_v^{[t]} \right) \right]^{\frac{1}{2}} \tag{4.99}$$

$$l_{vk} = \frac{\pi_k \prod_{t=1}^{N_t} \mathcal{N}\left(y_v^{[t]}; \mu_{y,k}^{[t](n)}, \sigma_{y,k}^{[t](n)} \right)}{\sum_{k'=1}^{N_k} \pi_{k'} \prod_{t=1}^{N_t} \mathcal{N}\left(y_v^{[t]}; \mu_{y,k'}^{[t](n)}, \sigma_{y,k'}^{[t](n)} \right)} \tag{4.100}$$

These are the formulae utilised in paragraph 4.3.

If the parameter θ_y is common to all observations, as in Fig. 4.15, then:

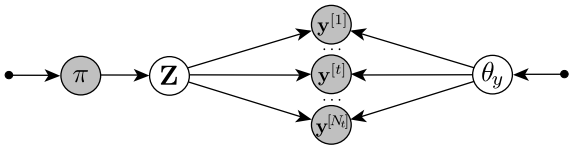


Figure 4.15: Naive multi-variate mixture model with common parameter.

$$\mu_{y,k}^{(n+1)} = \frac{1}{N_t \sum_{v=1}^{N_v} l_{v,k}^{(n+1)}} \sum_{t=1}^{N_t} \sum_{v=1}^{N_v} l_{v,k}^{(n+1)} \boldsymbol{y}^{[t]} \left(T_{\gamma^{[t]}} X_v \right) \tag{4.101}$$

$$\sigma_{y,k}^{(n+1)} = \left[\frac{1}{N_t \sum_{v=1}^{N_v} l_{v,k}^{(n+1)}} \sum_{t=1}^{N_t} \sum_{v=1}^{N_v} l_{v,k}^{(n+1)} \left(\mu_{y,k}^{(n+1)} - \boldsymbol{y}^{[t]} \left(T_{\gamma^{[t]}} X_v \right) \right)^2 \right]^{\frac{1}{2}} \tag{4.102}$$

$$l_{vk} = \frac{\pi_k \prod_{t=1}^{N_t} \mathcal{N} \left(y_v^{[t]}; \mu_{y,k}^{[t](n)}, \sigma_{y,k}^{[t](n)} \right)}{\sum_{k'=1}^{N_k} \pi_{k'} \prod_{t=1}^{N_t} \mathcal{N} \left(y_v^{[t]}; \mu_{y,k'}^{[t](n)}, \sigma_{y,k'}^{[t](n)} \right)} \tag{4.103}$$

Chapter 5

Unified 4-D computed tomography

5.1 4-D computed tomography

In medical applications, emission imaging system are operated in the *photon limited* regime. The small number of photons emitted during emission scans, as highlighted by the experiments in Chapter 3, may be considered the major limit to the amount of information that can be acquired in a single scan. Perhaps the second most important information bottleneck is *motion* of the patient during the acquisition. While motion poses similar challenges also in transmission tomography and in nuclear magnetic resonance imaging, the different constraints imposed by the three imaging techniques determine quite different problems and solutions.

The recent development of ultra-fast X-Ray imaging systems has enabled rates of acquisition of the information sufficient to obtain a tomographic image within time frames as short as a few milliseconds. These new technologies enable the acquisition of sequences of motion-free 3-D images. Ultra-fast X-Ray acquisitions are enabled by gantries that rotate at speeds as high as 300 rounds per minute (RPM) [74] and by the use of multiple sources and multiple detectors [75]. Beyond the enabling technologies, the key enabling factor for ultra-fast X-Ray CT is that safety constraints relate to the *integral* dose of radiation. Reducing the acquisition time allows for higher intensity of the X-Ray source.

Radiation dose is not a concern in MRI. Fast gradient-echo MRI sequences enable the acquisition of multiple MRI images per second, however electronic noise becomes predominant for short time frames, making the fast acquisition of sequences of images a challenging problem. Despite the difficulties, 4-D MR imaging has proven diagnostic value in several areas of application, including heart imaging.

The fundamental challenge of 4-D imaging is that the information that can be acquired within short time frames is limited. The shorter the time frames, the more noise dominates over the signals. Operation in the photon-limited regime in emission imaging imposes long acquisition times. Time frames short enough to render negligible the effect of motion, due to the small number of photons observed, generally present insufficient information for reconstruction.

Frame-by-frame reconstruction, however, presents redundancy. The state of the variables of interest,

be it anatomical, functional, or a joint state, in a given time frame, is related to the state in other time frames. If one chooses time frames short enough to make motion negligible, one may then consider the state at a given time frame approximately equal to the state at the next time frame. Spatio-temporal regularisation, or with a choice of wording more suited to the Bayesian approach, *4-D modelling* is the focus of this chapter. The Poisson observation model and the joint model of anatomy and function of Chapter 4 are extended to four dimensions, presenting a mathematical framework that enables the exploration of some of the possibilities offered by PET and PET-MR systems to cope with motion.

4-D emission tomography has been addressed in the literature with different approaches. The first categorisation is that between the model-based and model-free approaches. Frame-by-frame reconstruction may be considered the model-free alternative, where one ignores any relation between the time frames. The broad category of the model-based approaches includes all the methods that aim at describing or capturing the time dependence of the imaging data. The second categorisation that one may describe is one that separates *data-driven* and *sensor-based* approaches. The sensor-based approaches make use of sensors of motion and motion-tracking devices [76], while the data-driven approaches aim at extracting the information about motion directly from the imaging data (i.e. the photon counts) [77, 78, 79, 80]. The model-based approaches, regardless if data-driven or sensor-based, differ in the choice of the deformation model. Rigid, affine and many forms of non-rigid parametrisations of motion may be adopted depending on the imaging application (see e.g. [81]). In brain imaging, the deformations are well represented by a rigid model, with typical translations in the range of 5-20 mm and rotations of 1-4 deg observed during PET and SPECT scans [76]. Different types of motion have determined a number of different approaches. The typical motion pattern of the head of a patient laying in a scanner is characterised by large movements occurring at discrete times, sporadically (once every 10-20 seconds) [76]. Such pattern suggests *event-driven* motion compensation methods based on motion information provided by a external motion-tracking devices [79]. The approximately cyclical motion patterns that occur due to breathing and to the heart cycle have suggested methods that merge the information acquired over multiple cycles [82]. While we are referring here in particular to emission computed tomography, some of the approaches described above have been applied also to transmission tomography, in particular for applications where the ultra-fast acquisition is prohibited. A review of motion correction approaches in X-Ray CT can be found in [83].

The inherent co-registration and simultaneity of the acquisitions in PET-MR and the harmlessness of magnetic resonance introduce unprecedented possibilities to address to confounding effect of motion. The present work is inspired by the new possibilities offered by PET-MR. A unifying mathematical framework for 4-D multi-modal reconstruction is described based on the physiologically-based modelling paradigm introduced in Chapter 4. Though inspired on the new possibilities offered by PET-MR, the mathematical framework described in the following unifies the approaches described above: the model-based and model-free approaches; data-driven and sensor-based measurements; event-driven,

continuous and cyclical estimation of motion; rigid and non-rigid parametrisation.

The first part of the chapter (paragraphs 5.2-5.3) describes the mathematical model, while the second part (paragraphs 5.4-5.7) describes its applications. Paragraph 5.4 describes a robust algorithm for data-driven rigid motion reconstruction; paragraph 5.5 describes the extension to 4-D of the MLEM algorithm for sensor-based motion correction; paragraph 5.6 describes an event-driven motion reconstruction strategy for PET-MR and paragraph 5.7 an algorithm for longitudinal registration of images.

5.2 The 4-D unified modeling approach

The model described in the following extends to four dimensions (3-D plus time), the model described in chapter 4. Analogously to the discretisation of space in voxels (paragraph 2.3), let us discretise time, for numerical purposes, in N_t discrete frames indexed by $t = 1, 2, \ldots, N_t$. The object observed by the imaging system is subject to deformations, function of time. Let us define a homogeneous coordinates system:

$$X = [X_1, X_2, X_3, 1]^T \tag{5.1}$$

In order to discretise the problem spatially, let us define a grid of regularly spaced voxels indexed by $v = 1, 2, \ldots, N_v$ and let X_v denote their coordinates:

$$X_v = [X_{v1}, X_{v2}, X_{v3}, 1]^T \tag{5.2}$$

Let us define the spatial transformation $T^{\gamma^{[t]}}$, parametrised by parameters $\gamma^{[t]}$, function of time:

$$T^{\gamma^{[t]}} = \begin{bmatrix} \gamma_{11} & \gamma_{12} & \gamma_{13} & \gamma_{14} \\ \gamma_{21} & \gamma_{22} & \gamma_{23} & \gamma_{24} \\ \gamma_{31} & \gamma_{32} & \gamma_{33} & \gamma_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.3}$$

$T^{\gamma^{[t]}}$ maps locations X_v in the reference frame to locations $X_v^{[t]}$ at time t:

$$X_v^{[t]} \triangleq T^{\gamma^{[t]}} X_v \tag{5.4}$$

For simplicity, in this formulation, the spatial transformation is parametrised with a global transformation matrix. This simple parametrisation simplifies the derivations that follow. However, the model is extended to account for non-rigid deformations in the second part of the chapter (experiments with non-rigid transformations are described in 5.4.2). Considering the linear transformation (5.3), $\gamma_{14}^{[t]}$, $\gamma_{24}^{[t]}$, $\gamma_{34}^{[t]}$ are translation parameters and the remaining parameters encode rotation, scaling and shearing.

The first assumption that underlies the unified model for multi-modal imaging, already described in 4, is that there exist hidden anatomical-functional states, defined at discrete spatial locations. It is assumed that the hidden variables that represent the anatomical-functional states are constant (though

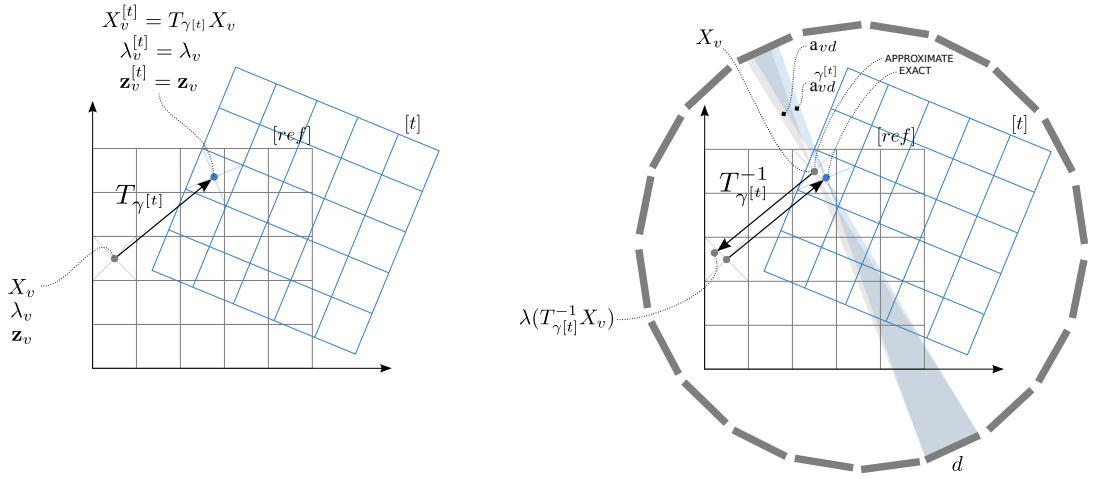


Figure 5.1: Left: transformation from reference frame to frame $[t]$. Right: approximated transformation for the efficient estimation of the gradients in projection space (see (5.19)-(5.23) and (5.40)-(5.55)).

unknown) over time. As in 4.2, the hidden variables are expressed with a 1-of-N_k representation:

$$\mathbf{z}_v = [z_{v1}, z_{v2}, \ldots, z_{vk}, \ldots, z_{vN_k}] \tag{5.5}$$

$$z_{vk} \in \{0, 1\} \qquad \sum_{k=1}^{N_k} z_{vk} = 1 \tag{5.6}$$

$$\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_v, \ldots, \mathbf{z}_{N_v}] \tag{5.7}$$

$\mathbf{z}_v$ can be thought of as the value of function of space $\mathbf{z}$ in X_v:

$$\mathbf{z}_v = \mathbf{z}(X_v) \tag{5.8}$$

As locations X_v are mapped to $X_v^{[t]}$, under the assumption that the hidden states are constant over time, letting $\mathbf{z}^{[t]}$ be the hidden states at time t:

$$\mathbf{z}^{[t]}(X_v^{[t]}) = \mathbf{z}^{[t]}(T^{\gamma^{[t]}} X_v) \triangleq \mathbf{z}_v^{[t]} \tag{5.9}$$

$$\mathbf{z}_v^{[t]} = \mathbf{z}_v \tag{5.10}$$

Notice that though $\mathbf{z}$ and $\mathbf{z}^{[t]}$ are indicated a functions of space, $\mathbf{z}$ only needs to be defined in X_v and $\mathbf{z}^{[t]}$ in $X_v^{[t]}$.

The second assumption is that activity is non-zero only at locations X_v. It is assumed that the rate of emission of the point sources does not change, although they are dislocated over time according to $T^{\gamma^{[t]}}$. If one could go and measure the rate of emission at location $X_v^{[t]}$ at time t, it would be equal to the emission rate at location X_v in the reference frame. Let us use again superscript $[t]$ to indicate the function of space at time $[t]$:

$$\boldsymbol{\lambda}^{[t]}(X_v^{[t]}) = \boldsymbol{\lambda}^{[t]}(T^{\gamma^{[t]}} X_v) \triangleq \lambda_v^{[t]} \tag{5.11}$$

$$\lambda_v^{[t]} = \lambda_v \tag{5.12}$$

The rate of emission, proportional to the spatial concentration of radio-pharmaceutical, is assumed to be the realisation of a random process characterised by normal probability distribution, when the hidden state is known (see 4.2):

$$p(\lambda_v | z_{vk} = 1, \theta_\lambda) = \mathcal{N}(\lambda_v; \mu_{\lambda_k}, \sigma_{\lambda_k}) \tag{5.13}$$

θ_λ is the set of the parameters of the pharmaceutical uptake model, as already discussed 4.2. Because the hidden state $\mathbf{z}_v$ in v is expressed with a 1-of-N_k representation, this last expression can be written as:

$$p(\lambda_v | \mathbf{z}_v, \theta_\lambda) = \prod_{k=1}^{N_k} \mathcal{N}(\lambda_v; \mu_{\lambda_k}, \sigma_{\lambda_k})^{z_{vk}} \tag{5.14}$$

This second notation will come to use when we will derive the marginal probability distributions.

Denoting with $\boldsymbol{y}^{[t]}$ the MR image acquired at time t, given the assumption that the hidden states are not a function of time, which we have expressed in (5.10), we get:

$$\boldsymbol{y}^{[t]}(X_v^{[t]}) = \boldsymbol{y}^{[t]}(T^{\gamma^{[t]}} X_v) \triangleq y_v^{[t]} \tag{5.15}$$

$$p(y_v^{[t]} | z_{vk} = 1, \theta_y, \gamma[t]) = \mathcal{N}\left(y_v^{[t]}; \mu_{y_k}, \sigma_{y_k}\right) \tag{5.16}$$

Equivalently, given the 1-of-N_k representation of $\mathbf{z}_v$:

$$p(y_v^{[t]} | \mathbf{z}_v, \theta_y, \gamma[t]) = \prod_{k=1}^{N_k} \mathcal{N}\left(y_v^{[t]}; \mu_{y_k}, \sigma_{y_k}\right)^{z_{vk}} \tag{5.17}$$

Notice that if one questions what the value of the t-th MR image is expected to be in location $X_v^{[t]}$, the model says that if the hidden state is known, then the probability distribution associated to the value of the MR image is Gaussian (it is a mixture density if one does not know the state of the hidden variable). However, if one questions what the value of the MR image at time t is expected to be in a location other than $X_v^{[t]}$, the model does not (and does not need to) give an answer.

We have introduced the following assumptions about the rate of emission: 1) it is non-zero only at locations X_v in the reference frame, relocated over time by $T^{\gamma^{[t]}}$; 2) that the rate of emission of each point source does not change over time (5.12). The conditional probability distribution associated to the photon counts at time frame t, when activity is known, is then:

$$p(\boldsymbol{q}^{[t]} | \boldsymbol{\lambda}, \gamma^{[t]}) \quad = \quad \prod_{d}^{N_d} \mathcal{P}\left(q_d^{[t]}; \sum_v a_{vd}^{\gamma^{[t]}} \lambda_v^{[t]}\right) \tag{5.18}$$

where $a_{vd}^{\gamma^{[t]}}$ is the probability that a pair of photons emitted from location $X_v^{[t]}$ is detected in correspondence of Line of Response (LOR) d. The transition matrix $A^{\gamma^{[t]}} = \{a_{vd}^{\gamma^{[t]}}\}$ is parametrised by $\gamma^{[t]}$, therefore time dependent. Though (5.18) is the correct notation, let us express the generative model of the emission imaging system with an alternative notation that breaks the discrete formulation described thus far but enables a more computationally efficient computation of the conditional probabilities and of

the derivatives of the conditional probability distributions. If the spatial transformations are invertible, from (5.12) we have:

$$\boldsymbol{\lambda}^{[t]}(T^{\gamma^{[t]}}X_v) = \boldsymbol{\lambda}(X_v) \tag{5.19}$$

$$\boldsymbol{\lambda}^{[t]}(T^{\gamma^{[t]}}T^{\gamma^{[t]}}X_v) = \boldsymbol{\lambda}(T^{\gamma^{[t]}}X_v) \tag{5.20}$$

$$\boldsymbol{\lambda}^{[t]}(X_v) = \boldsymbol{\lambda}(T^{\gamma^{[t]}}X_v) \tag{5.21}$$

Then we can write, substituting in (5.18):

$$p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda},\gamma^{[t]}) = \prod_d^{N_d} \mathcal{P}\left(q_d^{[t]}; \sum_v a_{vd}\boldsymbol{\lambda}^{[t]}(X_v)\right) \tag{5.22}$$

$$= \prod_d^{N_d} \mathcal{P}\left(q_d^{[t]}; \sum_v a_{vd}\boldsymbol{\lambda}(T^{\gamma^{[t]}}X_v)\right) \tag{5.23}$$

Strictly speaking $\boldsymbol{\lambda}(T^{\gamma^{[t]}}X_v)$ is not defined, having assumed that $\boldsymbol{\lambda}$ is defined only in X_v. While this last expression breaks the formulation, we will use it, where specified (equations (5.40)-(5.55))), along with an interpolation scheme to compute $\boldsymbol{\lambda}(T^{\gamma^{[t]}}X_v)$, for the purpose of reducing the computational complexity.

The hidden variables are considered to be a priori independent from one another and their probability distribution is multinomial. This choice allows us to introduce population-derived information.

$$p(z_{vk} = 1|\pi) = \pi_{vk} \tag{5.24}$$

Let us indicate with π_v the prior probability of each of the allowed states in voxel v:

$$\pi_v = [\pi_{v1}, \pi_{v2}, \ldots, \pi_{vk}, \ldots, \pi_{vN_k}] \tag{5.25}$$

$$0 \leq \pi_{vk} \leq 1 \qquad \sum_{k=1}^{N_k} \pi_{vk} = 1 \tag{5.26}$$

and with π the set of all the parameters of the (spatially varying) multinomial prior probability distributions of the hidden states:

$$\pi = [\pi_1, \pi_2, \ldots, \pi_v, \ldots, \pi_{N_v}] \tag{5.27}$$

Because the hidden state $\mathbf{z}_v$ in v uses a 1-of-N_k representation, the prior probability distribution associated to the state in v can also be written as:

$$p(\mathbf{z}_v|\pi) = \prod_{k=1}^{N_k} \pi_{vk}^{z_{vk}} \tag{5.28}$$

The prior probability associated to the set of all the hidden variables is:

$$p(\mathbf{Z}|\pi) = \prod_{v=1}^{N_v}\prod_{k=1}^{N_k} \pi_{vk}^{z_{vk}} \tag{5.29}$$

Fig. 5.2 reports the directed acyclic graph of the model.

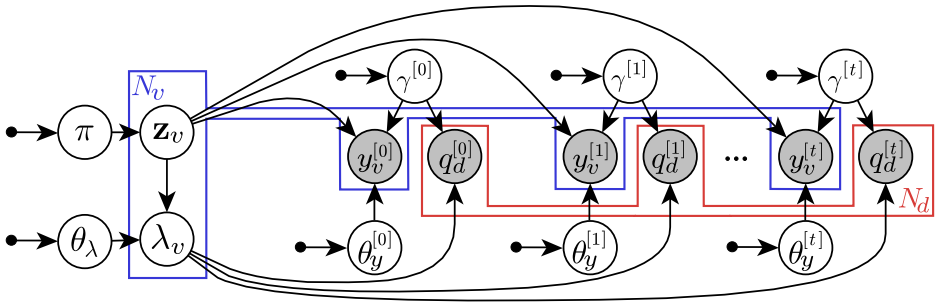


Figure 5.2: DAG of the unified model for multi-modal 4-D tomography. Observed (known) quantities are shaded. Voxels are indexed by v and lines of response (LOR) of the PET detector are indexed by d. The hidden anatomical/functional state $\mathbf{z}_v$ determines activity λ and, along with deformation parameters $\gamma^{[t]}$, determines MRI intensity $y^{[t]}$ at time frame t. The dependence of activity and MRI intensity from the hidden states is parametrised by $\theta = \{\theta_{\boldsymbol{\lambda}}, \theta_y\}$. Activity and the deformation parameters determine the photon counts $q^{[t]}$ at time t. The parameters π parameterise the prior probability distribution of the hidden states.

5.3 Joint estimation of the model parameters

From the graph in Fig. 5.2, the joint probability distribution is:

$$p(\boldsymbol{\lambda}, \boldsymbol{y}, \mathbf{Z}, \boldsymbol{q}, \gamma, \theta, \pi) = p(\pi)p(\theta_\lambda)p(\theta_y)p(\gamma)p(\mathbf{Z}|\pi)p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_\lambda) \prod_t^{N_t} p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda}, \gamma^{[t]})p(\boldsymbol{y}^{[t]}|\mathbf{Z}, \theta_y, \gamma^{[t]}) \tag{5.30}$$

Substituting the expressions (5.14), (5.17), (5.18), (5.29) and summing over $\mathbf{Z}$ as in 4.3, one obtains again a differentiable expression of the log of the posterior (as long as the prior probability distributions are differentiable). However we wish to alternate the optimisation of subsets of the variables. Let us consider the following subsets:

- $\mathcal{S}_1 : \boldsymbol{\lambda}$
- $\mathcal{S}_2 : \gamma = \{\gamma_1, \ldots, \gamma_t, \ldots, \gamma_{N_t}\}$
- $\mathcal{S}_3 : \theta = \{\theta_\lambda, \theta_y\}$,

and derive analytical expressions of the log marginal probability distributions of each subset conditioned on the other subsets and of their first derivatives. The following paragraphs make use of these expressions to derive algorithms for the optimisation of the parameters. Notice that the model of the magnetic field distortion may be included as in the previous chapter, entailing a 4-th subset; the model of the magnetic field distortion has been left out in this chapter in order to keep the formulation uncluttered.

5.3.1 Marginal posterior distribution of the activity

The marginal posterior probability distribution of the activity $\boldsymbol{\lambda}$, conditioned on the motion parameters γ and on the model parameters θ has the following expression:

$$p(\boldsymbol{\lambda}|\boldsymbol{q},\boldsymbol{y},\gamma,\theta,\pi) = \sum_{\mathbf{Z}} p(\boldsymbol{\lambda},\mathbf{Z}|\boldsymbol{q},\boldsymbol{y},\gamma,\theta,\pi) \tag{5.31}$$

$$= \sum_{\mathbf{Z}} \frac{p(\boldsymbol{\lambda},\mathbf{Z},\boldsymbol{q},\boldsymbol{y},\gamma,\theta,\pi)}{p(\boldsymbol{q},\boldsymbol{y},\gamma,\theta,\pi)} \tag{5.32}$$

Substituting from (5.30):

$$p(\boldsymbol{\lambda}|\boldsymbol{q},\boldsymbol{y},\gamma,\theta,\pi) \propto \prod_{t}^{N_t}\left[p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda},\gamma^{[t]})\right] \sum_{\mathbf{Z}}\left[p(\mathbf{Z}|\pi)p(\boldsymbol{\lambda}|\mathbf{Z},\theta_\lambda)\prod_{t}^{N_t}\left[p(\boldsymbol{y}^{[t]}|\mathbf{Z},\theta_y,\gamma^{[t]})\right]\right] \tag{5.33}$$

Let us differentiate it with respect to the components of vector-valued $\boldsymbol{\lambda}$ and get:

$$\frac{\partial}{\partial\lambda_r}\log p(\boldsymbol{\lambda}|\boldsymbol{q},\boldsymbol{y},\gamma,\theta,\pi) = \underbrace{\frac{\partial}{\partial\lambda_r}\log\prod_{t}^{N_t}p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda},\gamma^{[t]})}_{\text{Emission term}} + \tag{5.34}$$

$$+ \underbrace{\frac{\partial}{\partial\lambda_r}\log\sum_{\mathbf{Z}}\left[p(\mathbf{Z}|\pi)p(\boldsymbol{\lambda}|\mathbf{Z},\theta_\lambda)\prod_{t}^{N_t}\left[p(\boldsymbol{y}^{[t]}|\mathbf{Z},\theta_y,\gamma^{[t]})\right]\right]}_{\text{Finite mixture term}}$$

Let us consider the two terms one at the time.

Emission term

Let us expand the emission term of equation (5.35):

$$\frac{\partial}{\partial\lambda_r}\log\prod_{t}^{N_t}p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda},\gamma^{[t]}) = \sum_{t=1}^{N_t}\frac{\partial}{\partial\lambda_r}\log\prod_{d=1}^{N_d}\mathcal{P}\left(q_d^{[t]};\sum_{v=1}^{N_v}a_{vd}^{\gamma^{[t]}}\right) \tag{5.35}$$

$$= \sum_{t=1}^{N_t}\frac{\partial}{\partial\lambda_r}\sum_{d=1}^{N_d}\log\frac{e^{-\sum_{v=1}^{N_v}a_{vd}^{\gamma^{[t]}}\lambda_v}\left[\sum_{v=1}^{N_v}a_{vd}^{\gamma^{[t]}}\lambda_v\right]^{q_d^{[t]}}}{q_d^{[t]}!} \tag{5.36}$$

$$= \sum_{t=1}^{N_t}\sum_{d=1}^{N_d}\frac{\partial}{\partial\lambda_r}\left[-\sum_{v=1}^{N_v}a_{vd}^{\gamma^{[t]}}\lambda_v + q_d^{[t]}\log\sum_{v=1}^{N_v}a_{vd}^{\gamma^{[t]}}\lambda_v\right] \tag{5.37}$$

$$= -\sum_{t=1}^{N_t}\sum_{d=1}^{N_d}a_{rd}^{\gamma^{[t]}} + \sum_{t=1}^{N_t}\sum_{d=1}^{N_d}a_{rd}^{\gamma^{[t]}}\frac{q_d^{[t]}}{\sum_{v=1}^{N_v}a_{vd}^{\gamma^{[t]}}\lambda_v} \tag{5.38}$$

$$\underbrace{\frac{\partial}{\partial\lambda_r}\log\prod_{t}^{N_t}p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda},\gamma^{[t]})}_{\text{Emission term}} = -\sum_{t=1}^{N_t}\sum_{d=1}^{N_d}a_{rd}^{\gamma^{[t]}} + \sum_{t=1}^{N_t}\sum_{d=1}^{N_d}a_{rd}^{\gamma^{[t]}}\frac{q_d^{[t]}}{\sum_{v=1}^{N_v}a_{vd}^{\gamma^{[t]}}\lambda_v} \tag{5.39}$$

The computation of the emission term therefore entails a series of N_t projections and back-projections. The system matrix is different for each of the N_t projections and back-projections, as expressed by $a^{[t]}$ in the formula (5.39). The following paragraph describes how (5.39) can be re-formulated in order to obtain an efficient algorithm to compute the N_t projections and back-projections.

Efficient computation of the gradient

Computing the gradient (5.38) consists in the computation of N_t projections and back-projections. As expressed by the summation

$$\sum_{v=1}^{N_v} a_{vd}^{\gamma^{[t]}} \lambda_v \tag{5.40}$$

that appears in (5.38), each of the N_t projections has a different system matrix. In fact, as the point sources, initially located in X_v, are dislocated over time to $T^{\gamma^{[t]}}$, the probability that a photon emitted from the v-th point source changes. One way to compute the projections and the back-projections is to use a ray-casting algorithm that accounts for the location of the point sources at each time point, implementing directly (5.40). As discussed in chapter 10, when the point sources are displaced on a regular grid, the numerical implementations of the projector and back-projector can be considerably more efficient. If the point sources are considered to be located on a regularly spaced grid in the reference frame, with X_v representing the locations of the point sources, considering that transformations are non-rigid, the transformed locations $T^{\gamma^{[t]}}$ are no longer regularly spaced.

In (5.11) we have defined the function of 3-D space $\boldsymbol{\lambda}^{[t]} : \boldsymbol{\lambda}^{[t]}(T^{\gamma^{[t]}} X_v) = \boldsymbol{\lambda}(X_v)$. According to this definition $\boldsymbol{\lambda}^{[t]}$, consistent with the framework, which defines the hidden states only at discrete locations, needs only to be defined in $T^{\gamma^{[t]}} X_v$. However, in order to reduce the computational complexity, let use break the formulation and consider $\boldsymbol{\lambda}^{[t]}$ and $\boldsymbol{\lambda}$ continuous functions of space. Then, if the transformation is invertible, we get:

$$\begin{aligned}
\boldsymbol{\lambda}^{[t]}(T^{\gamma^{[t]}} X_v) &= \boldsymbol{\lambda}(X_v) && (5.41)\\
\boldsymbol{\lambda}^{[t]}(T^{\gamma^{[t]}-1} T^{\gamma^{[t]}} X_v) &= \boldsymbol{\lambda}(T^{\gamma^{[t]}-1} X_v) && (5.42)\\
\boldsymbol{\lambda}^{[t]}(X_v) &= \boldsymbol{\lambda}(T^{\gamma^{[t]}-1} X_v) && (5.43)\\
& && (5.44)
\end{aligned}$$

Let us define:

$$A = \{a_{vd}\} \tag{5.45}$$

as the matrix of the probabilities that a pair of photons emitted in X_v are detected in d, with X_v being the locations of the voxels in the reference frame. Notice that we have already defined $A^{\gamma^{[t]}} = \{a_{vd}^{\gamma^{[t]}}\}$ as the matrix of the probabilities that a pair of photons emitted from location $T^{\gamma^{[t]}} X_v$ is detected in d.

In the infinite dimensional case (activity is a continuous function of space), the following expressions of the (vector-valued) expectation of $\boldsymbol{q}$ are equivalent:

$$A^{\gamma^{[t]}}\boldsymbol{\lambda}(X) = A\boldsymbol{\lambda}(T^{\gamma^{[t]}-1}X) \tag{5.46}$$

Then, discretising the integration:

$$A^{\gamma^{[t]}}\boldsymbol{\lambda}(X_v) = A\boldsymbol{\lambda}(T^{\gamma^{[t]}-1}X_v) \tag{5.47}$$

This breaks the discrete formulation given thus far (although for the sole purpose of obtaining an implementation that makes better use of the computing resources): strictly speaking $\boldsymbol{\lambda}$ is only defined at locations X_v, thus it is not defined in $T^{\gamma^{[t]}-1}X_v$. However it can be obtained by interpolation of the point sources X_v nearest to each of $T^{\gamma^{[t]}-1}X_v$. In other words, in order to compute $\boldsymbol{\lambda}^{[t]}(X_v) = \boldsymbol{\lambda}(T^{\gamma^{[t]}-1}X_v)$ one may proceed by computing locations $T^{\gamma^{[t]}-1}X_v$, then sampling $\boldsymbol{\lambda}$ in each of the transformed locations, interpolating from the nearest voxels (see Fig. 5.1-right). Then the transformation and re-interpolation can be expressed as a linear operator $\tau_{\gamma^{[t]}}$:

$$\boldsymbol{\lambda}^{[t]}(X_v) = \tau_{\gamma^{[t]}}\boldsymbol{\lambda}(X_v) \tag{5.48}$$

Substituting in (5.38):

$$\frac{\partial}{\partial\lambda_r}\log\prod_{t=1}^{N_t}p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda},\gamma) = \tag{5.49}$$

$$= \sum_{t=1}^{N_t}\frac{\partial}{\partial\lambda_r}\sum_{d=1}^{N_d}\log\frac{e^{-\left[P\tau_{\gamma^{[t]}}\boldsymbol{\lambda}(X_v)\right]_d}\left[P\tau_{\gamma^{[t]}}\boldsymbol{\lambda}(X_v)\right]_d^{q_d^{[t]}}}{q_d^{[t]}!} \tag{5.50}$$

$$= \sum_{t=1}^{N_t}\frac{\partial}{\partial\lambda_r}\sum_{d=1}^{N_d}-\left[P\tau_{\gamma^{[t]}}\boldsymbol{\lambda}(X_v)\right]_d + q_d^{[t]}\log\left[P\tau_{\gamma^{[t]}}\boldsymbol{\lambda}(X_v)\right]_d \tag{5.51}$$

$$= \sum_{t=1}^{N_t}\frac{\partial}{\partial\lambda_r}\sum_{d=1}^{N_d}-\sum_{v=1}^{N_v}a_{vd}\sum_{v'=1}^{N_v}\tau_{v'v}^{\gamma^{[t]}}\lambda_{v'} + q_d^{[t]}\log\sum_{v=1}^{N_v}a_{vd}\sum_{v'=1}^{N_v}\tau_{v'v}^{\gamma^{[t]}}\lambda_{v'} \tag{5.52}$$

$$= \sum_{t=1}^{N_t}\sum_{v=1}^{N_v}\tau_{rv}^{\gamma^{[t]}}\sum_{d=1}^{N_d}a_{rd} + \sum_{t=1}^{N_t}\sum_{v=1}^{N_v}\tau_{rv}^{\gamma^{[t]}}\sum_{d=1}^{N_d}a_{rd}\frac{q_d^{[t]}}{\sum_{v=1}^{N_v}a_{vd}\sum_{v'=1}^{N_v}\tau_{v'v}^{\gamma^{[t]}}\lambda_{v'}} \tag{5.53}$$

$$= \sum_{t=1}^{N_t}\tau_{\gamma^{[t]}}^T P^T\mathbf{1} + \sum_{t=1}^{N_t}\tau_{\gamma^{[t]}}^T P^T\frac{\boldsymbol{q}^{[t]}}{P\tau_{\gamma^{[t]}}\boldsymbol{\lambda}} \tag{5.54}$$

$$\underbrace{\frac{\partial}{\partial\lambda_r}\log\prod_{t=1}^{N_t}p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda},\gamma)}_{\text{Approximate emission term}} = \sum_{t=1}^{N_t}\tau_{\gamma^{[t]}}^T P^T\mathbf{1} + \sum_{t=1}^{N_t}\tau_{\gamma^{[t]}}^T P^T\frac{\boldsymbol{q}^{[t]}}{P\tau_{\gamma^{[t]}}\boldsymbol{\lambda}} \tag{5.55}$$

In summary, the gradient of the marginal probability distribution of the activity with respect to the emission rate parameters λ_r is composed of two terms, just like in the static case, which can be computed as follows:

- Left term of (5.55):
  1. Back-projection of the vector $[1, 1, 1, 1, \dots]$;
  2. Transformation of the back-projection by each of the N_t inverse transformations;
  3. Summation of the N_t transformed back-projections.
- Right term of (5.55):
  1. Re-sampling of the current estimate of the activity $\boldsymbol{\lambda}$ according to each transformation;
  2. Projection of each transformed activity;
  3. Point-wise division of the measurement (photon counts) at time t by the projection of the transformed activity;
  4. Back-projection of each of the N_t terms obtained at the previous step;
  5. Inverse transformation of each of the N_t back-projections;
  6. Summation of the N_t back-projections.

Finite mixture term

Let us now expand the finite mixture term of equation (5.35). Proceeding as in 4.3, the summation over $\mathbf{Z}$ can be written as a product over v of sums over $\mathbf{z}_v$:

$$\frac{\partial}{\partial \lambda_r} \log \sum_{\mathbf{Z}} \left[p(\mathbf{Z}|\pi) p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_\lambda) \prod_{t}^{N_t} \left[p(\boldsymbol{y}^{[t]}|\mathbf{Z}, \theta_y, \gamma^{[t]}) \right] \right] = \tag{5.56}$$

$$= \frac{\partial}{\partial \lambda_r} \log \prod_{v=1}^{N_v} \left[\sum_{\mathbf{z}_v} \prod_{k=1}^{N_k} \pi_{vk}^{z_{vk}} \mathcal{N}\left(\lambda_v, \mu_{\lambda,k}, \sigma_{\lambda,k}\right)^{z_{vk}} \prod_{t=1}^{N_t} \mathcal{N}\left(y_v^{[t]}, \mu_{y,k}, \sigma_{y,k}\right)^{z_{vk}} \right] \tag{5.57}$$

$$= \frac{\partial}{\partial \lambda_r} \sum_{v=1}^{N_v} \log \sum_{k=1}^{N_k} \pi_{vk} \mathcal{N}\left(\lambda_v, \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \prod_{t=1}^{N_t} \mathcal{N}\left(y_v^{[t]}, \mu_{y,k}, \sigma_{y,k}\right) \tag{5.58}$$

$$= \frac{\sum_{k=1}^{N_k} \frac{\mu_{\lambda,k} - \lambda_r}{\sigma_{\lambda,k}^2} \pi_{rk} \mathcal{N}\left(\lambda_r, \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \prod_{t=1}^{N_t} \mathcal{N}\left(y_r^{[t]}, \mu_{y,k}, \sigma_{y,k}\right)}{\sum_{k=1}^{N_k} \pi_{rk} \mathcal{N}\left(\lambda_r, \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \prod_{t=1}^{N_t} \mathcal{N}\left(y_r^{[t]}, \mu_{y,k}, \sigma_{y,k}\right)} \tag{5.59}$$

$$= \sum_{k=1}^{N_k} l_{rk} \frac{\mu_{\lambda,k} - \lambda_r}{\sigma_{\lambda,k}^2} \tag{5.60}$$

$$\underbrace{\frac{\partial}{\partial \lambda_r} \log \sum_{\mathbf{Z}} \left[p(\mathbf{Z}|\pi) p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_\lambda) \prod_{t}^{N_t} \left[p(\boldsymbol{y}^{[t]}|\mathbf{Z}, \theta_y, \gamma^{[t]}) \right] \right]}_{\text{Finite mixture term}} = \sum_{k=1}^{N_k} l_{rk} \frac{\mu_{\lambda,k} - \lambda_r}{\sigma_{\lambda,k}^2} \tag{5.61}$$

with

$$l_{rk} = \frac{\pi_{rk}\mathcal{N}\left(\lambda_r, \mu_{\lambda,k}, \sigma_{\lambda,k}\right)\prod_{t=1}^{N_t}\mathcal{N}\left(y_r^{[t]}, \mu_{y,k}, \sigma_{y,k}\right)}{\sum_{k=1}^{N_k}\pi_{rk}\mathcal{N}\left(\lambda_r, \mu_{\lambda,k}, \sigma_{\lambda,k}\right)\prod_{t=1}^{N_t}\mathcal{N}\left(y_r^{[t]}, \mu_{y,k}, \sigma_{y,k}\right)} \tag{5.62}$$

In summary, computing the gradient of the finite mixture term with respect to the activity consists in the following steps:

1. Compute the partial membership (5.62);
2. Compute the gradient according to (5.57) and (5.60).

5.3.2 Marginal posterior distribution of the deformation parameters

The marginal posterior probability distribution of the deformation parameters γ, conditioned on the activity $\boldsymbol{\lambda}$ and on the model parameters θ has the following expression:

$$p(\gamma|\boldsymbol{q}, \boldsymbol{y}, \pi, \theta, \boldsymbol{\lambda}) = \sum_{\mathbf{Z}} p(\gamma, \mathbf{Z}|\boldsymbol{\lambda}, \boldsymbol{y}, \boldsymbol{q}, \theta, \pi) \tag{5.63}$$

$$= \sum_{\mathbf{Z}} \frac{p(\boldsymbol{\lambda}, \boldsymbol{y}, \mathbf{Z}, \boldsymbol{q}, \gamma, \theta, \pi)}{p(\boldsymbol{\lambda}, \boldsymbol{y}, \boldsymbol{q}, \theta, \pi)} \tag{5.64}$$

substituting the expression of the joint posterior probability distribution (5.30), taking the $\log$ and differentiating with respect to the r-th parameter of the deformation model at time t:

$$\frac{\partial}{\partial \gamma_r^{[t]}} \log p(\gamma|\boldsymbol{q}, \boldsymbol{y}, \pi, \theta, \boldsymbol{\lambda}) = \tag{5.65}$$

$$= \frac{\partial}{\partial \gamma_r^{[t]}} \log \left[p(\gamma^{[t]}) p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda}, \gamma^{[t]}) \sum_{\mathbf{Z}} \left[p(\mathbf{Z}|\pi) \prod_{t=1}^{N_t} p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_\lambda, \gamma^{[t]}) p(\boldsymbol{y}^{[t]}|\mathbf{Z}, \theta_y, \gamma^{[t]}) \right] \right]$$

$$= \frac{\partial}{\partial \gamma_r^{[t]}} \log p(\gamma^{[t]}) + \tag{5.66}$$

$$+ \underbrace{\frac{\partial}{\partial \gamma_r^{[t]}} \log p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda}, \gamma^{[t]})}_{\text{Emission term}} + \tag{5.67}$$

$$+ \underbrace{\frac{\partial}{\partial \gamma_r^{[t]}} \log \sum_{\mathbf{Z}} \left[p(\mathbf{Z}|\pi) \prod_{t=1}^{N_t} p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_\lambda, \gamma^{[t]}) p(\boldsymbol{y}^{[t]}|\mathbf{Z}, \theta_y, \gamma^{[t]}) \right]}_{\text{Finite mixture term}} \tag{5.68}$$

Let us consider the two terms one at the time.

Emission term

Let us expand the emission term of equation (5.68). The likelihood of the activity and of the motion parameters is given by equation (5.18):

$$
\begin{aligned}
p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda},\gamma^{[t]}) &= \prod_{d=1}^{N_d} p(q_d^{[t]}|\boldsymbol{\lambda},\gamma^{[t]}) && (5.69)\\
&= \prod_{d=1}^{N_d} \mathcal{P}\left(q_d^{[t]};\sum_{v=1}^{N_v} a_{vd}^{\gamma^{[t]}}\lambda_v\right) && (5.70)\\
&= \prod_{d=1}^{N_d} \frac{e^{-\sum_{v=1}^{N_v} a_{vd}^{\gamma^{[t]}}\lambda_v}\left[\sum_{v=1}^{N_v} a_{vd}^{\gamma^{[t]}}\lambda_v\right]^{q_d^{[t]}}}{q_d^{[t]}} && (5.71)\\
& && (5.72)
\end{aligned}
$$

Taking the $\log$ and differentiating, we obtain:

$$
\begin{aligned}
\frac{\partial}{\partial\gamma_r^{[t]}}\log p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda},\gamma^{[t]}) &= \sum_{d=1}^{N_d}\left[\frac{\partial}{\partial\gamma_r^{[t]}} - \sum_{v=1}^{N_v} a_{vd}^{\gamma^{[t]}}\lambda_v + q_d^{[t]}\frac{\partial}{\partial\gamma_r^{[t]}}\log\sum_{v=1}^{N_v} a_{vd}^{\gamma^{[t]}}\lambda_v\right] && (5.73)\\
&= \sum_{d=1}^{N_d}\left[\frac{\partial}{\partial\gamma_r^{[t]}} - \left[A^{\gamma^{[t]}}\boldsymbol{\lambda}(X_v)\right]_d + q_d^{[t]}\frac{\partial}{\partial\gamma_r^{[t]}}\log\left[A^{\gamma^{[t]}}\boldsymbol{\lambda}(X_v)\right]_d\right] && (5.74)\\
&= \sum_{d=1}^{N_d}\left[-\frac{\partial}{\partial\gamma_r^{[t]}}\left[A^{\gamma^{[t]}}\boldsymbol{\lambda}(X_v)\right]_d + q_d^{[t]}\frac{\frac{\partial}{\partial\gamma_r^{[t]}}\left[A^{\gamma^{[t]}}\boldsymbol{\lambda}(X_v)\right]_d}{\left[A^{\gamma^{[t]}}\boldsymbol{\lambda}(X_v)\right]_d}\right], && (5.75)
\end{aligned}
$$

with

$$
A^{\gamma^{[t]}}\boldsymbol{\lambda}(X_v) = A\boldsymbol{\lambda}(T^{\gamma^{[t]}}X_v) \tag{5.76}
$$

Letting:

$$
W_v^{\gamma^{[t]}} = T^{\gamma^{[t]}-1}X_v \tag{5.77}
$$

$$
\frac{\partial}{\partial\gamma_r^{[t]}}W_v^{\gamma^{[t]}} = \frac{\partial}{\partial\gamma_r^{[t]}}T^{\gamma^{[t]}-1}X_v, \tag{5.78}
$$

by the chain rule of differentiation, we obtain:

$$
\begin{aligned}
A\frac{\partial}{\partial\gamma_r^{[t]}}\boldsymbol{\lambda}\left(T^{\gamma^{[t]}-1}X_v\right) &= A\frac{\partial}{\partial\gamma_r^{[t]}}\boldsymbol{\lambda}T^{\gamma^{[t]}}X_v && (5.79)\\
&= A\frac{\partial}{\partial\gamma_r^{[t]}}\boldsymbol{\lambda}\left(W_v^{\gamma^{[t]}}\right) && (5.80)\\
&= A\frac{\partial\boldsymbol{\lambda}(W_v^{\gamma^{[t]}})}{\partial W_v^{\gamma^{[t]}}}\frac{\partial W_v^{\gamma^{[t]}}}{\partial\gamma_r^{[t]}} && (5.81)\\
&= A\left[\frac{\boldsymbol{\lambda}(W_v^{\gamma^{[t]}})}{\partial W_{v,1}^{\gamma^{[t]}}}\frac{\partial W_{v,1}^{\gamma^{[t]}}}{\partial\gamma_r^{[t]}} + \frac{\boldsymbol{\lambda}(W_v^{\gamma^{[t]}})}{\partial W_{v,2}^{\gamma^{[t]}}}\frac{\partial W_{v,2}^{\gamma^{[t]}}}{\partial\gamma_r^{[t]}} + \frac{\boldsymbol{\lambda}(W_v^{\gamma^{[t]}})}{\partial W_{v,3}^{\gamma^{[t]}}}\frac{\partial W_{v,3}^{\gamma^{[t]}}}{\partial\gamma_r^{[t]}}\right] && (5.82)\\
&= A\left[\nabla\boldsymbol{\lambda}\left(T^{\gamma^{[t]}-1}X_v\right)\cdot\left(\frac{\partial}{\partial\gamma_r^{[t]}}T^{\gamma^{[t]}-1}\right)\cdot X_v\right] && (5.83)
\end{aligned}
$$

where the operator ∇ denotes the spatial gradient (in homogeneous coordinates). Expanding (5.75), the derivatives of the emission term with respect to the motion parameters at time t are expressed by

$$\underbrace{\frac{\partial}{\partial \gamma_r^{[t]}} \log p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda}, \gamma^{[t]})}_{\text{Emission term}} = \sum_{d=1}^{N_d} \left[- \left[A \frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{\lambda}(T^{\gamma^{[t]}-1} X_v) \right]_d + q_d^{[t]} \frac{\left[A \frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{\lambda}(T^{\gamma^{[t]}-1} X_v) \right]_d}{\left[A \boldsymbol{\lambda}(T^{\gamma^{[t]}-1} X_v) \right]_d} \right] \quad (5.84)$$

$$\text{where: } A \frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{\lambda} \left(T^{\gamma^{[t]}-1} X_v \right) = A \left[\nabla \boldsymbol{\lambda} \left(T^{\gamma^{[t]}-1} X_v \right) \cdot \left(\frac{\partial}{\partial \gamma_r^{[t]}} T^{\gamma^{[t]}-1} \right) \cdot X_v \right] \quad (5.85)$$

Notice, from (5.84), that the gradient is zero everywhere if $q_d^{[t]} = \left[A \boldsymbol{\lambda}(T^{\gamma^{[t]}-1} X_v) \right]_d$.

The derivative of the emission term with respect to parameter r at time t is obtained, according to (5.84), by performing a summation in detector space ($\sum_d^{N_d}$). The summand is given by two terms. The left term of (5.84) is the projection of the volumetric quantity

$$A \frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{\lambda}(T^{\gamma^{[t]}-1} X_v) \quad (5.86)$$

The right term of (5.84) is equal to the same quantity divided, point-wise, by the projection of $\boldsymbol{\lambda}$ re-sampled in $T^{\gamma^{[t]}-1} X_v$ and multiplied, point wise, by the photon counts. The volume (5.86) expresses the change in the transformed activity in $T^{\gamma^{[t]}-1} X_v$ when parameters r relative to time t is perturbed. Each element is obtained, according to (5.83), by the dot-product of the spatial gradient of $\boldsymbol{\lambda}$ (re-sampled in $T^{\gamma^{[t]}-1} X_v$) with the 4×4 matrix of the derivatives of the spatial transformation, multiplied by the location X_v in the reference space. In summary, the computation of the derivative of the emission term with respect to parameter r relative to time t consists in the following steps:

1. Warp the image support $T^{\gamma^{[t]}-1}$;
2. Re-sampling of the activity on the warped image support;
3. Project the re-sampled activity;
4. Compute the spatial gradient of the re-sampled activity ($N_v \times 4$ matrix - homogeneous coordinates);
5. Compute the matrix of the derivatives of the transformation operator;
6. Multiply (dot-product) the spatial gradient of the re-sampled activity, the matrix of the derivatives of the transformation operator, the image support;
7. Project the volume obtained at the previous step;
8. Point-wise divide the result of step (3) by the result of step (7);
9. Sum, in projection space, the results of (7) and (8).

6 degrees of freedom

Let us consider the case of 6 degrees of freedom (DOF) rigid transformation. Assigning $\gamma_1, \gamma_2, \gamma_3$ to the rotation around axis $1, 2, 3$ respectively and $\gamma_4, \gamma_5, \gamma_6$ to the translation in the direction of axis $1, 2, 3$, the transformation matrix may be parametrised as follows:

$$T^{\gamma^{[t]}} = R_1(\gamma_1^{[t]}) \cdot R_2(\gamma_2^{[t]}) \cdot R_3(\gamma_3^{[t]}) \cdot U(\gamma_4^{[t]}, \gamma_5^{[t]}, \gamma_6^{[t]}) \tag{5.87}$$

$$R_1(\gamma_1^{[t]}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\gamma_1^{[t]} & -\sin\gamma_1^{[t]} & 0 \\ 0 & \sin\gamma_1^{[t]} & \cos\gamma_1^{[t]} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad R_2(\gamma_2^{[t]}) = \begin{bmatrix} \cos\gamma_2^{[t]} & 0 & \sin\gamma_2^{[t]} & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\gamma_2^{[t]} & 0 & \cos\gamma_2^{[t]} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.88}$$

$$R_3(\gamma_3^{[t]}) = \begin{bmatrix} \cos\gamma_3^{[t]} & -\sin\gamma_3^{[t]} & 0 & 0 \\ \sin\gamma_3^{[t]} & \cos\gamma_3^{[t]} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad U(\gamma_4^{[t]}, \gamma_5^{[t]}, \gamma_6^{[t]}) = \begin{bmatrix} 1 & 0 & 0 & \gamma_4^{[t]} \\ 0 & 1 & 0 & \gamma_5^{[t]} \\ 0 & 0 & 1 & \gamma_6^{[t]} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.89}$$

The inverse of $T^{\gamma^{[t]}}$ is:

$$T^{\gamma^{[t]\,-1}} = U(-\gamma_4^{[t]}, -\gamma_5^{[t]}, -\gamma_6^{[t]}) \cdot R_3(-\gamma_3^{[t]}) \cdot R_2(-\gamma_2^{[t]}) \cdot R_1(-\gamma_1^{[t]}) \tag{5.90}$$

$$T^{\gamma^{[t]\,-1}} = \begin{bmatrix} \cos\gamma_1^{[t]}\cos\gamma_2^{[t]} & \cos\gamma_3^{[t]}\sin\gamma_2^{[t]} + \sin\gamma_3^{[t]}\sin\gamma_1^{[t]}\cos\gamma_2^{[t]} & \sin\gamma_3^{[t]}\sin\gamma_2^{[t]} - \cos\gamma_3^{[t]}\sin\gamma_1^{[t]}\cos\gamma_2^{[t]} & \gamma_4^{[t]} \\ -\cos\gamma_1^{[t]}\sin\gamma_2^{[t]} & \cos\gamma_3^{[t]}\cos\gamma_2^{[t]} - \sin\gamma_3^{[t]}\sin\gamma_1^{[t]}\sin\gamma_2^{[t]} & \sin\gamma_3^{[t]}\cos\gamma_2^{[t]} + \cos\gamma_3^{[t]}\sin\gamma_1^{[t]}\sin\gamma_2^{[t]} & \gamma_5^{[t]} \\ \sin\gamma_1^{[t]} & -\sin\gamma_3^{[t]}\cos\gamma_1^{[t]} & \cos\gamma_3^{[t]}\cos\gamma_1^{[t]} & \gamma_6^{[t]} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.91}$$

The derivative of the transformation operator with respect to parameter e.g. $\gamma_2^{[t]}$ (rotation around axis 2 at time t) is then:

$$\frac{\partial}{\partial\gamma_1^{[t]}} T^{\gamma^{[t]\,-1}} = U(-\gamma_4^{[t]}, -\gamma_5^{[t]}, -\gamma_6^{[t]}) \cdot R_3(-\gamma_3^{[t]}) \cdot \begin{bmatrix} -\sin\gamma_2^{[t]} & 0 & \cos\gamma_2^{[t]} & 0 \\ 0 & 1 & 0 & 0 \\ -\cos\gamma_2^{[t]} & 0 & -\sin\gamma_2^{[t]} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot R_1(-\gamma_1^{[t]}) \tag{5.92}$$

This is the matrix required in (5.83) in order to compute the derivative of the emission term with respect to rotation parameter $\gamma_2^{[t]}$. For the 6-DOF parametrisation one has to compute 6 such matrices, 1 spatial transformation of $\boldsymbol{\lambda}$, 1 spatial gradient, 6 dot products and 6 projections. Paragraph 5.4 reports images of the (5.86) and of the projections of (5.86) for the rigid motion parametrisation.

Finite mixture term

Once again, proceeding as in 4.3, the summation over $\mathbf{Z}$ can be written as a product over v of sums over $\mathbf{z}_v$:

$$\underbrace{\frac{\partial}{\partial \gamma_r^{[t]}} \log \sum_{\mathbf{Z}} \left[p(\mathbf{Z}|\pi) \prod_{t=1}^{N_t} p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_\lambda, \gamma^{[t]}) p(\boldsymbol{y}^{[t]}|\mathbf{Z}, \theta_y, \gamma^{[t]}) \right]}_{\text{Finite mixture term}} = \tag{5.93}$$

$$= \frac{\partial}{\partial \gamma_r^{[t]}} \log \prod_{v=1}^{N_v} \sum_{\mathbf{z}_v} \left[\pi_{vk} p(\lambda_v|\mathbf{z}_v, \theta_\lambda) \prod_{t=1}^{N_t} p(y_v^{[t]}|\mathbf{z}_v, \theta_y, \gamma^{[t]}) \right]^{z_{vk}} \tag{5.94}$$

$$= \frac{\partial}{\partial \gamma_r^{[t]}} \log \prod_{v=1}^{N_v} \sum_{k=1}^{N_k} \pi_{vk} p(\lambda_v|z_{vk}, \theta_\lambda) \prod_{t=1}^{N_t} p(y_v^{[t]}|z_{vk}, \theta_y, \gamma^{[t]}) \tag{5.95}$$

$$= \sum_{v=1}^{N_v} \frac{\sum_{k=1}^{N_k} \pi_{vk} p(\lambda_v|z_{vk}, \theta_\lambda) \frac{\partial}{\partial \gamma_r^{[t]}} p(y_v^{[t]}|z_{vk}, \theta_y, \gamma^{[t]}) \prod_{t' \neq t} p(y_v^{[t']}|z_{vk}, \theta_y, \gamma^{[t']})}{\sum_{k=1}^{N_k} \pi_{vk} p(\lambda_v|z_{vk}, \theta_\lambda) \prod_{t=1}^{N_t} p(y_v^{[t]}|z_{vk}, \theta_y, \gamma^{[t]})} \tag{5.96}$$

where:

$$\frac{\partial}{\partial \gamma_r^{[t]}} p\left(y_v^{[t]}(T^{\gamma^{[t]}} X_v)|z_{vk}, \theta_y \right) = \tag{5.97}$$

$$= \frac{\partial}{\partial \gamma_r^{[t]}} \mathcal{N}\left(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \tag{5.98}$$

$$= \frac{\mu_{y,k} - \boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}} X_v\right)}{\sigma_{v,k}^2} \mathcal{N}\left(\boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}} X_v\right); \mu_{y,k}, \sigma_{y,k}\right) \frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}} X_v\right) \tag{5.99}$$

Let us expand $\frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}} X_v\right)$. With the change of variable (as in (5.77)), we obtain:

$$W_v^{\gamma^{[t]}} = T^{\gamma^{[t]}-1} X_v \tag{5.100}$$

$$\frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}} X_v\right) \tag{5.101}$$

$$= \frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{y}^{[t]}\left(W_v^{\gamma^{[t]}}\right) \tag{5.102}$$

$$= \frac{\partial \boldsymbol{y}^{[t]}(W_v^{\gamma^{[t]}})}{\partial W_v^{\gamma^{[t]}}} \frac{\partial W_v^{\gamma^{[t]}}}{\partial \gamma_r^{[t]}} \tag{5.103}$$

$$= \frac{\boldsymbol{y}^{[t]}(W_v^{\gamma^{[t]}})}{\partial W_{v,1}^{\gamma^{[t]}}} \frac{\partial W_{v,1}^{\gamma^{[t]}}}{\partial \gamma_r^{[t]}} + \frac{\boldsymbol{y}^{[t]}(W_v^{\gamma^{[t]}})}{\partial W_{v,2}^{\gamma^{[t]}}} \frac{\partial W_{v,2}^{\gamma^{[t]}}}{\partial \gamma_r^{[t]}} + \frac{\boldsymbol{y}^{[t]}(W_v^{\gamma^{[t]}})}{\partial W_{v,3}^{\gamma^{[t]}}} \frac{\partial W_{v,3}^{\gamma^{[t]}}}{\partial \gamma_r^{[t]}} \tag{5.104}$$

$$= \nabla \boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}} X_v\right) \cdot \left(\frac{\partial}{\partial \gamma_r^{[t]}} T^{\gamma^{[t]}} \right) \cdot X_v \tag{5.105}$$

Notice that (5.96) contains the product of $N_t - 1$ terms $p(y_v^{[t']}|z_{vk}, \theta_y, \gamma^{[t']})$ for $t' \neq t$. Such product is multiplied by $\frac{\partial}{\partial \gamma_r^{[t]}} p\left(y_v^{[t]}(T^{\gamma^{[t]}} X_v)|z_{vk}, \theta_y \right)$. With the particular choice of normal density functions for $p\left(y_v^{[t]}(T^{\gamma^{[t]}} X_v)|z_{vk}, \theta_y \right)$, the product is completed for all t terms as the derivative of a normal density function is proportional to the function, as expressed by (5.99). Substituting (5.99) and (5.105) in (5.96), in fact, the expression simplifies to:

$$\underbrace{\frac{\partial}{\partial \gamma_r^{[t]}} \log \sum_{\mathbf{Z}} \left[p(\mathbf{Z}|\pi) \prod_{t=1}^{N_t} p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_\lambda, \gamma^{[t]}) p(\boldsymbol{y}^{[t]}|\mathbf{Z}, \theta_y, \gamma^{[t]}) \right]}_{\text{Finite mixture term}} = \tag{5.106}$$

$$= \sum_{v=1}^{N_v} \left[\frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{y}^{[t]} \left(T^{\gamma^{[t]}} X_v \right) \sum_{k=1}^{N_k} \iota_{vk} \frac{\mu_{y,k} - \boldsymbol{y}^{[t]} \left(T^{\gamma^{[t]}} X_v \right)}{\sigma_{y,k}^2} \right] \tag{5.107}$$

with l_{vk} expressed by (5.62) and, from (5.105), we obtain:

$$\frac{\partial}{\partial \gamma_r^{[t]}} y_v^{[t]} \left(T^{\gamma^{[t]}} X_v \right) = \nabla \boldsymbol{y}^{[t]} \left(T^{\gamma^{[t]}} X_v \right) \cdot \left(\frac{\partial}{\partial \gamma_r^{[t]}} T^{\gamma^{[t]}} \right) \cdot X_v \tag{5.108}$$

Then, the computation of the derivative of the finite mixture term with respect to the motion parameter r relative to time frame t involves the following steps:

1. Warp the support of the MR image t;
2. Re-sample the MR image t on the warped image support;
3. Compute the spatial gradient of the re-sampled image;
4. Compute the partial membership l_{vk};
5. Multiply (dot-product) the spatial gradient of the re-sampled image, the matrix of the derivatives of the transformation operator, the image support;
6. Compute a weighted sum in image space (see (5.107));
7. Sum all the elements of the volume from the previous step (see (5.107)).

Notice that the derivative of the finite mixture term with respect to the motion parameter does depend on $\boldsymbol{\lambda}$, through l (see (5.62)).

5.3.3 Marginal posterior distribution of the model parameters

The marginal posterior probability distribution of the model parameters θ, conditioned on the activity $\boldsymbol{\lambda}$ and on the motion parameters γ has the following expression:

$$p(\theta|\boldsymbol{\lambda}, \boldsymbol{q}, \boldsymbol{y}, \gamma, \pi) = \sum_{\mathbf{Z}} p(\theta, \mathbf{Z}|\boldsymbol{\lambda}, \boldsymbol{q}, \boldsymbol{y}, \gamma, \pi) \tag{5.109}$$

$$= \sum_{\mathbf{Z}} \frac{p(\boldsymbol{\lambda}, \boldsymbol{y}, \mathbf{Z}, \boldsymbol{q}, \gamma, \theta, \pi)}{p(\boldsymbol{\lambda}, \boldsymbol{q}, \boldsymbol{y}, \gamma, \pi)} \tag{5.110}$$

with $\theta = \{\theta_\lambda, \theta_y\}$. $\theta_\lambda = \{\mu_{\lambda,k}, \sigma_{\lambda,k}\}$ $\theta_y = \{\mu_{y,k}, \sigma_{y,k}\}$, $k = 1, 2, \ldots, N_k$. The gradient with respect to parameters $\mu_{\lambda,k}$, dropping the terms that are not functions of θ and $\mathbf{Z}$, is expressed by:

$$\frac{\partial}{\partial \mu_{\lambda,k}} \log p(\theta|\boldsymbol{\lambda}, \boldsymbol{q}, \boldsymbol{y}, \gamma, \pi) = \tag{5.111}$$

$$= \frac{\partial}{\partial \mu_{\lambda,k}} \log p(\theta) + \log \sum_{\mathbf{Z}} p(\theta) p(\mathbf{Z}|\pi) p(\boldsymbol{\lambda}|\mathbf{Z}, \theta_\lambda) \prod_{t=1}^{N_t} p(\boldsymbol{y}^{[t]}|\mathbf{Z}, \theta_y, \gamma^{[t]}) \tag{5.112}$$

$$= \frac{\partial}{\partial \mu_{\lambda,k}} \log p(\theta) + \sum_{v=1}^{N_v} \log \sum_{k=1}^{N_k} \pi_{vk} \mathcal{N}\left(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \prod_{t=1}^{N_t} \mathcal{N}\left(y_v^{[t]}; \mu_{y,k}, \sigma_{y,k}\right) \tag{5.113}$$

$$= \log p(\theta) + \sum_{v=1}^{N_v} \frac{\pi_{vk} \frac{\partial}{\partial \mu_{\lambda,k}} \mathcal{N}\left(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \prod_{t=1}^{N_t} \mathcal{N}\left(y_v^{[t]}; \mu_{y,k}, \sigma_{y,k}\right)}{\sum_{k=1}^{N_k} \pi_{vk} \mathcal{N}\left(\lambda_v; \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \prod_{t=1}^{N_t} \mathcal{N}\left(y_v^{[t]}; \mu_{y,k}, \sigma_{y,k}\right)} \tag{5.114}$$

$$= \log p(\theta) + \sum_{v=1}^{N_v} l_{vk} \frac{\lambda_v - \mu_{\lambda,k}}{\sigma_{\lambda,k}^2} \tag{5.115}$$

$$\frac{\partial}{\partial \mu_{\lambda,k}} \log p(\theta|\boldsymbol{\lambda}, \boldsymbol{q}, \boldsymbol{y}, \gamma, \pi) = \frac{\partial}{\partial \mu_{\lambda,k}} \log p(\theta) + \sum_{v=1}^{N_v} l_{vk} \frac{\lambda_v - \mu_{\lambda,k}}{\sigma_{\lambda,k}^2} \tag{5.116}$$

differentiating (5.109) with respect to $\sigma_{\lambda,k}$, we obtain:

$$\frac{\partial}{\partial \sigma_{\lambda,k}} \log p(\theta|\boldsymbol{\lambda}, \boldsymbol{q}, \boldsymbol{y}, \gamma, \pi) = \frac{\partial}{\partial \sigma_{\lambda,k}} \log p(\theta) + \sum_{v=1}^{N_v} l_{vk} \frac{\mu_{\lambda,k}^2 - \sigma_{\lambda,k}^2 - 2\mu_{\lambda,k}\sigma_{\lambda,k} + \lambda_v^2}{\sigma_{\lambda,k}^3} \tag{5.117}$$

And differentiating with respect to $\mu_{y,k}$ and $\sigma_{y,k}$ we obtain:

$$\frac{\partial}{\partial \mu_{y,k}} \log p(\theta|\boldsymbol{\lambda}, \boldsymbol{q}, \boldsymbol{y}, \gamma, \pi) = \frac{\partial}{\partial \mu_{y,k}} \log p(\theta) + \sum_{v=1}^{N_v} l_{vk} \frac{y_v - \mu_{y,k}}{\sigma_{y,k}^2} \tag{5.118}$$

$$\frac{\partial}{\partial \sigma_{y,k}} \log p(\theta|\boldsymbol{\lambda}, \boldsymbol{q}, \boldsymbol{y}, \gamma, \pi) = \frac{\partial}{\partial \sigma_{y,k}} \log p(\theta) + \sum_{v=1}^{N_v} l_{vk} \frac{\mu_{y,k}^2 - \sigma_{y,k}^2 - 2\mu_{y,k}\sigma_{y,k} + y_v^2}{\sigma_{y,k}^3} \tag{5.119}$$

The expressions (5.116), (5.117), (5.118) and (5.119), though derived here to complete the derivations of the previous sections, will not be used in the following paragraphs. The problem of maximising

the posterior probability distribution of θ, given all other variables, in fact, is equivalent to the problem described in 4.A. Therefore updates of the parameters θ can be computed with the EM algorithm.

5.4 I Deformations from the emission data

The unified model described in the previous paragraph relates the time-dependent emission data to the MR data assuming the existence of finite anatomical-functional states and parameterising the spatial deformations. Modelling the statistical dependence between the time-dependence emission imaging, the MR imaging data and the deformation parameters, it eventually enables us to make use of the MR and PET data in order to estimate the deformation parameters jointly with the spatial density of the radio-pharmaceutical. In order to investigate applications of the unified model, let us break it down into simpler models. Each of the simpler models described in the next paragraphs could have been described independently. However we preferred the unified approach as it enables us to reason about the interaction of data obtained from multiple sources and to device new algorithms.

One may simply ignore subsets of the variables of the unified model of 5.2. Let us start by ignoring the finite labels $\mathbf{Z}$ and θ. Then $\boldsymbol{\lambda}$ and $\boldsymbol{y}$ become statistically independent. The model then expresses the interaction between the time-dependent emission data $\boldsymbol{q}$, the activity $\boldsymbol{\lambda}$ and the time-dependent deformation parameters γ. Fig. 5.3 reports the DAG of the model when $\mathbf{Z}$ is ignored. The variable $\boldsymbol{y}$ is not reported in the image as it is independent from $\boldsymbol{q}$, $\boldsymbol{\lambda}$ and γ.

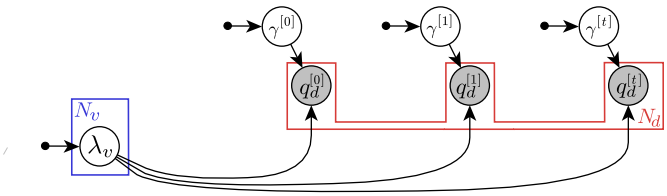


Figure 5.3: Deformations from the emission data: the model relates the time-dependent emission data $\boldsymbol{q}^{[t]}$ to the underlying radio-pharmaceutical density $\boldsymbol{\lambda}$ and to time-dependent spatial transformations $\gamma^{[t]}$. Fig. 5.4 reports the moralised graph.

The generative model of the imaging system, parametrised by the deformation parameters, allows us to estimate the motion parameters, along with the spatial concentration of the radio-pharmaceutical, from the emission data. In order to optimise the posterior probability distribution given the photon counts $p(\boldsymbol{\lambda}, \gamma | \boldsymbol{q})$, we decide once again to split the unknowns according to their natural separation

- $\mathcal{S}_1 : \boldsymbol{\lambda}$
- $\mathcal{S}_2 : \gamma$

and to alternate the optimisation of the posterior probability distribution of the activity $\boldsymbol{\lambda}$ conditional to γ and the optimisation of the posterior probability distribution of the deformation parameters γ conditional

to the activity (Iterated conditional modes - see paragraph 2.1.10). One may verify, from 5.4 and 5.5, that the posterior probability distribution of the motion parameters conditional to the activity factorises in N_t independent terms. Therefore the parameters for different time points are optimised independently.

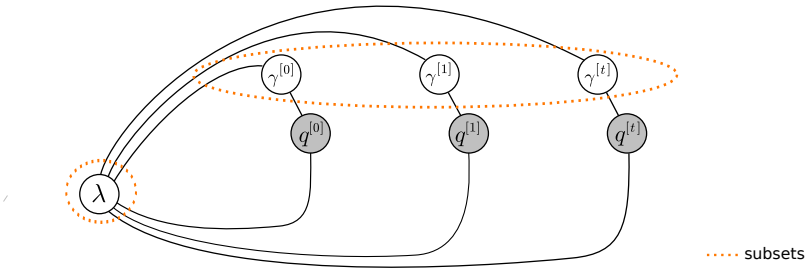


Figure 5.4: Deformations from the emission data: moralised graph of the model of Fig. 5.3. Given $\boldsymbol{\lambda}$ and $\boldsymbol{q}^{[t]}$, the deformation parameters at time t are independent from the deformation parameters at all other time points and from the photon counts at all other time points. Fig. 5.5 reports the Markov blankets of $\boldsymbol{\lambda}$ and γ for the alternated optimisation of the joint posterior of $\boldsymbol{\lambda}$ and γ.

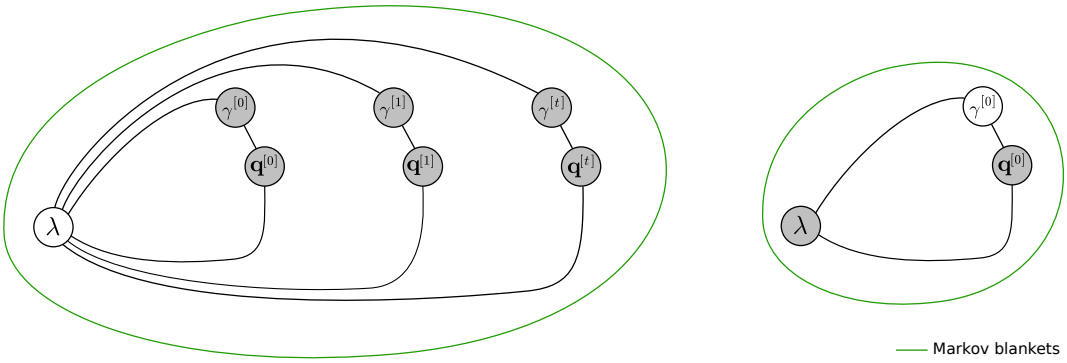


Figure 5.5: Deformations from the emission data: the posterior probability distribution of $\boldsymbol{\lambda}$, conditional to γ, depends on the photon counts at all time points and on the motion parameters at all time points. The posterior probability distribution of γ conditional to $\boldsymbol{\lambda}$, however, factorises in N_t terms (there is no open path connecting $\gamma^{[t]}$ to $\gamma^{[t']}$ when $\boldsymbol{\lambda}$ is given). Therefore, alternated optimisation of $Activity$ and γ consists in estimating the activity given all the deformation parameters, then estimating the deformation parameters independently, given the activity, as expressed by (5.120) and (5.122).

Equation (5.38) expresses the first derivative of the posterior probability distribution of the activity for the 4-D model of 5.3. The expression of equation (5.38) is the sum of N_t terms, each equivalent to the gradient of the likelihood of the static measurement (2.46). In order to optimise with respect of $\boldsymbol{\lambda}$, one may once again proceed by gradient ascent. Recognising how the two terms of (2.46) are rearranged in the MLEM update formula (3.14), the two terms of (5.38) can be rearranged to obtain the 4-dimensional version of the MLEM algorithm for the model of Fig. 5.3:

$$\lambda_v^{(n+1)} = \lambda_v^{(n)} \frac{1}{\sum_{t=1}^{N_t}\sum_{d=1}^{N_d} a_{rd}^{\gamma^{[t]}}} \sum_{t=1}^{N_t}\sum_{d=1}^{N_d} a_{rd}^{\gamma^{[t]}} \frac{q_d^{[t]}}{\sum_{v=1}^{N_v} a_{vd}^{\gamma^{[t]}} \lambda_v} \tag{5.120}$$

The optimisation of the motion parameters is to some extent simpler than the optimisation of the activity for two reasons. The first reason is that there are no negativity constraints. The second reason is that the dimensionality of γ is typically much smaller than that of $\boldsymbol{\lambda}$, enabling the use of sophisticated optimisation algorithms prohibited for $\boldsymbol{\lambda}$. The following section describes experiments with spatial transformations with 6 degrees of freedom. In such experiments, the motion parameters are optimised, at each step of the ICM algorithm, using the BroydenFletcherGoldfarbShanno (BFGS) quasi-Newton algorithm [84].

At every iteration, the BFGS algorithm makes use of all previous searching directions and of the first derivatives of the cost function in order to decide in which direction and how far to move in the space of the parameters. BFGS requires storage of all previous searching directions as well as many matrix vector multiplications, making it unsuited to problems with more than a few thousand variables, but very well suited to the experiments reported in the following section. In these experiments the BFGS algorithm implemented in the Matlab minFunc open source toolbox (`http://www.di.ens.fr/~mschmidt/Software/minFunc.html`) was used as a black-box optimisation tool, only requiring the first derivatives of the cost function. The first derivative of the posterior probability distribution of the motion parameters conditional to the activity, with respect to deformation parameter $\gamma_r^{[t]}$ have been derived in 5.3.2 (eq. (5.84) and (5.84)) and are reported here:

$$\frac{\partial}{\partial \gamma_r^{[t]}} \log p(\boldsymbol{q}^{[t]}|\boldsymbol{\lambda}, \gamma^{[t]}) = \tag{5.121}$$

$$\frac{\partial}{\partial \gamma_r^{[t]}} \log p(\gamma^{[t]}) + \sum_{d=1}^{N_d} \left[- \left[A \frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{\lambda}(T^{\gamma^{[t]}-1} X_v) \right]_d + q_d^{[t]} \frac{\left[A \frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{\lambda}(T^{\gamma^{[t]}-1} X_v) \right]_d}{\left[A \boldsymbol{\lambda}(T^{\gamma^{[t]}-1} X_v) \right]_d} \right] \tag{5.122}$$

where

$$A \frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{\lambda} \left(T^{\gamma^{[t]}-1} X_v \right) = A \left[\nabla \boldsymbol{\lambda} \left(T^{\gamma^{[t]}-1} X_v \right) \cdot \left(\frac{\partial}{\partial \gamma_r^{[t]}} T^{\gamma^{[t]}-1} \right) \cdot X_v \right] \tag{5.123}$$

As also expressed by the graph 5.5, the derivative with respect to motion parameter γ_r at time t does not depend on the motion parameters and on the photon counts in other time frames.

OSEM in time and space for the efficient estimation of the activity

The 4-D MLEM update formula differs from the static formula for the additional sum over t. The summations over d of the static MLEM update formula (3.14) are replaced, in the 4-D update formula (5.120), by summations over d and t. The Ordered Subsets Expectation Maximisation (OSEM) algorithm (see 3.1.3) can then be simply extended to the 4-D case. Each of the summations over all values of d and all values of t is replaced with the summation over subsets of d and t:

$$\lambda_v^{(n+1)} = \lambda_v^{(n)} \frac{1}{\sum_{t\in\mathcal{T}^{(n)}} \sum_{d\in\mathcal{D}^{(n)}} a_{rd}^{\gamma^{[t]}}} \sum_{t\in\mathcal{T}^{(n)}} \sum_{d\in\mathcal{D}^{(n)}} a_{rd}^{\gamma^{[t]}} \frac{q_d^{[t]}}{\sum_{v=1}^{N_v} a_{vd}^{\gamma^{[t]}} \lambda_v} \tag{5.124}$$

$$\mathcal{T}^{(n)} \subseteq \{1, 2, \ldots, N_t\} \tag{5.125}$$

$$\mathcal{D}^{(n)} \subseteq \{1, 2, \ldots, N_d\} \tag{5.126}$$

The updated value of the activity is therefore obtained in accordance to the photon counts observed in a subset of the detectors and in a subset of the time frames. The subsets can be chosen, at each iteration, according to a pre-defined scheme, hence the name *ordered subsets* given in [29], or randomly. In the experiments that follow, new subsets, in time and space, are chosen randomly at each iteration.

5.4.1 Rigid motion

Simple phantom

The first experiment points out how the effect of motion is more significant in emission imaging due to the photon limited regime of acquisition. Synthetic emission data was generated from the pattern of figure 5.6. The acquisition parameters used in the simulation were set to mimic the characteristics of a clinical 3-D PET scanner, but the activity was set to non-zero values only in a 3 mm thick 2-D slice perpendicular to the axis of the scanner. The simulated imaging volume was $300\ mm\ \times\ 300\ mm\ \times\ 3\ mm$. In order to emulate the level of noise typical of a 3-D medical acquisition, the number of detected photons for the 2-D experiment was set to the number of photons of a typical 3-D acquisition (50×10^6) divided by the number of axial slices of the 3-D system (100): 0.5×10^6 photons. Two different simulations were performed. In the first case, the rate of emission was considered to be constant during the entire acquisition and proportional to the grey levels of image 5.6-A. In the second case, the phantom was translated and rotated randomly 50 times, defining 50 time frames of equal duration. The range of motion and frequency of the motion events were deliberately exaggerated, drawing the random translations from a uniform distribution in the range $(-20\ mm\ \div\ 20\ mm)$ and the rotations from a uniform distribution in the range $(-12\ \deg \div 12\ \deg)$. With 50 time frames, the average photon counts for each time frame was equal to $0.5 \times 10^6/50 = 10^4$ photons. The activity of the first three time frames is reported in Fig. 5.6-A,B,C. Fig. 5.6-D reports the sum of the 50 phantoms. Fig. 5.7-A,B,C reports the sinograms corresponding to the first three time frames and 5.7-D the sinogram obtained by disregarding

the timing information (i.e. the sum of the time-indexed sinograms).

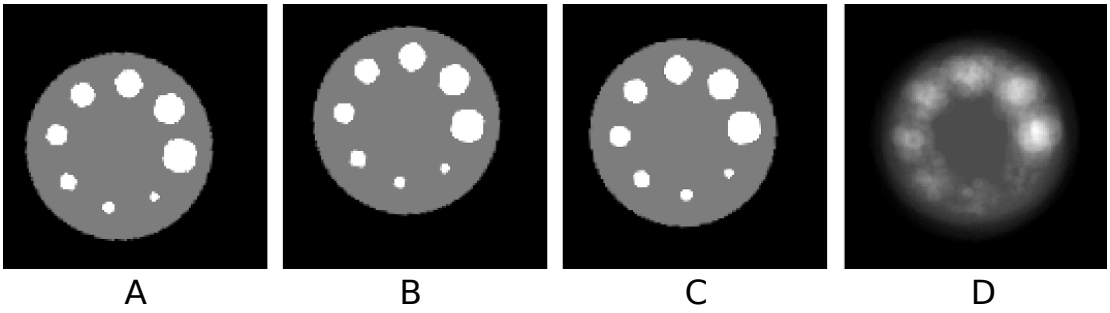


Figure 5.6: Rigid motion - synthetic activity. (A,B,C) Synthetic activity during time frames 1,2 and 3, respectively. (D) Sum of the activity during all $N_t = 50$ time frames.

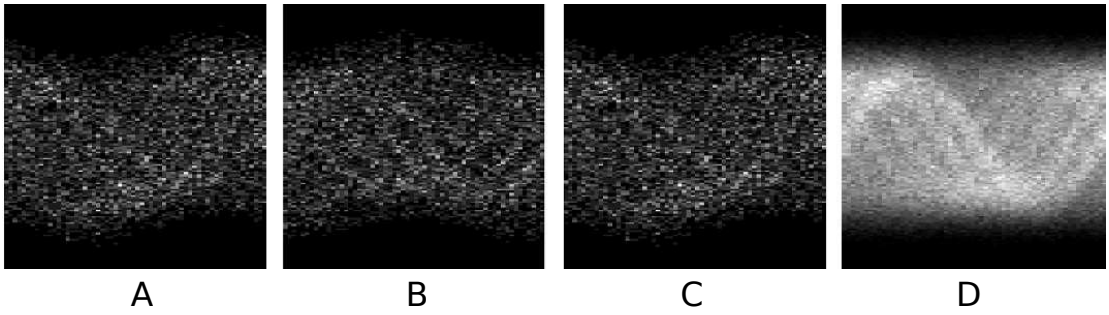


Figure 5.7: Rigid motion - sinograms. (A,B,C) Sinograms measured during time frames 1, 2 and 3, respectively. (D) Sinogram measured during the entire duration of the acquisition (sum of the $N_t = 50$ time-indexed sinograms).

Four reconstructions were performed:

- Reconstruction of the static measurement, MLEM (Fig. 5.8-A);
- Reconstruction of the motion-corrupted measurement, MLEM (Fig. 5.8-B);
- Frame-by-frame reconstruction, MLEM (Fig. 5.8-C);
- Joint reconstruction of the activity and of the motion parameters, ICM (Fig. 5.8-D);

No smoothness constraints were adopted in the reconstruction; however reconstructions were terminated after 160 iterations of MLEM. The early termination effectively introduces some degree of smoothness in the reconstructions. The 0.5×10^6 photons, as expected, were found to be sufficient to obtain an image of the phantom of reasonably good quality (Fig. 5.8-A). However, the information contained in a single time frame is not sufficient to reconstruct an image (Fig. 5.8-C). Joint reconstruction of activity and of the motion parameters was obtained by alternating, for 16 times, 10 iterations of MLEM and the optimisation of the motion parameters with the BFGS algorithm cycled until convergence. The MLEM step was implemented according to (5.120) and the gradient of the motion parameters is given

by (5.122) and (5.123). The transformation matrix was parametrised as described in (5.87),(5.88),(5.89), setting the derivatives with respect to $\gamma_2^{[t]}, \gamma_3^{[t]}, \gamma_6^{[t]}$ to zero.

Fig. 5.9 displays the recovery of the motion parameters as a function of the ICM iteration (light blue to dark blue) and the ground-truth motion parameters (red). The translation parameters were recovered within a few iterations of the ICM algorithm, however the rotation parameters were found to converge slowly. It was found that such problem, visible in 5.9-right, is solved by alternating the estimation of the translation and rotation parameters, as discussed in the note that follows. Fig. 5.8-D reports the activity obtained by the joint reconstruction of the activity and of the translation and rotation parameters (alternating the optimisation of translation and rotation). Remarkably, though the reconstructions of the single time frames present insufficient information for visual interpretation, a few iterations of ICM are sufficient to recover, from the multiple observations, the motion parameters and the underlying activity. The activity of Fig. 5.8-D, obtained by joint estimation of the activity and of the motion parameters, appears visually indistinguishable from the image obtained from the static experiment.

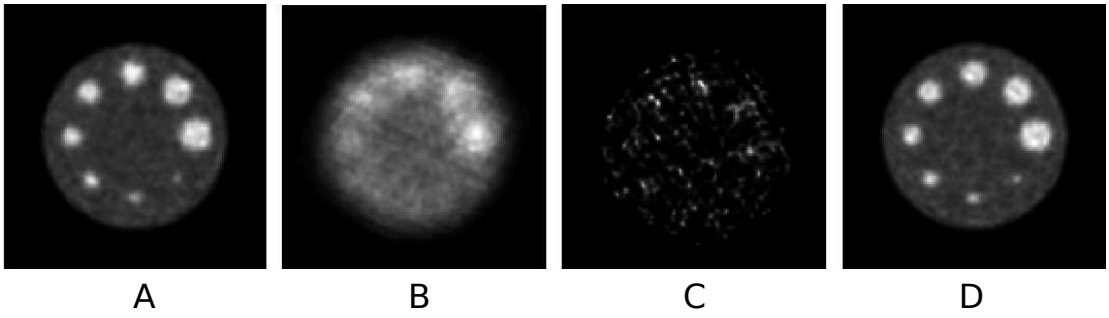


Figure 5.8: Rigid motion - reconstructions. (A) Reconstruction of the static measurement (MLEM). (B) Reconstruction of the motion-corrupted measurement (MLEM), - sinogram of Fig. 5.7-D. (C) Reconstruction of a single time frame (MLEM). (D) Joint reconstruction of the activity and of the motion parameters (ICM).

Fig. 5.11 and 5.10 are meant to aid the interpretation of the formulae of the derivatives of the posterior with respect to the motion parameters, reporting images of the quantities expressed by (5.122) and (5.123).

Note on the selection of the subsets

It was found that, when alternating the optimisation of the activity and of the motion parameters, the rate of convergence of the rotation parameters is much slower than for the translation parameters. Such problem is related to the different order of magnitude of the gradients of the translation and rotation parameters. The step size, selected by the line search algorithm, appears to be limited by the translation parameters. Alternating the optimisation of the activity, the optimisation of the translation parameters and the optimisation of the rotation parameters, convergence is faster for all the parameters.

One may think that joint optimisation of the parameters of a model is in general a better idea than

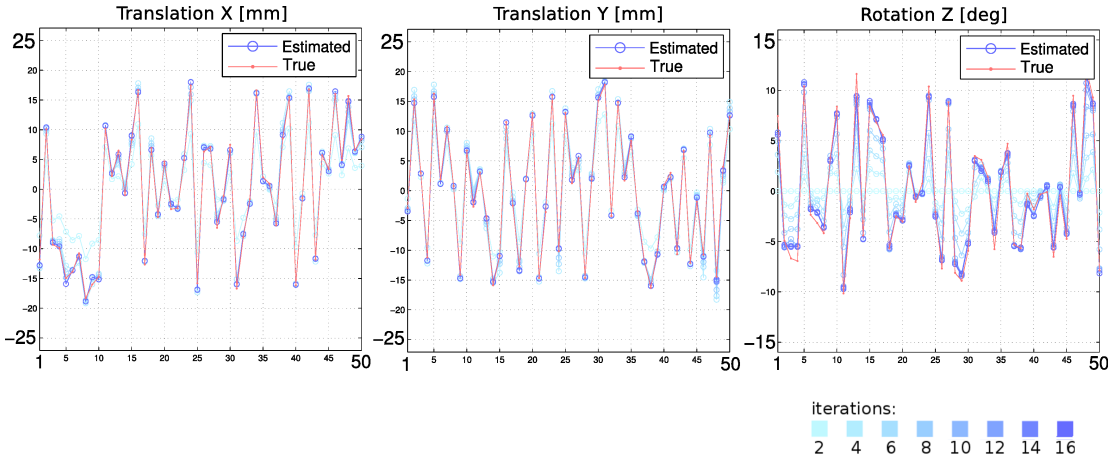


Figure 5.9: Rigid motion - recovery of the motion parameters. The plots report the true motion parameters (red) and the estimated motion parameters (blue) for each of the 50 time points. The estimated motion parameters are pictured as a function of the ICM iteration: light blue indicates low iteration number, dark blues high iteration number (up to 16). The rotation parameter converges slowly, eventually never reaching convergence. Such problem is solved by alternating the optimisation of the translation and rotation parameters, as discussed in the note of 5.4.1. The translation parameters are recovered accurately in just a few iterations of ICM.

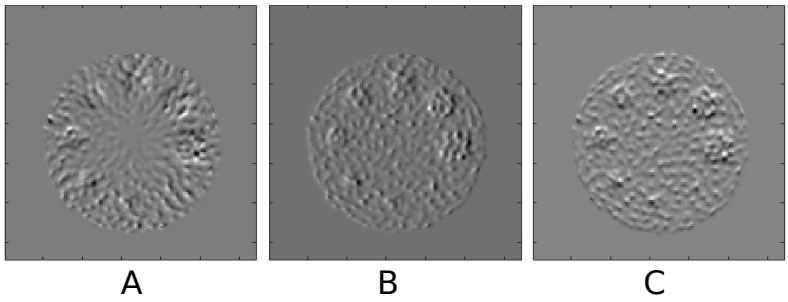


Figure 5.10: Rigid motion - derivatives with respect to the motion parameters in image space. The three images display the quantity obtained by evaluating the expression in the square brackets of (5.123) at ICM iteration 10 for the three rigid motion parameters: (A) rotation, (B) translation X, (C) translation Y. See also 5.11.

alternated optimisation, considering the latter a simpler alternative for certain problems. Alternated optimisation, in fact, may eventually converge to saddle points of the cost function, as discussed in [85]. However, alternated optimisation may be a better choice when the joint cost function is affected by the parameters to a largely different extent. Partitioning of the variables that appear to be natural, such as for example grouping the rates of emission in all voxels into a subset and the translation parameters into another subset, indeed often correspond to the selection of sets of variables that present commensurate derivatives.

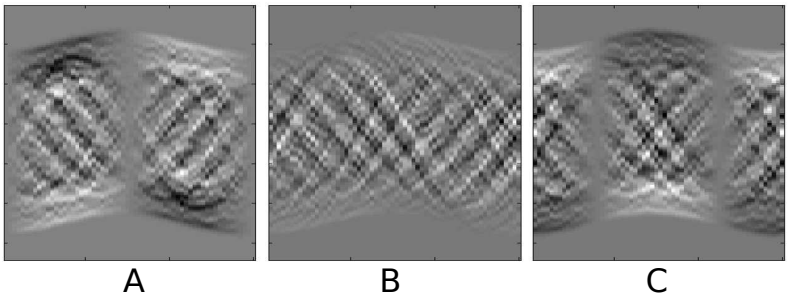


Figure 5.11: Rigid motion - derivatives with respect to the motion parameters in detector space. The three images display the quantity obtained by evaluating (5.123) at ICM iteration 10 for the three rigid motion parameters: (A) rotation, (B) translation X, (C) translation Y. The three images therefore represent the projections of the images in 5.10. As expressed by (5.122)-right, the derivative of each motion parameter is obtained by performing a sum over all the elements of the corresponding image (multiplied by the photon counts and by the projection of the activity estimate).

The experiment that follows, which evaluates the joint estimation of the activity and the rigid motion parameters in the case of fully three-dimensional imaging, makes use of the alternated optimisation of the translation and rotation parameters. Fig. 5.15 will show how the translation and rotation parameters converge at equal rates.

Brain phantom

The ICM algorithm (5.120), (5.122), (5.123), at the first iteration, reconstructs the activity given the initial estimates of the motion parameters. If the motion (translation and rotation) parameters are initially considered to be zero at all time frames, the estimate of the activity obtained at the first iteration of ICM is equal to the reconstruction obtained by ignoring motion and thus summing the time-indexed sinograms into a single sinogram. For the experiment of the previous section, the first estimate of the activity then corresponds approximately to a smoothed version of 5.8-B (as in the experiment MLEM is iterated only 10 times within each ICM step). The second part of the first step of ICM is to estimate the motion parameters. Such optimisation essentially consists in moving around the estimate of the activity, until its projection matches the measurements. One may think that, starting with such a poor initial estimate of the activity, it would be difficult to make a good guess of the motion parameters. However, surprisingly, the motion parameters and the activity are recovered in just a few iterations of ICM. One may argue that the simplicity of the object is the cause of the good behaviour of the optimisation. This second experiment considers a synthetic brain imaging study in the three-dimensional imaging space a 6 DOF rigid motion.

The synthetic activity and the characteristics of the PET imaging system are the same that were used in the experiments described in 4.5. The activity phantom was transformed, according to random translations and rotations, into 50 volumetric images. Translations were sampled from a uniform dis-

tribution in the range $(-15\ mm\ \div\ 15\ mm)$ and rotations from a uniform distribution in the range $(-5\ \text{deg} \div 5\ \text{deg})$. Fig. 5.12(A,B,C) reports the central transverse slice of the activity phantom relative to time frames 1,2 and 3 and Fig. 5.12-D reports the central transverse slice the sum of the 50 volumes. The 50 frames were considered to be of equal duration, with the collection of 50×10^6 photons, 1×10^6 per frame. Fig. 5.13 reports one plane of the viewgrams (i.e. the emission data re-binned from the cylindrical geometry to the planar geometry) relative to time frames 1,2,3 and to all frames (D).

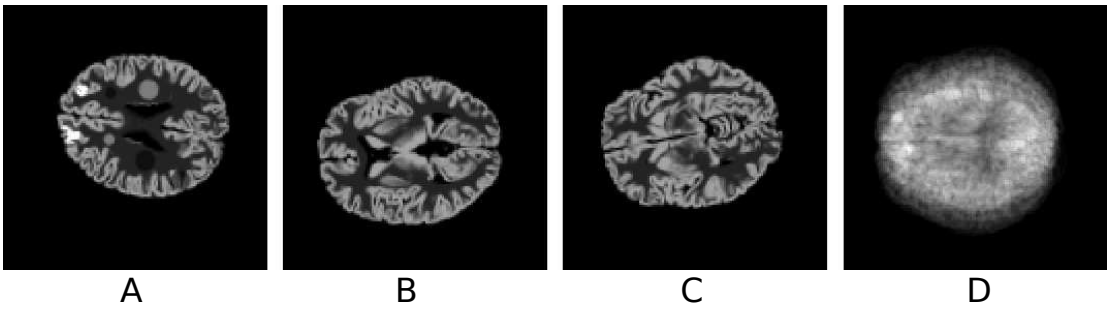


Figure 5.12: Rigid motion - phantom. (A,B,C) Activity phantoms at time frames 1,2 and 3, respectively. The phantoms are obtained by random translation and rotation of a static 3-D synthetic spatial concentration of the pharmaceutical. (D) Sum of the 50 phantoms.

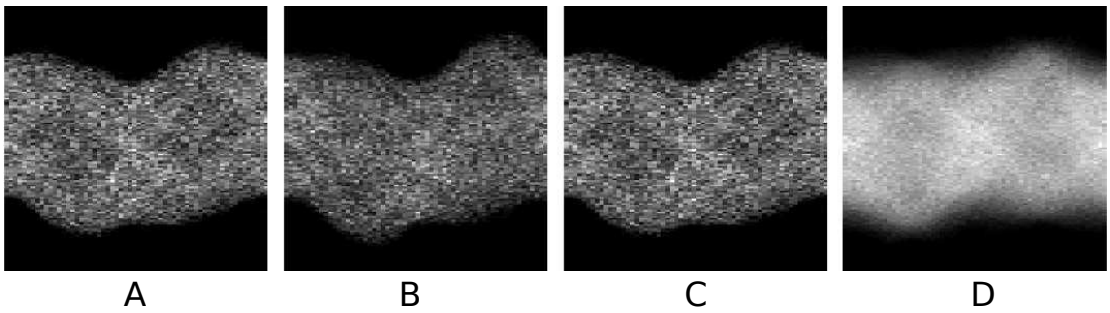


Figure 5.13: Rigid motion - viewgrams. (A,B,C) Viewgrams at time frames 1,2 and 3, respectively. (D) Sum of all 50 viewgrams. This is the measurement obtained by discarding the timing information.

As in the previous experiment, no smoothness constraints were adopted in the reconstruction, however reconstructions were terminated after 160 iterations of MLEM. Joint reconstruction of activity and of the motion parameters was obtained by alternating, for 16 times, 10 iterations of MLEM and the optimisation of the motion parameters with the BFGS algorithm cycled until convergence. The transformation matrix was parametrised as described in (5.87),(5.88),(5.89).

Fig. 5.14-D reports the activity obtained by the joint reconstruction of the activity and of the 6 motion parameters and Fig. 5.15 displays the recovery of the motion parameters as a function of the ICM iteration. The motion parameters are recovered in 15 iterations of ICM, with residual average errors of $0.4\ mm$ ($\frac{1}{4}$ of a voxel) and $0.65\ \text{deg}$. The activity obtained by joint estimation of the activity

and of the motion parameters, though somewhat distinguishable from the image obtained from the static experiment, is well recovered.

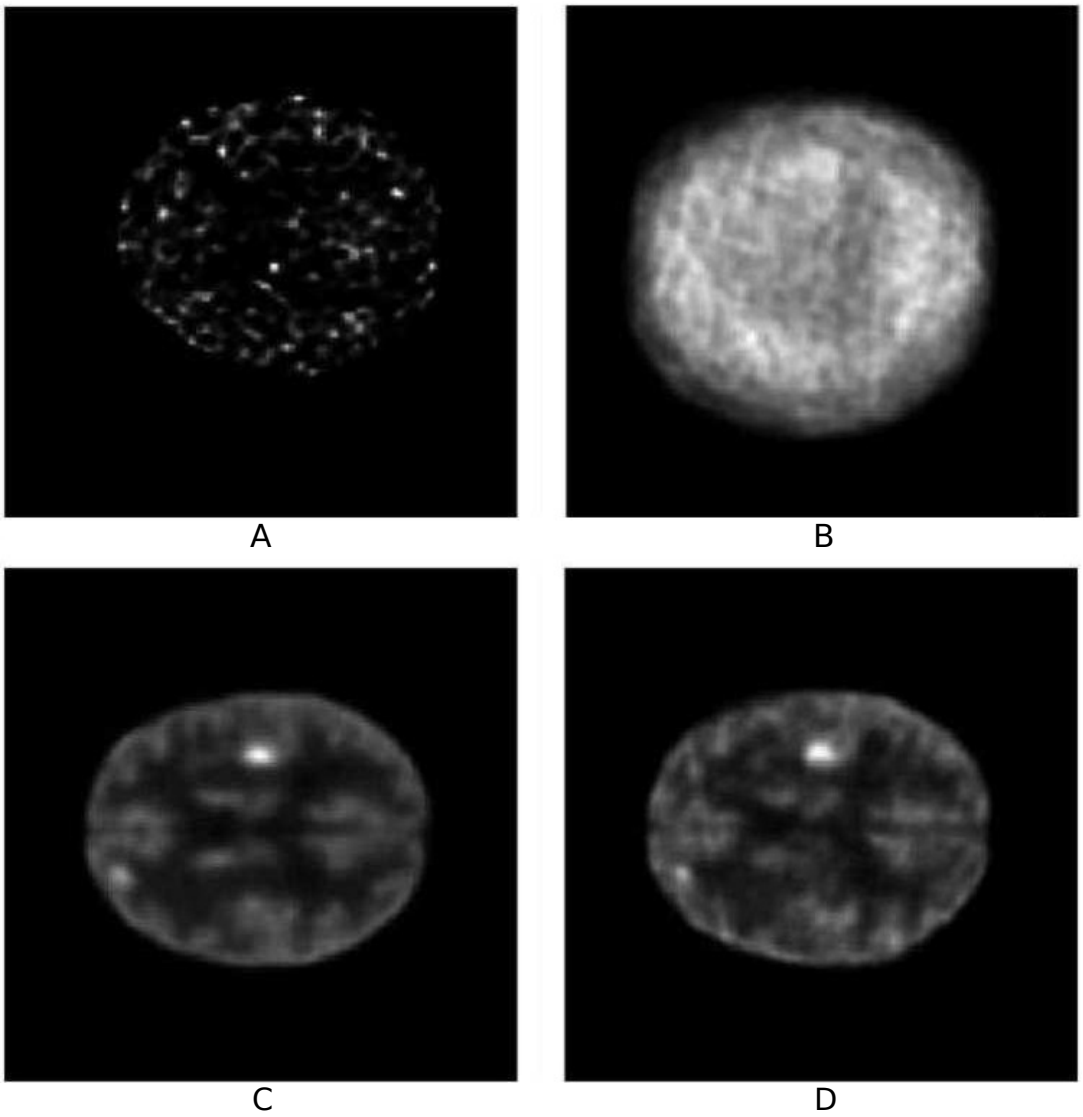


Figure 5.14: Rigid motion - reconstructions. (A) Reconstruction of a single time frame (MLEM). (B) Reconstruction of the motion-corrupted measurement (MLEM), - sinogram of Fig. 5.13-D. (C) Reconstruction of the motion-free measurement (MLEM). (D) Joint reconstruction of the activity and of the motion parameters (ICM).

5.4.2 Non-rigid motion

The gradient of the non-rigid motion parameters may be obtained with the same formulae described in the previous paragraph, just by replacing the transformation matrix. Such problem has not been explored in this work, however, given the effectiveness of gradient-based estimation of the rigid motion parameters from the emission data, highlighted by the experiments in the previous section, estimation of the non-

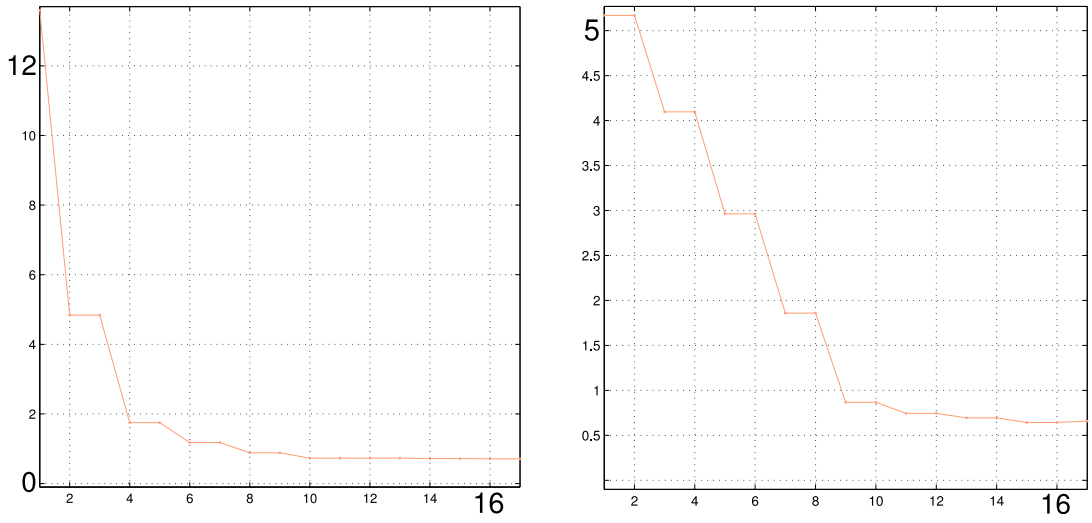


Figure 5.15: Rigid motion - recovery of the motion parameters. The plots display the root mean squared error (RMSE) of the translation and rotation parameters. RMSE is obtained by summing over the three translation and rotation parameters and over all time points. The convergence of the rotation parameters was found to be slightly slower than for the translation parameters. Translation and rotation converge within 15 iterations, with residual RMSE of $0.4\ mm$ ($\frac{1}{4}$ of a voxel) and $0.65\ \deg$

rigid deformation parameters directly from the emission data becomes an interesting problem. From the algorithmic perspective, the main challenge with the estimation of the non-rigid deformation parameters from the emission data is that, in order to compute the derivatives of the deformation parameters, it is required to perform one projection for each parameter.

5.5 II 4-D reconstruction with known deformations

The model described in the previous paragraph enables the estimation of the motion parameters from the emission imaging data. The experiments described in 5.4.1 show the feasibility of direct estimation of the rigid motion parameters for complex objects and with a higher number of time frames. Paragraph 5.4.2 discusses the extension to non-rigid deformations. While the update formulae (5.120), (5.122) and (5.123) are unchanged in the non-rigid case, the calculation of the derivatives of the motion parameters becomes computationally expensive, involving one projection for each deformation parameter.

In the experiments reported in 5.4.2 it was assumed that no information was available beforehand about the motion parameters, assigning the uninformative uniform distribution to each of the motion parameters. If motion parameters are measured by an external motion detection device, the prior probability distribution $p(\gamma)$ may express the acquired knowledge. In this paragraph we will consider the case in which the prior probability distribution over the motion parameters is a delta function, that is with an ideal motion detection device. The graphical model is then simply equal to the graph described

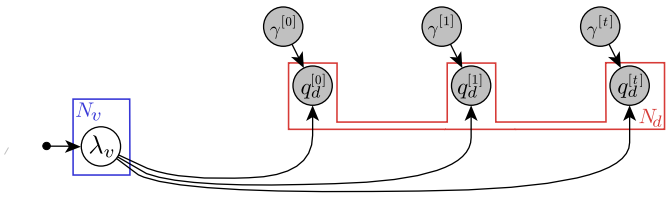


Figure 5.16: 4-D reconstruction with known deformations: the motion parameters γ are assumed to be known exactly. The only unknown is the activity $\boldsymbol{\lambda}$.

in the previous paragraph, with shaded nodes for the motion parameters (Fig. 5.16). The gradient of the probability distribution of $\boldsymbol{\lambda}$ is expressed by (5.38) and the EM update formula for the activity is expressed by (5.120). While the estimation of the motion parameters from the emission data presents the computational challenge associated with the calculation of the derivatives of the motion parameters, the estimation of the activity with known deformation parameters does not present a tremendous computational challenge. Let us rewrite the 4-D MLEM formula according to the expression of the gradient (5.54):

$$\lambda_v^{(n+1)} = \lambda_v^{(n)} \frac{1}{\sum_{t=1}^{N_t} \tau_{\gamma^{[t]}}^T P^T \mathbf{1}} \sum_{t=1}^{N_t} \tau_{\gamma^{[t]}}^T P^T \frac{\boldsymbol{q}^{[t]}}{P\tau_{\gamma^{[t]}} \boldsymbol{\lambda}} \tag{5.127}$$

The normalisation $\sum_{t=1}^{N_t} \tau_{\gamma^{[t]}}^T P^T \mathbf{1}$ can be pre-computed; then each MLEM update involves the following steps:

1. Transform the estimate of the activity according to the N_t sets of deformation parameters $\tau_{\gamma^{[t]}}$;
2. Project the N_t transformed activity images;
3. Divide (point-wise) each sinogram by the projection of the transformed activity;
4. Back-project each of the N_t detector space functions from step (3);
5. Transform each back-projection according to the inverse of the transformation used in (1);
6. Sum the N_t inverse transformed back-projections;
7. Multiply the provisional estimate of the activity by the volume obtained in step (5) and divide by the normalisation.

The full MLEM update may be replaced by the OSEM update reported in (5.124).

5.5.1 Non-rigid motion

The NiftyRec software (`http://niftyrec.scienceontheweb.net/`), developed along with this thesis and described in Chapter 10, was designed to enable 4-D reconstruction. The image support can be warped according to deformation parameters. The locations of the voxels X_v are parametrised by means of a b-spline model, which expresses locations of the voxels as a function of the location of

the control points of the splines. The control points, displaced on a 3-dimensional lattice, provide a low dimensional parametrisation of the spatial deformations, the position of each node controlling locally the location of several voxels X_v.

In the experiments that follow, synthetic deformations were generated according to a sinusoidal pattern emulating breathing motion. The breathing ferquency was considered to be of 5 breaths per minute and was sampled 10 times per cycle. A synthetic static phantom of the radio-activity, made of 8 hot spheres of varying radius in uniform background, was deformed according to such pattern of motion, obtaining a sequence of 50 images. Two experiments were performed. In the first experiment, the static phantom was imaged for $5\ min$, with an expected rate of detection of 10×10^6 photons pairs per minute, for a total of approximately 50×10^6 counts. In the second experiment 25 repetitions of the breathing cycle (5 per minute for $5\ min$) were simulated, obtaining 250 sinograms, each of duration $1.2\ sec$ and with 0.2×10^6 average detected events.

Fig. 5.17 reports viewgrams, from 5 different directions, of the static measurement; Fig. 5.18 reports the noiseless viewgrams for the dynamic experiment and Fig. 5.19 reports the noisy viewgrams for the dynamic experiment. It appears, from the images of the viewgrams, that while in absence of motion one can make sense of the measurement, the single frame does not have enough information to obtain a tomographic image (Fig. 5.19).

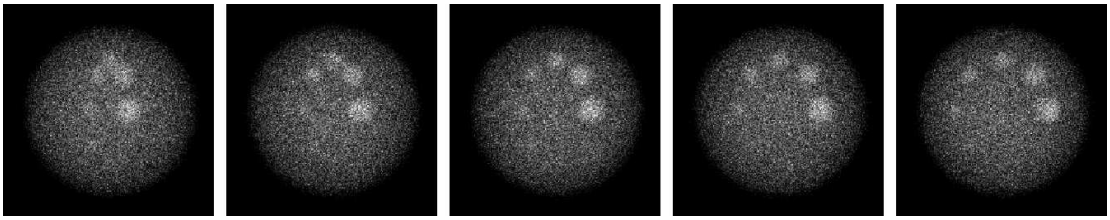


Figure 5.17: Non-rigid motion - motion-free viewgrams. The viewgrams are displayed from 5 different directions. The acquisition parameters emulate a $5\,\mathrm{min}$ PET scan, with a total of 50×10^6 counts. With such level of noise, one may recognise the structure of the underlying image.

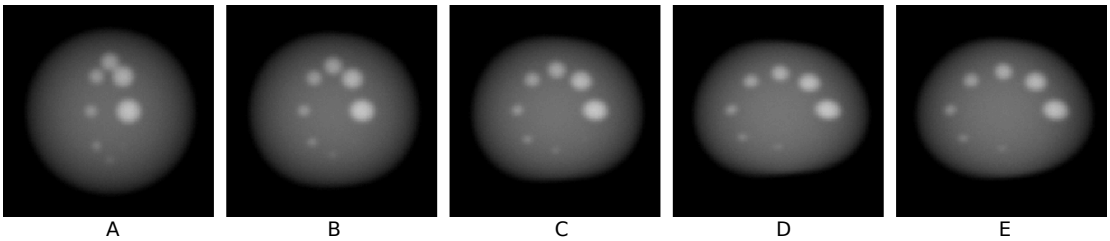


Figure 5.18: Non-rigid motion - ideal viewgrams. The noiseless viewgrams, displayed from different directions, are obtained by projecting the activity from 5 of the 10 samples of the breathing cycle (odd-indexed time frames). Fig. 5.19 displays the corresponding photon counts.

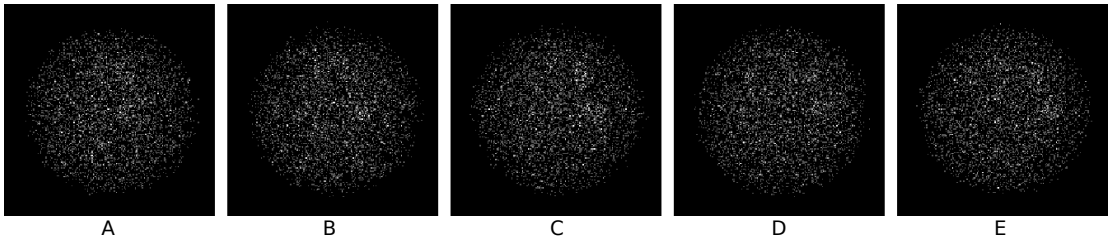


Figure 5.19: Non-rigid motion - noisy viewgrams. The viewgrams represent the photon counts observed (from different directions), during different time frames. (A) time frame 1; (B) time frame 3; (C) time frame 5; (D) time frame 7; (E) time frame 9. The structure of the underlying object is unrecognizable. Frame-by-frame reconstructions are reported in Fig. 5.22.

Four reconstructions were performed:

- Reconstruction of the static measurement, OSEM (Fig. 5.20);
- Reconstruction of the motion-corrupted measurement, OSEM (Fig. 5.21);
- Frame-by-frame reconstruction, OSEM (Fig. 5.22);
- 4-D reconstruction, 4-D OSEM (Fig. 5.23);

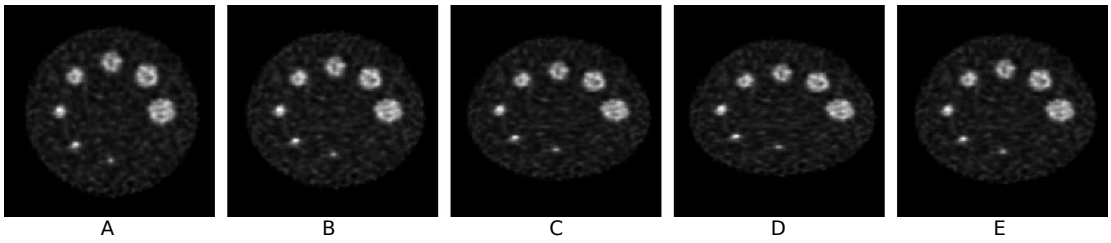


Figure 5.20: Non-rigid motion - (deformed) motion-free reconstruction. The motion-free reconstruction was warped according to the deformations parameters. (A,B,C,D,E) display the reconstruction deformed according to time frames 1,3,5,7,9. This visualisation serves to visualise the deformation parameters and as ground truth for visual evaluation of the 4-D MLEM reconstruction (Fig. 5.23).

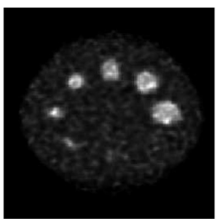


Figure 5.21: Non-rigid motion - motion-corrupted static reconstruction. This image is obtained reconstructing a single static image for the $5\ min$ scan, while the activity deforms due to breathing.

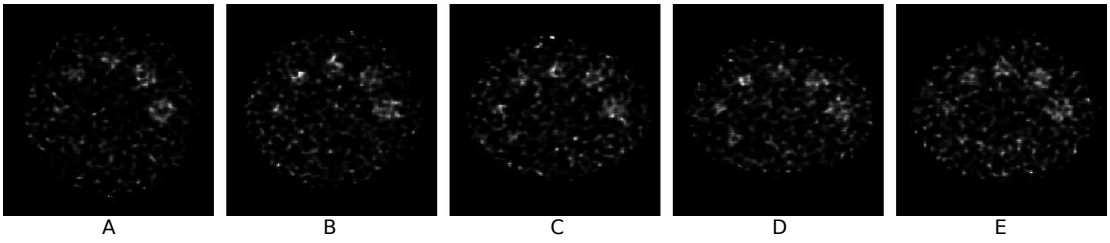


Figure 5.22: Non-rigid motion - frame-by-frame reconstruction. (A,B,C,D,E) correspond to time frames 1,3,5,7,9.

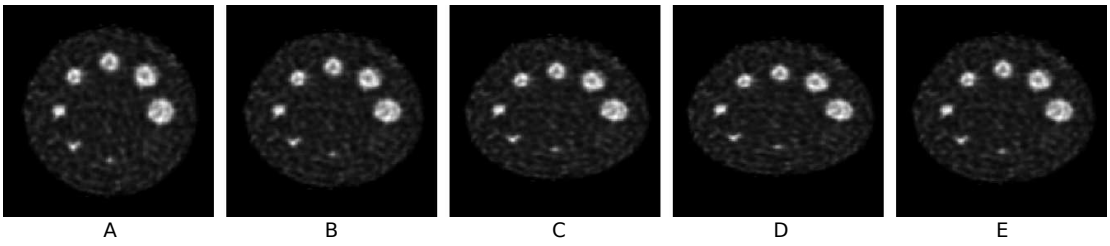


Figure 5.23: Non-rigid motion - 4-D reconstruction. (A,B,C,D,E) correspond to time frames 1,3,5,7,9.

All reconstructions were obtained with 160 iterations of OSEM with subsets of size $1/16$-th of the total number of detector bins. The 4-D reconstruction was obtained with 160 iterations of OSEM with subsets of size $1/16$-th of the total number of detectors from each time bin and selecting $1/16$-th of the time bins at each iteration (spatial and temporal OSEM - see paragraph 3.1.3) Fig. 5.24 displays a close up of the reconstructions obtained with the four methods for visual comparison (central slice of the imaging volume). The 4-D reconstruction presents small artifacts due to the approximation involved in the inversion of the spatial transformations (the b-spline model is non-invertible).

Computational complexity

The imaging volume was $192 \times 192 \times 192$ voxels and the detector bins $192 \times 192 \times 180$. With 16 spatial subsets and 16 temporal subsets, each step of 4-D MLEM required $\frac{250}{16 \times 16} \approx 1$ full projections and back-projections. The deformations were parametrised with a grid of $20 \times 20 \times 20$ nodes. The reconstruction, terminated after 160 iterations, require in total $4.3\ min$ on a CPU Xeon E5430@2.66GHz quad-core with GPU NVidia GTX285. In case of repetitive motion, multiple measurements relative to nearby locations in the motion cycle may be binned as one in order to reduce the computational complexity.

5.6 III Anatomically-constrained 4-D reconstruction

When all the variables of the joint model of Fig. 5.2 are considered at once, the joint optimisation can be addressed with the ICM algorithm, with the following subsets:

- $\mathcal{S}_1 : \boldsymbol{\lambda}$
- $\mathcal{S}_2 : \theta = \{\theta_y, \theta_{\boldsymbol{\lambda}}\}$

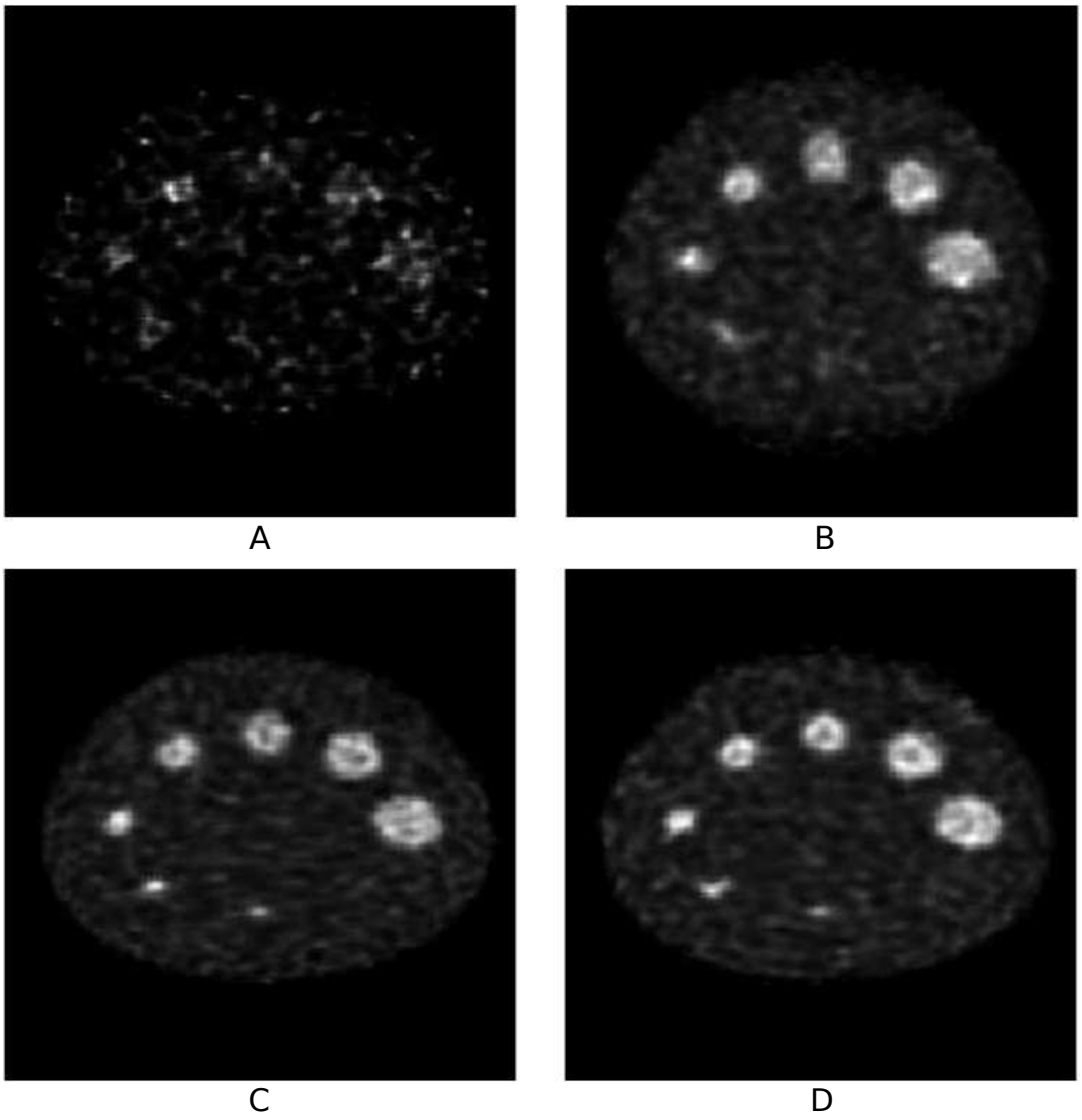


Figure 5.24: Non-rigid motion - comparison. (A) frame-by-frame reconstruction, frame 9; (B) motion-corrupted static reconstruction; (C) motion-free reconstruction deformed according to the parameters relative to frame 9; (D) 4-D reconstruction deformed according to the parameters relative to frame 9.

- $\mathcal{S}_3 : \gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_{N_t}\}$

and, in case of rigid motion, the motion parameters γ can be conveniently further separated, into translation and rotation parameters, as discussed in 5.4. Denoting with $\mathcal{S}_0$ observed variables (MRI images, photon counts, known model parameters) ICM then consists in alternating three optimisations:

- $\boldsymbol{\lambda}^{(n+1)} = \operatorname{argmax}_{\boldsymbol{\lambda}} \log p(\boldsymbol{\lambda}|\boldsymbol{q}, \boldsymbol{y}, \gamma, \theta, \pi)$
- $\gamma^{(n+1)} = \operatorname{argmax}_{\gamma} \log p(\gamma|\boldsymbol{q}, \boldsymbol{y}, \boldsymbol{\lambda}, \theta, \pi)$
- $\theta^{(n+1)} = \operatorname{argmax}_{\theta} \log p(\theta|\boldsymbol{\lambda}, \boldsymbol{q}, \boldsymbol{y}, \gamma, \pi)$

The gradients of the three cost functions have been derived in 5.3. While, once again, each maximisation (as well as the joint maximisation, without recurring to ICM) could be performed by gradient ascent, it is preferred to use EM type algorithms.

The maximisation 1. with respect to $\boldsymbol{\lambda}$ has been partly discussed in 5.4, which reports the MLEM update formula for $\boldsymbol{\lambda}$, given the motion parameters γ. However the derivation of the MLEM update formula in 5.4 did not account for the hidden variables $\mathbf{Z}$, which relate, through the parameters θ, the activity $\boldsymbol{\lambda}$ to the MRI images $\boldsymbol{y}$. Considering the posterior distribution of $\boldsymbol{\lambda}$ for the full joint model (5.35), one can verify that there is no longer a closed form expression for the M-step of the EM algorithm for the maximisation with respect to $\boldsymbol{\lambda}$. Proceeding like in 4.3, let us adopt the OSL algorithm:

$$\lambda_v^{(n+1)} = \lambda_v^{(n)} \frac{1}{\sum_{t\in\mathcal{T}^{(n)}} \sum_{d\in\mathcal{D}^{(n)}} a_{rd}^{\gamma^{[t]}} - \frac{\partial}{\partial\lambda_v} \log p(\boldsymbol{\lambda}|\boldsymbol{y},\gamma,\theta,\pi)\|_{\boldsymbol{\lambda}^{(n)}}} \sum_{t\in\mathcal{T}^{(n)}} \sum_{d\in\mathcal{D}^{(n)}} a_{rd}^{\gamma^{[t]}} \frac{q_d^{[t]}}{\sum_{v=1}^{N_v} a_{vd}^{\gamma^{[t]}} \lambda_v} \tag{5.128}$$

with

$$\frac{\partial}{\partial\lambda_v} \log p(\boldsymbol{\lambda}|\boldsymbol{y},\gamma,\theta,\pi) = \sum_{k=1}^{N_k} l_{vk}^{(n+1)} \frac{\mu_{\lambda,k} - \lambda_v}{\sigma_{\lambda,k}^2} \tag{5.129}$$

and

$$l_{vk}^{(n+1)} = \frac{\pi_{vk} \mathcal{N}\left(\lambda_v, \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \prod_{t=1}^{N_t} \mathcal{N}\left(y_v^{[t]}, \mu_{y,k}, \sigma_{y,k}\right)}{\sum_{k=1}^{N_k} \pi_{vk} \mathcal{N}\left(\lambda_v, \mu_{\lambda,k}, \sigma_{\lambda,k}\right) \prod_{t=1}^{N_t} \mathcal{N}\left(y_v^{[t]}, \mu_{y,k}, \sigma_{y,k}\right)} \tag{5.130}$$

The maximisation 2. with respect to the motion parameters has been partly discussed in 5.4, where it has been addressed using a quasi-Newton optimisation algorithm. Having discarded the hidden discrete labels, in 5.4, the gradient of the log posterior distribution of the motion parameters, expressed by (5.122), was a function of the activity and of the photon counts only. The gradient of the log posterior distribution of the motion parameters for the full joint model has been derived in 5.3.2 and comprises an emission term and a finite mixture term. Let us repeat it here, to specify the algorithm in full:

$$\frac{\partial}{\partial \gamma_r^{[t]}} \log p(\gamma|\boldsymbol{q},\boldsymbol{y},\boldsymbol{\lambda},\theta,\pi) = \tag{5.131}$$
$$\underbrace{\sum_{d=1}^{N_d}\left[-\left[A\frac{\partial}{\partial \gamma_r^{[t]}}\boldsymbol{\lambda}(T^{\gamma^{[t]}-1}X_v)\right]_d + q_d^{[t]}\frac{\left[A\frac{\partial}{\partial \gamma_r^{[t]}}\boldsymbol{\lambda}(T^{\gamma^{[t]}-1}X_v)\right]_d}{\left[A\boldsymbol{\lambda}(T^{\gamma^{[t]}-1}X_v)\right]_d}\right]}_{\text{Emission term}} +$$
$$\underbrace{\sum_{v=1}^{N_v}\left[\frac{\partial}{\partial \gamma_r^{[t]}}\boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}}X_v\right)\sum_{k=1}^{N_k}\iota_{vk}\frac{\mu_{y,k}-\boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}}X_v\right)}{\sigma_{y,k}^2}\right]}_{\text{Finite mixture term}}$$

with

$$\frac{\partial}{\partial \gamma_r^{[t]}}\boldsymbol{\lambda}\left(T^{\gamma^{[t]}-1}X_v\right) = \left[\nabla\boldsymbol{\lambda}\left(T^{\gamma^{[t]}-1}X_v\right)\cdot\left(\frac{\partial}{\partial \gamma_r^{[t]}}T^{\gamma^{[t]}-1}\right)\cdot X_v\right] \tag{5.132}$$

and

$$\frac{\partial}{\partial \gamma_r^{[t]}}\boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}}X_v\right) = \nabla\boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}}X_v\right)\cdot\left(\frac{\partial}{\partial \gamma_r^{[t]}}T^{\gamma^{[t]}}\right)\cdot X_v \tag{5.133}$$

The maximisation 3. with respect to the parameters θ when the activity $\boldsymbol{\lambda}$ and the motion parameters γ are given, corresponds again to the estimation of the parameters of a multi-variate Gaussian mixture with given observations. The problem is analogous to 4.3, where the dimensionality of the mixture model was 2. Here the dimensionality of the mixture model is $N_t + 1$. The EM update formulae have the following expressions:

$$\mu_{y,k}^{(n+1)} = \frac{1}{N_t\sum_{v=1}^{N_v} l_{v,k}^{(n+1)}}\sum_{t=1}^{N_t}\sum_{v=1}^{N_v} l_{v,k}^{(n+1)}\boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}}X_v\right) \tag{5.134}$$

$$\sigma_{y,k}^{(n+1)} = \left[\frac{1}{N_t\sum_{v=1}^{N_v} l_{v,k}^{(n+1)}}\sum_{t=1}^{N_t}\sum_{v=1}^{N_v} l_{v,k}^{(n+1)}\left(\mu_{y,k}^{(n+1)}-\boldsymbol{y}^{[t]}\left(T^{\gamma^{[t]}}X_v\right)\right)^2\right]^{\frac{1}{2}} \tag{5.135}$$

$$\mu_{\lambda,k}^{(n+1)} = \frac{1}{\sum_{v=1}^{N_v} l_{v,k}^{(n+1)}}\sum_{v=1}^{N_v} l_{v,k}^{(n+1)}\lambda_v \tag{5.136}$$

$$\sigma_{\lambda,k}^{(n+1)} = \left[\frac{1}{\sum_{v=1}^{N_v} l_{v,k}^{(n+1)}}\sum_{v=1}^{N_v} l_{v,k}^{(n+1)}\left(\mu_{\lambda,k}^{(n+1)}-\lambda_v\right)^2\right]^{\frac{1}{2}} \tag{5.137}$$

5.6.1 Event-driven 4-D PET-MR

In the context of PET-MR we propose an event-driven motion compensation algorithm based on the acquisition of an MRI image each time large motion is detected by a tracking system or by an MRI navigator, or at regular intervals throughout the PET acquisition process. The 4-D joint generative model allows us to estimate motion parameters from the MRI image and from the photon counts, obtaining a time consistent estimate of activity that accounts for its relation with the underlying anatomy, imaged by MRI.

Noisy simulation data modelling 3-D patient head movements were constructed by projecting at various orientations a brain phantom based on the BrainWeb database [86] and by warping the corresponding MRI image. Rigid motion was simulated by randomly generating rotation and translation along the three axes, according to amplitudes and frequency that resemble typical measurements of head movement within a scanner for Emission Tomography [76] (Fig. 5.25). The MRI and functional imaging processes were decoupled by running independent simulations based on the ground truth normal brain tissue model from Brain-Web.

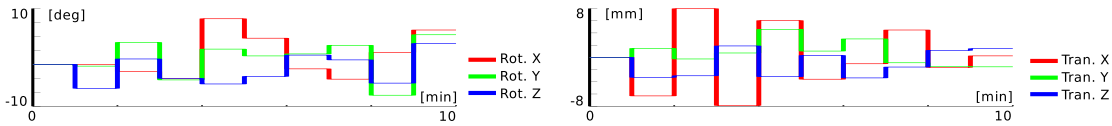


Figure 5.25: Simulated random rigid motion parameters. Rotation in degrees (left) and translation im mm (right) along the three axes.

The MRI image was generated with the BrainWeb simulator, which realistically accounts for noise of the imaging system. The parameters of the simulator were set for T1-weighted imaging with noise standard deviation set at 3% of the brightest tissue and perfect uniformity of the magnetic field (in accordance with the simplistic GM model). Activity of 18F-FDG was simulated by associating typical activity levels to different tissue types, proportionally to partial voxel occupation. Specifically the activity in grey matter was set to a value 4 times higher than in all other tissues. The total number of counts was set to 32×10^6. A 1.9×10^6 LOR PET imaging system was simulated by means of the NiftyRec rotation-based projector with realistic position dependent point source response and uniform attenuation within the area of the head. The effect of scatter and randoms was disregarded, at this stage, by simulating only the non-scattered photon pairs. The detectors were assumed to present uniform sensitivity. Note that scatter and randoms, as discussed in more depth in Chapter 8, when not accounted for in the reconstruction, degrade the quality of the reconstructed images; the degrading effect is reduced by accounting, in the imaging model, for scattered photons and random detections. In the experiments that follow, the effect of scatter and randoms was disregarded by simulating PET acquisitions free from scattered photons and randoms, isolating the problem of the estimation of the motion parameters. Further evaluations of the algorithms including scatter and randoms in the PET simulations and in the reconstruction process will be consid-

	GM ROC	GM SNR	WM ROC	WM SNR
SINGLE FRAME	0.71	2.4	0.76	2.2
NO MOTION	0.71	4.0	0.76	3.0
MOTION	0.68	2.9	0.60	3.5
JOINT FRAME	0.72	2.7	0.77	2.3
JOINT	0.78	4.3	0.82	3.2

Figure 5.26: Left: Expectation of the hidden discrete labels at convergence. Right: coefficient-of-recovery (COR) and signal-to-noise ratio (SNR) in the gray matter (GM) and white matter (WM) regions for the 5 methods employed to reconstruct the activity. The reconstructed images are reported in Fig. 5.27

.

ered in the future. Randoms and scatter are expected to have similar effects on the estimate of the activity as in the case of non motion-aware reconstruction (i.e. higher background uptake in the reconstructed image if scatter is not accounted for in the projector and back-projector utilised for the reconstruction) and to have little or no effect on the estimates of the motion parameters, due to their low number of degrees of freedom. The MRI and activity images were defined on a cubic grid of $(128 \times 128 \times 128)$ voxels. The number of tissue types was assumed to be $N_k = 4$; $\mu_{y,k}$ were initialised to evenly spaced values in the range of intensity of the MRI image; σ_y were initialised to $1/N_k$ of the image intensity range; $\mu_{\lambda,k}$ were initialised to 1 and $\sigma_{\lambda,k}$ to $1/N_k$ of the maximum activity assigned to the phantom. The transformations, the parameters of the Gaussian mixture and the activity were updated repeating 100 iterations of (5.128)-(5.137). Results of the reconstruction are reported in Fig. 5.27. Fig. 5.27 reports the coefficient-of-recovery (ROC) and the signal-to-noise ratio (SNR), defined in Chapter 4 - equations (4.72) and (4.73). The activity estimate produced by applying the joint generative model presents higher COR both in grey matter and white matter when compared to standard MLEM reconstruction of the motion-corrupted data (Fig. 5.26-right). Even when compared to motion-free MLEM reconstruction, the activity obtained with the generative model presents slightly higher COR and SNR, due to the MRI images improving the activity estimate.

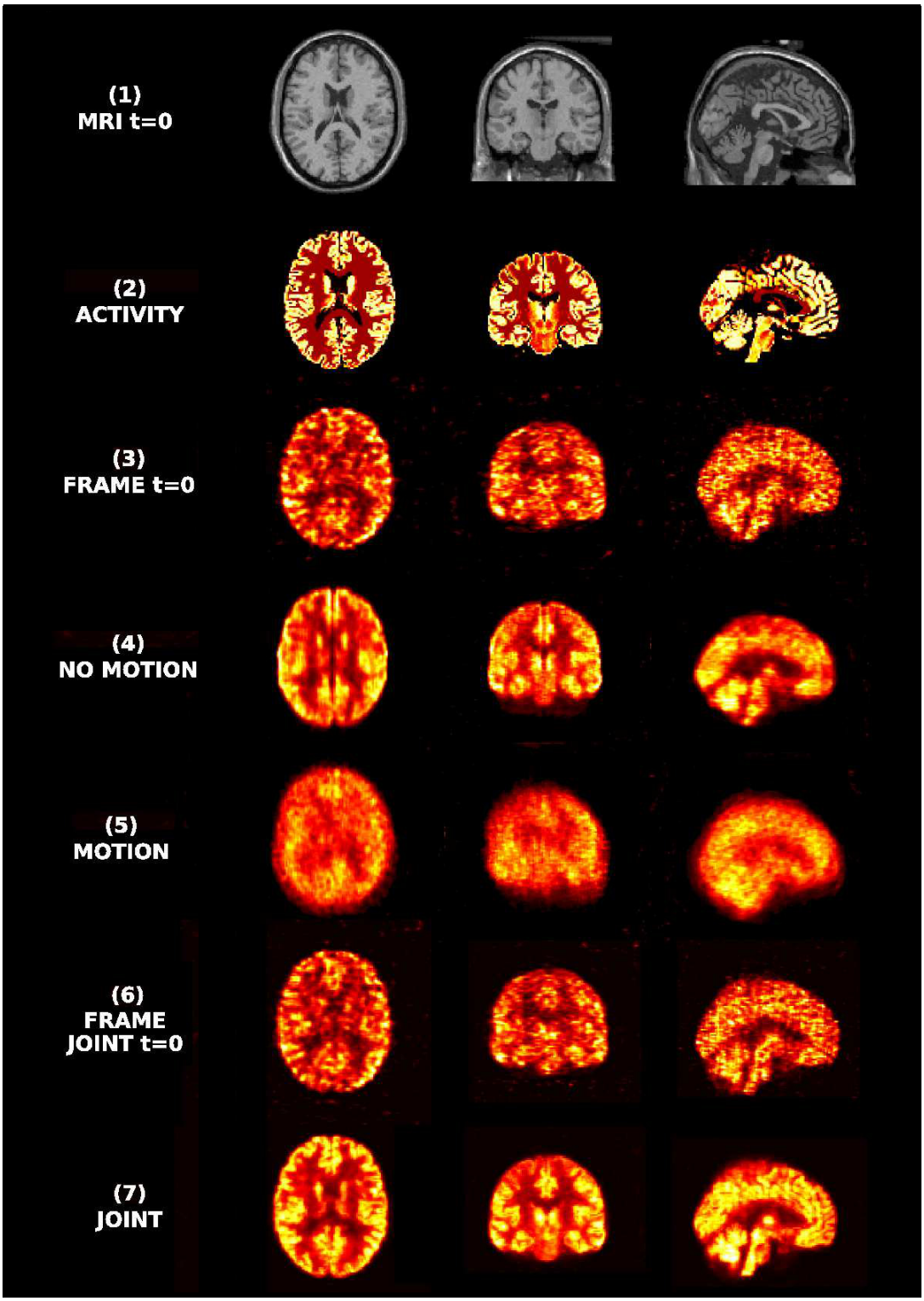


Figure 5.27: From top to bottom: (1) MRI image frame at time t = 0; (2) activity phantom; (3) MLEM reconstruction of single frame (t = 0); (4) MLEM reconstruction of motion-free simulation; (5) MLEM reconstruction with no motion compensation; (6) reconstruction of single frame using joint generative model; (7) joint estimation of motion and activity. The method estimates correctly the rigid motion parameters, and reduces noise.

5.7 IV A *plausible* model for longitudinal registration

Finally, disregarding the emission data, the model of Fig. 5.1 entails a model for the registration of sequences of images, represented by the DAG in Fig. 5.28. Alternating the optimisation of the motion parameters and of the parameters of the mixture model, we get:

$$\frac{\partial}{\partial \gamma_r^{[t]}} \log p(\gamma|\boldsymbol{y},\theta,\pi) = \sum_{v=1}^{N_v} \left[\frac{\partial}{\partial \gamma_r^{[t]}} \boldsymbol{y}^{[t]} \left(T^{\gamma^{[t]}} X_v \right) \sum_{k=1}^{N_k} \iota_{vk} \frac{\mu_{y,k} - \boldsymbol{y}^{[t]} \left(T^{\gamma^{[t]}} X_v \right)}{\sigma_{y,k}^2} \right] \tag{5.138}$$

$$\mu_{y,k}^{(n+1)} = \frac{1}{N_t \sum_{v=1}^{N_v} l_{v,k}^{(n+1)}} \sum_{t=1}^{N_t} \sum_{v=1}^{N_v} l_{v,k}^{(n+1)} \boldsymbol{y}^{[t]} \left(T^{\gamma^{[t]}} X_v \right) \tag{5.139}$$

$$\sigma_{y,k}^{(n+1)} = \left[\frac{1}{N_t \sum_{v=1}^{N_v} l_{v,k}^{(n+1)}} \sum_{t=1}^{N_t} \sum_{v=1}^{N_v} l_{v,k}^{(n+1)} \left(\mu_{y,k}^{(n+1)} - \boldsymbol{y}^{[t]} \left(T^{\gamma^{[t]}} X_v \right) \right)^2 \right]^{\frac{1}{2}} \tag{5.140}$$

Such model extends longitudinally the model described in the Unified Segmentation framework [5]. The main difference is that in the Unified Segmentation framework the location of the population with respect to the acquired image was considered an unknown parameter. Here the parametrisation is inverted, considering the location $\gamma^{[t]}$ of each frame with respect to a population of aligned images unknown. The inverted parametrisation trivially makes it possible to consider multiple time frames. The model addresses the problem of *plausibility* of the alignment model. The images are not just considered to be similar. They are considered to be generated by the same underlying object (a brain, the same brain). The anatomical states (i.e. the brain) determine the observation at time $[t]$. The deformation parameters also determine the observation at time $[t]$. The hidden states, defined in a reference space (the reference space is arbitrary if π is uniform and defined by a population of images otherwise), deformed by the transformation at time $[t]$, determine the observation at time $[t]$. Such model is plausible especially if

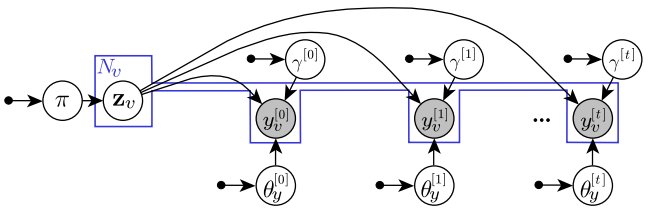


Figure 5.28: DAG of the model for longitudinal joint registration based on the finite mixture model. The MR image $\boldsymbol{y}^{[t]}$ at time $[t]$ is cased by the deformation parameters $\gamma^{[t]}$, by the parameters of the imaging system at time $[t]$ and by the hidden anatomical states in the reference space $\mathbf{Z}$. π is a statistical atlas.

one is ultimately interested in the hidden labels. Plausibility of the registration, in fact, depends on the use that one intends to make of the alignment. If one is interested in the hidden labels, these should be expressed in the transformation model.

Fig. 5.29 and Fig. 5.30 display the registration of two pairs of spherical phantoms. The incommensurate spheres (Fig. 5.30) cannot be aligned with the sum of squared differences cost function. Fig. 5.31 displays the registration of the 10 images used in the experiments of paragraph 5.6. The model can be extended to perform non-rigid registration.

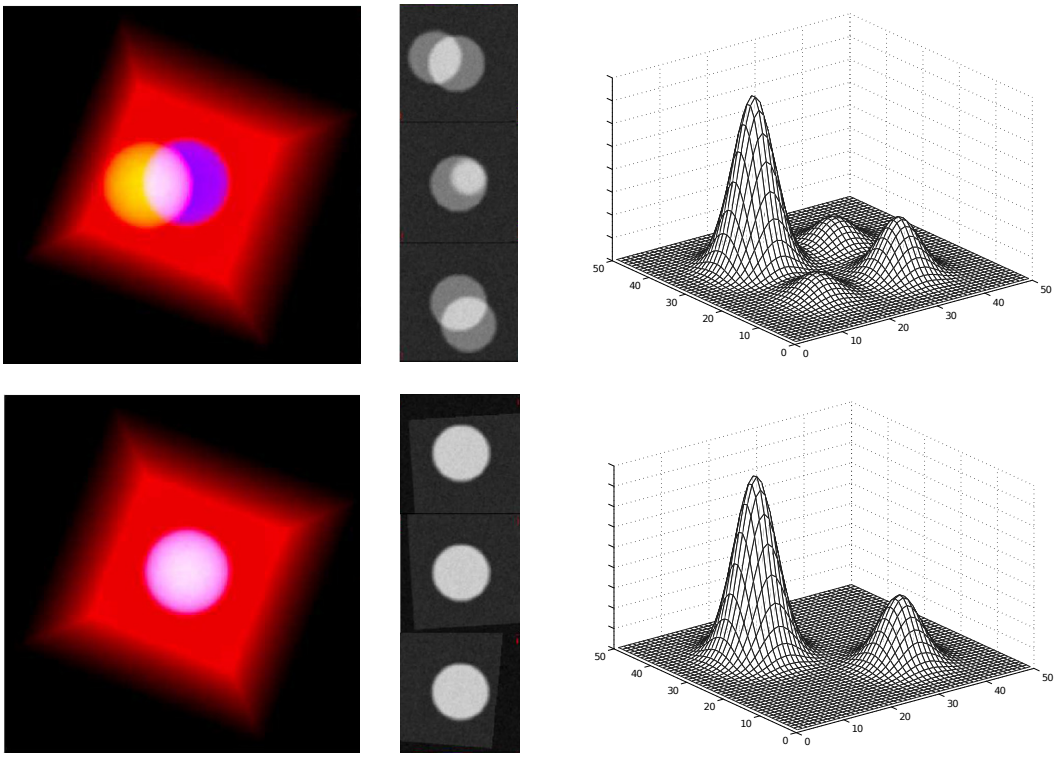


Figure 5.29: Rigid registration of simple phantoms with the mixture model of Fig. 5.28. Two spheres in uniform background with commensurate intensities (both spheres have higher intensity than the background). Top: misaligned images. Bottom: images registered with the finite mixture model. The plots report the smoothed joint histogram of the intensity of the two images before and after the registration. Note that this registration can easily be performed with the sum of squared differences (SSD) cost function.

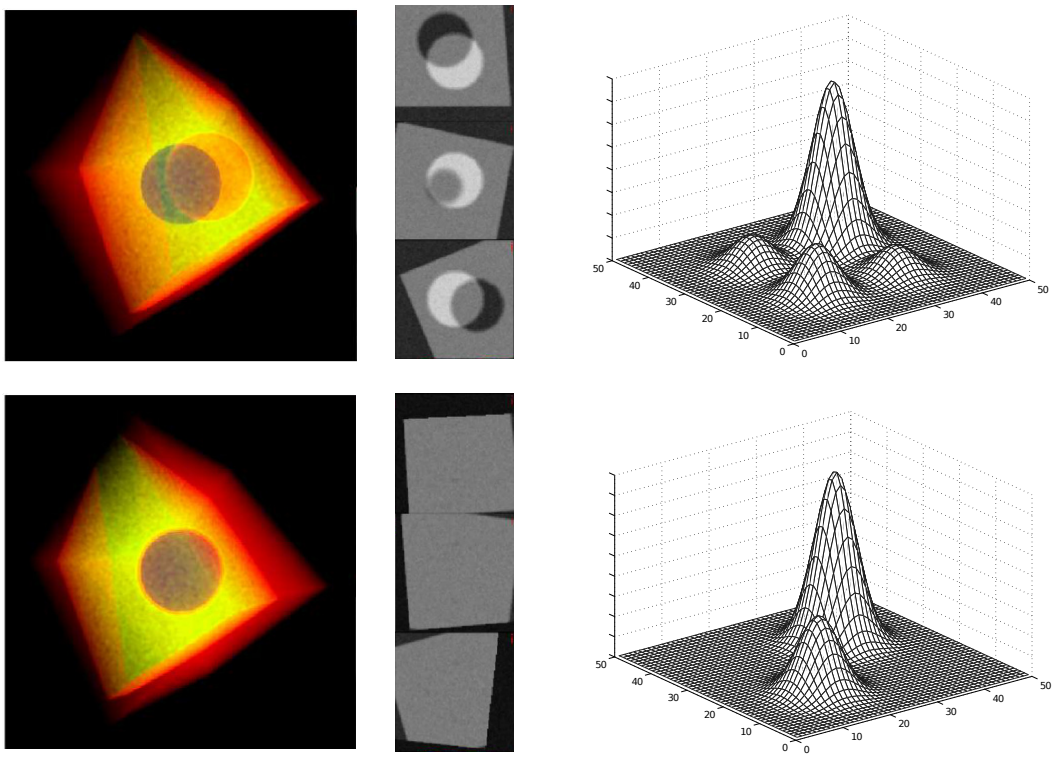


Figure 5.30: Rigid registration of simple phantoms with the mixture model of Fig. 5.28. Two spheres in uniform background with incommensurate intensities (high intensity sphere in low intensity background and low intensity sphere in high intensity background). Top: misaligned images. Bottom: images registered with the finite mixture model. The plots report the smoothed joint histogram of the intensity of the two images before and after the registration. Note that this registration cannot be obtained with the SSD cost function.

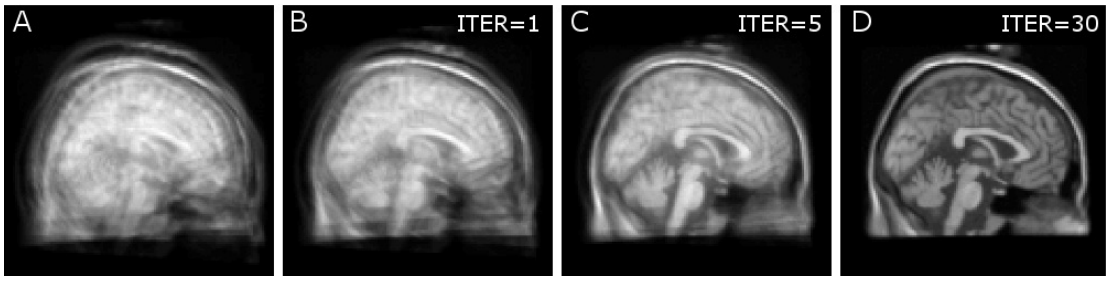


Figure 5.31: MR image randomly translated and rotated 10 times. Image A displays the overlay of the 10 randomly translated and rotated images. The rotations and translations are then recovered with the algorithm (5.138)-(5.140). Images B,C,D display the 10 overlayed images at iteration 1,5 and 30 respectively. The MR images are the same used for the experiments in paragraph 5.6.

Chapter 6

Semi-parametric learning

The model described in Chapter 4 assumes the existence of finite states of tissue and parametric distributions for the MR image intensity and for the concentration of pharmaceutical within each state. The first assumption that the MR image intensity is normally distributed for a given type of tissue is supported by the Gaussian approximation of Rician noise for high SNR and by the histogram of the intensity of T1-weighted and T2-weighted MR images, presenting bell-shaped peaks (see Fig. 6.4). The second assumption, that for each type of tissue the activity is distributed according again to a parametric distribution (a Gaussian function), is arbitrary. While the first assumption that there exist finite states, and that within each state the intensity of the MR image is normally distributed, is to some extent veryfiable, even from a single acquisition, the second assumption is difficult to verify.

As clarified in more depth in the following, one may describe the model of Chapter 4 in the context of *density estimation*, regarding the choice of the bivariate Gaussian mixture as the choice of a specific *hypothesis space* for the density that characterises the bivariate process that generates the MR image intensity and the spatial concentration of pharmaceutical. The expression *hypothesis space* is used here to denote the family of functions that is employed to represent the joint probability distribution of the variables, i.e. the space of the hypotheses of the joint probability distribution. In Chapter 4, the activity in a voxel and the MRI image intensity in the same voxel were considered realisations of a Gaussian mixture of unknown parameters. The choice of hypothesis space described in Chapter 4 may be regarded as *parametric* as we were searching for the parameters of a parametric function, the bivariate Gaussian mixture, that justify at best the observations (under the MAP criterion). We may have adopted a different choice of parametric functions, or, as discussed in this chapter, avoided the parametric approach altogether.

The main question addressed in this chapter is how to introduce hypothesis spaces with functional forms other than the particular choice of Chapter 4 (i.e. parametric, Gaussian). Fig. 6.1 and Fig. 6.2 report the graphical models and examples of joint probability distributions for the parametric, semi-parametric and non-parametric approaches. Formulating the generative model in the Information Theoretic framework, under the Principle of Maximum Uncertainty, enables the formulation of the assumption

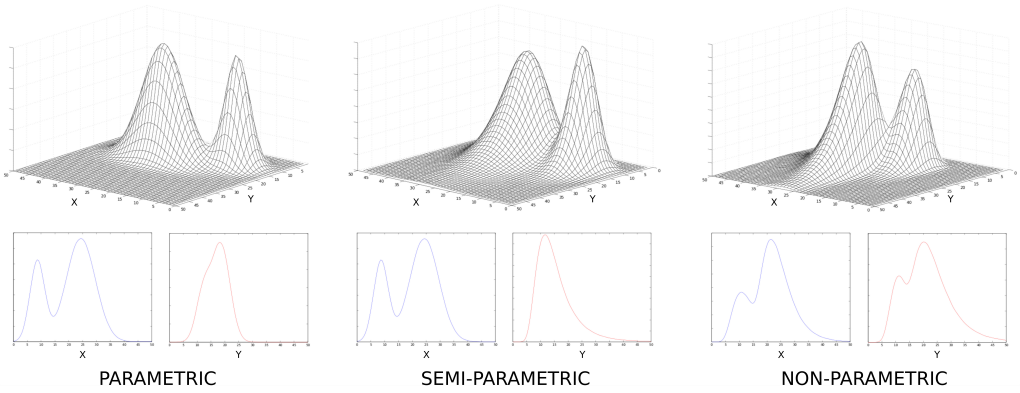


Figure 6.1: Examples of parametric, semi-parametric and non-parametric probability density functions. Left: probability density function of a parametric model (bivariate Gaussian mixture with two mixture components). Note that the marginal probability densities of X (blue) and Y (red) are univariate Gaussian mixtures. Centre: probability density function of a semi-parametric mixture model with two mixture components. Note that the marginal probability density of variable X (blue) is a univariate Gaussian mixture, while the marginal probability density of variable Y (red) is not a mixture of Gaussians. Right: non-parametric model: the joint density is the sum of smooth kernel functions. Note that the marginals of this joint density are not the sum of two Gaussians as they present asymmetric tails.

that the bivariate density function is the finite sum of kernel functions. Such formulation corresponds to the method of Joint Entropy described in [63] and in [87].

Formulating the parametric and non-parametric models in the same framework suggests a semi-parametric formulation, described in this chapter. Before diving into the semi-parametric formulation, the next paragraph contextualises the parametric vs. non-parametric approaches in the field of medical image registration (closely related to tomographic reconstruction), pointing out that the common use of the expression *Joint Entropy* to refer to a certain class of image registration methods can be misleading.

6.1 Two approaches to medical image registration

Joint Entropy (JE) and Mutual Information (MI) were introduced in the context of image registration in 1995 [88] and subsequently applied to the registration of medical images [89][90]. Recent publications have described the application of JE and MI as similarity measures for multi-modal tomographic reconstruction [63, 87].

In the same years (1999), finite mixture models were applied to the classification of brain tissue [70]. In 2005 the Unified Segmentation framework [5] extended the application of Gaussian mixtures to medical imaging applications, applying them to the problem of alingning non-rigidly an image of the brain to a population of similar images.

The model described in [5], which inspired the models described in chapter 4 and chapter 5 of this

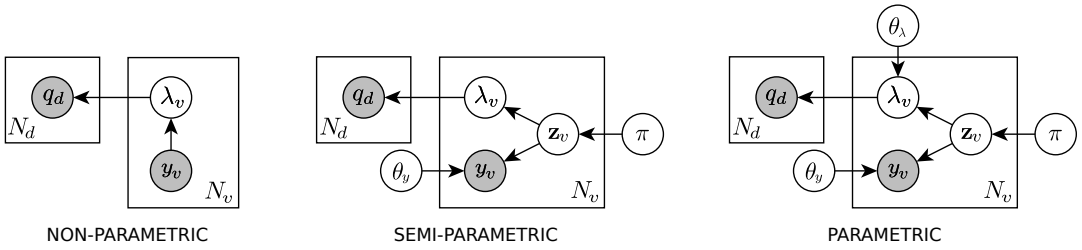


Figure 6.2: Probabilistic Graphical Models (PGM) representing the assumptions of conditional independence underlying the non-parametric method (left) - often referred to as Joint Entropy, the parametric method (right) described in Chapter 4, and the semi-parametric method (centre) described in this Chapter. Known (observed) variables are shaded. In the non-parametric model, the activity and the MRI image intensity are considered to be samples of a smooth non-parametric function. In the parametric model, the activity and the MRI image intensity are considered samples of a parametric function (a bivariate Gaussian mixture) - each voxels has a different mix of probabilities to be drawn from each of the mixture components, expressed by π; the mixture model corresponds to the assumption that there exist hidden discrete variables and that the probability distribution associated to the MRI image intensity y_v and to the activity λ_v are Gaussian (i.e. a parametric function) given the value of the hidden state. In the semi-parametric model, the probability distribution associated to y_v given the hidden state is Gaussian, however the probability distribution associated to the activity, given the hidden state is a smooth non-parametric function.

thesis, assumes that there exist a finite number of hidden types of tissue and that the intensity of a voxel of the MR image is the realisation of a Gaussian random process if the hidden state is known. The prior probability of the hidden states in each voxel is obtained from a population of images inter-aligned in a reference space. The algorithm finds the parameters of the Gaussians and the alignment of the image with the reference space such that the probability to observe the image is maximum.

Gaussian Mixtures have gained popularity in applications oriented towards the classification of regions, especially for the analysis of brain images, while the JE and MI similarity metrics have gained popularity in multi-modal registration and atlas-free applications.

JE and MI, in the context of image registration and reconstruction are often regarded as measures of *similarity* between two or eventually multiple images. Joint Entropy and Mutual Information, however, are metonymies: the methods are named after the inference criterion, rather than by the model, causing the same confusion that would be raised by naming *MAP* the registration algorithm based on the Gaussian mixture model. The difference between the method denominated, in the context of image registration (and also in the context of tomographic reconstruction) as Joint Entropy and the method based on the mixture of Gaussians stands the choice of the hypothesis space of the joint density, which, in the first case is *non-parametric* and *parametric* in the second case. These concepts will be more clear after defining

the parametric and non-parametric approaches in the next paragraph.

6.2 The semi-parametric model

6.2.1 Model of the emission imaging system

The model of the emission imaging system, described in detail in chapter 2.3, is expressed by the following conditional probability distribution of the photon counts $\boldsymbol{q}$ given the rate of emission of gamma radiation $\boldsymbol{\lambda}$, proportional to the spatial concentration of radio-pharmaceutical:

$$p(\boldsymbol{q}|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} \mathcal{P}\left(q_d; \sum_{v=1}^{N_v} a_{vd}\lambda_v\right) \tag{6.1}$$

As described in chapter 2.3, the system matrix a_{vd} encompasses the characteristics of the imaging system (location, size, sensitivity, spatial resolution of the Gamma detectors, eventual collimators) and the attenuation of Gamma radiation through the patient. We are interested in the MAP estimate:

$$\hat{\boldsymbol{\lambda}} = \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \log p(\boldsymbol{\lambda}|\boldsymbol{q}) + \log p(\boldsymbol{\lambda}) \tag{6.2}$$

Where the probability distribution $p(\boldsymbol{\lambda})$ represents the belief of the observer about $\boldsymbol{\lambda}$ prior to the observation of the photon counts.

6.2.2 Prior probability distribution

The Principle of Maximum Entropy, formulated by E.T. Jaynes in 1963 [9], expresses a general criterion for the choice of the prior probability distribution. The Principle of Maximum Entropy is based on the observation that probability is a measure of the state of knowledge of the observer and therefore subjective. It states that a prior probability distribution should express maximum uncertainty. As derived in Appendix 6.A, the Gibbs measure is the unique measure that maximizes the uncertainty for a given expected entropy (i.e. amount of information expressed by the prior):

$$p(\boldsymbol{\lambda}) = \frac{e^{-\beta E_{\boldsymbol{\lambda}}}}{\int e^{-\beta E_{\boldsymbol{\lambda}}} \mathrm{d}\boldsymbol{\lambda}} = \frac{1}{Z} e^{-\beta E_{\boldsymbol{\lambda}}} \tag{6.3}$$

where the Entropy $E_{\boldsymbol{\lambda}}$ associated with state $\boldsymbol{\lambda}$ is a measure of the information contained in the state $\boldsymbol{\lambda}$ (Appendix 6.A).

Assuming that the observer has no knowledge of structural properties of the random field $\boldsymbol{\lambda}$, the amount of information necessary to reveal the state of $\boldsymbol{\lambda}$ depends solely on the recurrence of symbols in vector $\boldsymbol{\lambda}$, where the image intensity at each voxel represents a symbol in the voxel-based approach; in other words, if the observer has no knowledge of the structure, the disclosure of $\boldsymbol{\lambda}$ is equivalent to the disclosure of independent symbols λ_v (see Shannon [91]). If symbols were discrete, the amount of information $E_{\boldsymbol{\lambda}}$ necessary to communicate a state $\boldsymbol{\lambda}$ could be calculated exactly by the histogram of the recurrence of the symbols, whereas in the continuous domain of the pharmaceutical density, the information necessary to reveal the state of $\boldsymbol{\lambda}$ is expressed by the Differential Entropy [91]:

$$E_{\boldsymbol{\lambda}} = -N_v \int p_{\boldsymbol{\lambda}}(\lambda) \ln p_{\boldsymbol{\lambda}}(\lambda) \mathrm{d}\lambda \tag{6.4}$$

where $p_{\boldsymbol{\lambda}}(\lambda)$ is the probability distribution of the symbols of vector $\boldsymbol{\lambda}$. $E_{\boldsymbol{\lambda}}$ can be calculated for a given pharmaceutical uptake $\boldsymbol{\lambda}$, and given an algorithm for the estimation of the probability distribution of the symbols $p_{\boldsymbol{\lambda}}(\lambda)$. Equation (6.3) then expresses the prior probability to observe that specific pharmaceutical uptake. The prior probability in (6.3) cannot be computed for every value of $\boldsymbol{\lambda}$, nor on a regular grid in the space of $\boldsymbol{\lambda}$ because of the size of the space of $\boldsymbol{\lambda}$, but we can walk along its gradient if (6.4) is differentiable.

Considering now an intra-subject image $\boldsymbol{y} = (y_1, \ldots, y_v, \ldots, y_{N_v})$, such as a magnetic resonance (MR) image, the amount of information necessary to communicate $\boldsymbol{\lambda}$ when $\boldsymbol{y}$ is observed equals the information that is left in $\boldsymbol{\lambda}$ when $\boldsymbol{y}$ is disclosed, expressed by the Conditional Entropy:

$$E_{\boldsymbol{\lambda}|\boldsymbol{y}} = E_{\boldsymbol{y},\boldsymbol{\lambda}} - E_{\boldsymbol{y}} \tag{6.5}$$

(which equals $E_{\boldsymbol{\lambda}}$ if $\boldsymbol{\lambda}$ and $\boldsymbol{y}$ are independent, in which case $\boldsymbol{y}$ is not informative of $\boldsymbol{\lambda}$). The prior probability distribution of $\boldsymbol{\lambda}$ when $\boldsymbol{y}$ is observed, applying again the Principle of Maximum Entropy, has the following expression:

$$p(\boldsymbol{\lambda}) = \frac{1}{Z} e^{-\beta E_{\boldsymbol{\lambda}|\boldsymbol{y}}} \tag{6.6}$$

In order to apply a gradient-type optimisation algorithm for the maximisation of (6.2), we are interested in the gradient of the $\log$ prior with respect to λ_v, which can be expressed from (6.5) and (6.6) equivalently in terms of the Conditional Entropy or of the Joint Entropy: $\frac{\partial \log p(\boldsymbol{\lambda})}{\partial \lambda_v} = -\beta \frac{\partial}{\partial \lambda_v} E_{\boldsymbol{\lambda}|\boldsymbol{y}} = -\beta \frac{\partial}{\partial \lambda_v} E_{\boldsymbol{y},\boldsymbol{\lambda}}$. We will use the second expression.

The first assumption that underlies our semi-parametric model (and the non-parametric model in [92, 19, 93]) is that knowledge of the MR image intensity in one voxel v gives information about the activity in the same voxel, however when the MR intensity in the voxel v is known, knowledge of the neighbouring MR voxels does not add information about activity in v. This assumption corresponds to the factorisation of the joint probability distribution $p_{\boldsymbol{\lambda}}(\lambda, y)$ represented by the Directed Acyclical Graph (DAG) in figure 6.3-left. Under this assumption, the amount of information necessary to communicate $\boldsymbol{y}$ and $\boldsymbol{\lambda}$ depends on the recurrence of symbols (λ, y):

$$E_{\boldsymbol{y},\boldsymbol{\lambda}} = -\int p_{\boldsymbol{\lambda}}(\lambda, y) \log p_{\boldsymbol{\lambda}}(\lambda, y) \, \mathrm{d}y \mathrm{d}\lambda \tag{6.7}$$

If the joint probability distribution of the symbols $p_{\boldsymbol{\lambda}}(\lambda, y)$ is estimated with a non-parametric method, one obtains the classical non-parametric Joint Entropy based regularisation reported in Appendix 6.B and in [92, 65, 63].

6.2.3 Semi-parametric model

The histogram of T1-weighted and T2-weighted MR images (figure 6.4 reports the histogram for one of the T1-weighted images from the DaTSCAN study in section 6.4) suggests that the image intensity across different voxels is generated by a finite number of noisy processes. The probability distribution of image intensity can be described as the sum of simple parametric functions, obtaining a finite mixture

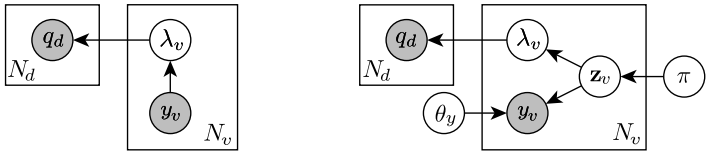


Figure 6.3: Probabilistic Graphical Models (PGM) representing the assumptions of conditional independence underlying the non-parametric JE prior (left) and the proposed semi-parametric Joint Entropy prior SP-JE (right). Known (observed) variables are shaded. In the graph on the right the activity in a voxel λ_v and the MR image intensity y_v are independent conditionally to the hidden state $\mathbf{z}_v$. θ are the parameters that describe the MR imaging process and π is a multinomial latent prior probability associated to the hidden state. Activity determines photon counts q_d in detector bins d.

model. Estimation of the model parameters of the finite mixture can be treated in terms of Maximum Likelihood Estimation (MLE), however the MLE problem for the mixture does not have a closed form solution, which makes the calculation of the model parameters notoriously difficult. This computational issue has been largely resolved with the development of the Expectation Maximisation (EM) algorithm [12].

Multi-variate mixture models have also been adopted to regularise the reconstruction of Emission Tomographic images [69, 94]. Such parametric multi-modal imaging models assume that there exist clusters in the joint probability distribution of pharmaceutical density and MR image intensity, each cluster corresponding, under the hidden variable interpretation, to one of the allowed (now anatomical-functional) finite states. Fully parametric mixture models, in other words, may be considered joint models of the pharmaceutical uptake in tissue and of the MR acquisition, which assume that for a given type of tissue the pharmaceutical uptake is uncertain and distributed according to some parametric function. Tissue type is not certain either, but is related to the MR acquisition through a parametric probabilistic model. The parameters of such models are estimated along with the activity, providing activity estimates that are consistent with the hypothesised model. The choice of a particular functional form for the mixture components should reflect the characteristics of the data; for the purpose of T1-weighted and T2-weighted MR tissue classification, Gaussian or Gamma functions are commonly adopted due to the bell shaped peaks in the image histograms. Bivariate Gaussian and Gamma mixtures have been employed in [69, 94].

Here we adopt the hidden variable interpretation of mixture models in order to develop a semi-parametric model with more loose assumptions about the pharmaceutical uptake in tissue.

Fully parametric mixture models can be described in the framework of Maximum Entropy, obtaining optimisation algorithms by maximising the MAP (6.2) with respect to the activity and the parameters of the parametric distribution $p_{\boldsymbol{\lambda}}(\lambda, y)$ in (6.7). However, while there are simpler routes to develop optimisation algorithms for the fully parametric models [69, 94], Maximum Entropy seems the natural

choice for non-parametric (Appendix 6.B) and semi-parametric models.

Let $\mathbf{z}_v$ be the hidden discrete state in voxel v and $f_k(\cdot)$ be the mixture component that describes the joint probability of (λ, y) if the hidden state is $z_{vk} == 1$; the joint probability of (λ, y) is a finite mixture of the form:

$$p_{\boldsymbol{\lambda}}(\lambda, y) = \sum_{k=1}^{N_k} \pi_k f_k(\lambda, y) \tag{6.8}$$

where π_k is the mixing coefficient or multinomial prior probability of the latent label. Note that, for the sake of simplicity, π_k is considered here equal for each voxel (i.e. not a function of v). In order to estimate the joint probability of (λ, y), if all the mixture components f_k are assumed to come from a particular parametric family of densities such as the Gaussian, then standard mixture model techniques such as EM may be employed [12]. However we wish to avoid the parametric assumption for the mixture component that models the activity distribution for a given anatomical state $p_{\boldsymbol{\lambda}}(y|\mathbf{z}_v)$. While for the MR image the histogram (see figure 6.4) justifies the assumption that there exist finite anatomical states and that the MR image intensity for a given state has a Gaussian probability distribution, we are not so confident about the probability distribution that describes the pharmaceutical density within each anatomical state.

A common restriction placed on f_k, which we adopt here as the third assumption, is that the mixture component f_k is equal to the product of its marginal densities. This means, under the hidden variable interpretation of the mixture model, that if the hidden state $\mathbf{z}_v$ in voxel v were known, the probability distribution of y_v would not be affected by knowledge of λ_v. This condition, expressed by the Directed Acyclical Graph (DAG) of the model in figure 6.3-right, corresponds to the assumption that the hidden state is the unique cause of interdependence of the imaging modalities [8]:

$$p_{\boldsymbol{\lambda}}(\lambda, y) = \sum_{k=1}^{N_k} \pi_k p_{\boldsymbol{\lambda}}(\lambda|\mathbf{z}_v) p_{\boldsymbol{\lambda}}(y|\mathbf{z}_v) \tag{6.9}$$

Factorisation of f_k is justified by the following arguments: 1) in fully parametric models, the use of a factorised multi-modal Gaussian Mixture has been shown to yield results virtually identical to the non-factorised multivariate Gaussian Mixture when the model is applied to the segmentation of regions of tissue with multi-spectral MR images [73]; 2) though some special cases have been treated (see [95]), a general algorithm that deals with multivariate non-parametric mixture components has not been developed, to the best of our knowledge; 3) restrictions on f_k generally improve the identifiability of the parameters, as discussed in [95].

An algorithm for density estimation with mixture models that factor according to (6.9) has been presented by Benaglia *et al.* [95] for the semi-parametric case where one or more of the factors are non-parametric functions. Semi-parametric density estimation is interpreted as a parametric problem with parameters $\varphi = (\theta, g)$, where here $\theta = (\mu_k, \sigma_k)$, with $k = 1, 2, \ldots, N_k$, are the parameters of the Gaussian mixtures for the MR image and g is the distribution that represents the mixture component for the functional image and is drawn from a set of smooth functions. Representing g as the sum of kernel

functions Φ of bandwidth σ_λ, the algorithm for the estimation of the parameters φ proposed in [95] consists in the following steps that are reminiscent of the EM algorithm for purely parametric mixture models. $\mathcal{N}$ denotes the Gaussian distribution. For each $t = 0 \ldots T$:

E-step

$$l_{vk}^{[t+1]} = \frac{\pi_k^{[t]} \tilde{p}_{\boldsymbol{\lambda}}^{[t]}(\lambda_v|\mathbf{z}_v)\mathcal{N}\left(y_v; \mu_k^{[t]}, \sigma_k^{[t]}\right)}{\sum_{k=1}^{N_k} \pi_k^{[t]} \tilde{p}_{\boldsymbol{\lambda}}^{[t]}(\lambda_v|\mathbf{z}_v)\mathcal{N}\left(y_v; \mu_k^{[t]}, \sigma_k^{[t]}\right)} \tag{6.10}$$

M-step parametric

$$\pi_k^{[t+1]} = \frac{1}{N_v} \sum_{v=1}^{N_v} l_{vk}^{[t+1]} \tag{6.11}$$

$$\mu_k^{[t+1]} = \frac{\sum_{v=1}^{N_v} l_{vk}^{[t+1]} y_v}{\sum_{v=1}^{N_v} l_{vk}^{[t+1]}} \tag{6.12}$$

$$\sigma_k^{2\,[t+1]} = \frac{\sum_{v=1}^{N_v} l_{vk}^{[t+1]} \left(y_v - \mu_k^{[t+1]}\right)^2}{\sum_{v=1}^{N_v} l_{vk}^{[t+1]}} \tag{6.13}$$

$$\tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(y|\mathbf{z}_v) = \mathcal{N}\left(y; \mu_k^{[t+1]}, \sigma_k^{[t+1]}\right) \tag{6.14}$$

M-step non-parametric

$$\tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(\lambda|\mathbf{z}_v) = \frac{1}{N_v} \sum_{v=1}^{N_v} l_{vk}^{[t+1]} \Phi\left(\frac{\lambda - \lambda_v}{\sigma_\lambda^2}\right) \tag{6.15}$$

$l_{vk}^{[t+1]}$ is the expectation of the membership of voxel pair (λ_v, y_v) to each of the N_k classes.

The estimate of the joint probability of (y_v, λ_v) at step t is the marginal over $\mathbf{z}_v$ of the estimate of the joint probability of $(y_v, \lambda_v, \mathbf{z}_v)$:

$$\tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(\lambda, y) = \frac{1}{N_v} \sum_{k=1}^{N_k} \left[\pi_k^{[t+1]} \tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(\lambda|\mathbf{z}_v)\mathcal{N}\left(y; \mu_k^{[t+1]}, \sigma_k^{[t+1]}\right)\right] \tag{6.16}$$

Which is differentiable with respect to λ_v if Φ is a differentiable kernel, such as the Gaussian kernel:

$$\frac{\partial \tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(\lambda, y)}{\partial \lambda_v} =$$
$$\frac{1}{N_v} \sum_{k=1}^{N_k} \left[l_{vk}^{[t+1]} \Phi'\left(\frac{\lambda - \lambda_v}{\sigma_\lambda^2}\right) \mathcal{N}\left(y; \mu_k^{[t+1]}, \sigma_k^{[t+1]}\right)\right] \tag{6.17}$$

Approximating the integral in (6.7) with the sum of rectangular parallelepipeds of base $(\Delta_\lambda, \Delta_y)$, the gradient of the $\log$ prior has the following expression:

$$\frac{\partial \log p(\boldsymbol{\lambda})}{\partial \lambda_v} = \frac{\partial}{\partial \lambda_v} E_{\boldsymbol{\lambda}, \boldsymbol{y}} \simeq$$
$$-\Delta_\lambda \Delta_y \sum_{i,j}^{M} \left(1 + \log \tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(y_i, \lambda_j)\right) \frac{\partial \tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(y_i, \lambda_j)}{\partial \lambda_v} \tag{6.18}$$

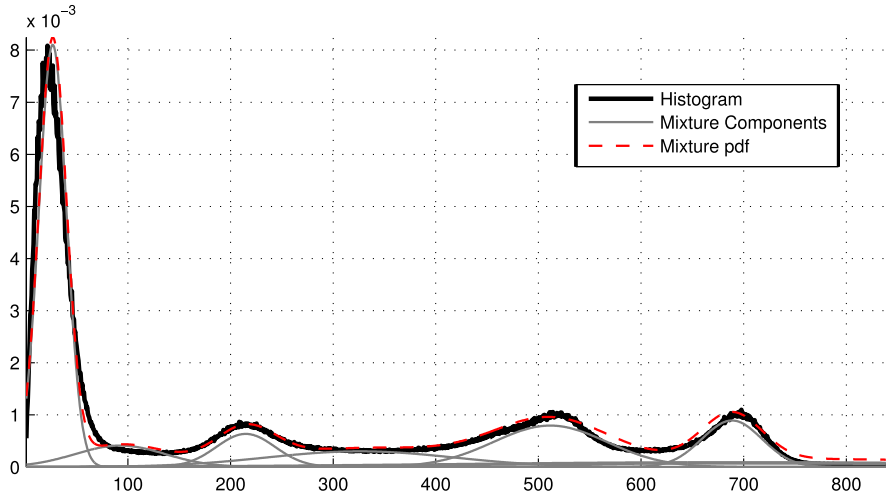


Figure 6.4: Histogram of T1-weighted image intensity and Gaussian Mixture estimate of its probability distribution function. The peaks of the histogram correspond to the finite states of the mixture model. In this example the number of classes is set to $N_k = 7$ in order to capture the 4 peaks and the 3 partial volume classes.

6.2.4 Optimisation

In order to obtain a point estimate of $\boldsymbol{\lambda}$ that maximises its $\log$ posterior probability distribution (6.2), we use the Expectation Maximisation One Step Late (EM-OSL) gradient-type optimisation algorithm proposed by Green [30]. The Expectation Maximisation algorithm for the maximisation of the Poisson likelihood was derived by Shepp and Vardi [17] by formulating the problem in terms of hidden labels (a variable indicating from which voxel a photon is emitted), obtaining an update formula by maximising the expectation of the complete data $\log$ likelihood with respect to the posterior distribution of the hidden labels [17]. The M-step does not have a closed form solution when a generic prior probability for the activity is taken into account, however the EM update can still be calculated considering the gradient of the prior at the previous iteration [30] (one step late). The EM algorithm (and EM-OSL) have the advantage, over other gradient-type algorithms, of getting rid of the step-size parameter, thanks to the optimisation scheme being tailored specifically for the optimisation of the Poisson likelihood. EM-OSL is not guaranteed to converge, but it is stable in practice as long as the gradient of the prior is small, which is true if the data evidence (photon counts) is given more importance than the prior (by controlling the parameter β).

$$\hat{\lambda}_v^{[n+1]} = \hat{\lambda}_v^{[n]} \frac{1}{\sum_{d=1}^{N_d} a_{vd} + \beta \frac{\partial}{\partial \lambda_v} \log p(\boldsymbol{\lambda})\Big|_{\boldsymbol{\lambda}^{[n]}}} \sum_{d=1}^{N_d} a_{vd} \frac{q_d}{\sum_{v'=1}^{N_v} a_{v'd}\, \hat{\lambda}_{v'}^{[n]}} \tag{6.19}$$

In summary, each update of activity involves the computation of the projection, the back-projection, and the gradient of the $\log$ prior (6.18), which involves computing the joint probability distribution of (λ, y)

(6.16) and it's gradient with respect to the activity estimate (6.17). (6.16) involves iterating through (6.10)-(6.15). Efficient estimation of the joint probability distribution (6.16) and its gradient (6.17) and their numerical integration (6.18) are described in the section 6.3-Implementation.

6.2.5 Extension to multi-spectral anatomical images

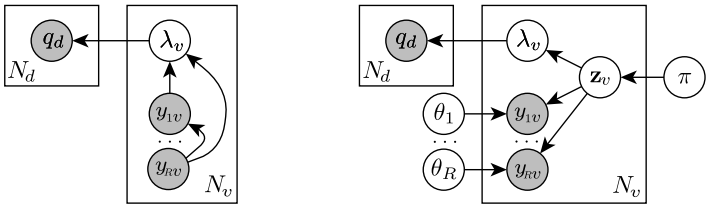


Figure 6.5: Non-parametric model (left) and the semi-parametric model (right) for multi-spectral anatomical images $(\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_R)$. The proposed model is based on the assumption that a latent discrete state is the unique cause of interdependence of the images. Under this assumption the complexity of the algorithm grows polynomially with the number of images rather than exponentially (see 6.3.1).

The semi-parametric model is based on the assumption that the joint probability distribution that describes the activity and MR image intensity in a voxel pair is a mixture distribution (6.8) and that it factors according to (6.9). Under these assumptions, the complexity of the model grows polynomially with the number of intra-subject images (see 6.3.1), rather than exponentially, due to independence across images conditionally to the hidden state (figure 6.5). The model can be extended to use information from multi-spectral MR images extending the mixture model to more dimensions. This has proven valuable for the segmentation of tissue with multi-spectral MR images such as T1-weighted, T2-weighted and proton-density images. [70]. Equations (6.9)-(6.4) have the following expressions for R anatomical images:

$$p_{\boldsymbol{\lambda}}(\lambda, y_1, y_2, \ldots, y_R) = \sum_{k=1}^{N_k} \pi_k p_{\boldsymbol{\lambda}}(\lambda|\mathbf{z}_v) \prod_{r=1}^{R} p_{\boldsymbol{\lambda}}(y_r|\mathbf{z}_r) \tag{6.20}$$

$$l_{vk}^{[t+1]} = \frac{\pi_k^{[t]} \tilde{p}_{\boldsymbol{\lambda}}^{[t]}(\lambda|\mathbf{z}_v) \prod_{r=1}^{R} \mathcal{N}\left(y_r; \mu_{rk}^{[t]}, \sigma_{rk}^{[t]}\right)}{\sum_{k=1}^{N_k} \pi_k^{[t]} \tilde{p}_{\boldsymbol{\lambda}}^{[t]}(\lambda|\mathbf{z}_v) \prod_{r=1}^{R} \mathcal{N}\left(y_r; \mu_{rk}^{[t]}, \sigma_{rk}^{[t]}\right)} \tag{6.21}$$

$$\pi_k^{[t+1]} = \frac{1}{N_v} \sum_{v=1}^{N_v} l_{vk}^{[t+1]} \tag{6.22}$$

$$\mu_{rk}^{[t+1]} = \frac{\sum_{v=1}^{N_v} l_{vk}^{[t+1]} y_{rk}}{\sum_{v=1}^{N_v} l_{vk}^{[t+1]}} \tag{6.23}$$

$$\sigma_{rk}^{2\,[t+1]} = \frac{\sum_{v=1}^{N_v} l_{vk}^{[t+1]} \left(y_{rk} - \mu_{rk}^{[t]}\right)^2}{\sum_{v=1}^{N_v} l_{vk}^{[t+1]}} \tag{6.24}$$

$$\tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(y_r|\mathbf{z}_r) = \mathcal{N}\left(y_r; \mu_{rk}^{[t+1]}, \sigma_{rk}^{[t+1]}\right) \tag{6.25}$$

$$\tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(\lambda|\mathbf{z}_v) = \frac{1}{N_v}\sum_{v=1}^{N_v} l_{vk}^{[t+1]}\Phi\left(\frac{\lambda-\lambda_v}{\sigma_\lambda^2}\right) \tag{6.26}$$

$$\tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(\lambda, y_1, y_2, \ldots, y_R) = \frac{1}{N_v}\sum_{k=1}^{N_k}\left[\pi_k^{[t+1]}\tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(\lambda|\mathbf{z}_v)\prod_{r=1}^{R}\mathcal{N}\left(y_r;\mu_{rk},\sigma_{rk}\right)\right] \tag{6.27}$$

$$\frac{\partial\tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(\lambda, y_1, y_2, \ldots, y_R)}{\partial\lambda_v} = \frac{1}{N_v}\sum_{k=1}^{N_k}\left[l_{vk}^{[t+1]}\Phi'\left(\frac{\lambda-\lambda_v}{\sigma_\lambda^2}\right)\prod_{r=1}^{R}\mathcal{N}\left(y_r;\mu_{rk}^{[t+1]},\sigma_{rk}^{[t+1]}\right)\right] \tag{6.28}$$

$$\frac{\partial}{\partial\lambda_v}E_{\boldsymbol{\lambda},\mathbf{y_1},\mathbf{y_2},\ldots,\mathbf{y_R}} \simeq -\Delta\sum_{i,j_1,\ldots,j_r}^{M^(1+R)}\left(1+\log\tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(\cdot)_{i,j_1,\ldots,j_r}\right)\frac{\partial\tilde{p}_{\boldsymbol{\lambda}}^{[t+1]}(\cdot)_{i,j_1,\ldots,j_r}}{\partial\lambda_v} \tag{6.29}$$

6.3 Numerical implementation

6.3.1 Efficient computation of the prior term

The evaluation of the gradient of the Differential Entropy term (6.18), (6.17), (6.16) is computationally expensive, requiring MN_v evaluations of the Gaussian kernel, which is unacceptable with 3D images due to the order of magnitude of N_v. We adopt the approximation of the gradient described by Shwartz et al. [96], which is significantly more efficient than the exact computation and proves sufficiently accurate. Referring to the computation of the uni-dimensional Differential Entropy in (6.40), the gradient with respect to λ_v is replaced by the gradient with respect to the nearest λ_n, with λ_n being the nodes of the regular grid for the integration. Letting $h(j) = 1+\log p(\lambda_j)$, substituting (6.41) into (6.40), the gradient with respect to y_n is

$$\frac{\partial E_{\boldsymbol{\lambda}}}{\partial\lambda_n} \simeq -\Delta_\lambda\sum_{j}^{M} h(j)\Phi'\left(\frac{\Delta_\lambda(j-n)}{\sigma_\lambda^2}\right) \tag{6.30}$$

This is a good approximation for sufficiently high number of nodes and the complexity is greatly reduced for $M << N_v$ as the evaluation of the convolution in (6.30) can be performed in $\mathcal{O}(M^2)$ or in $\mathcal{O}(M\log M)$ by Fast Fourier Transform and multiplication.

Equivalently, for the gradient of the Differential Joint Entropy in (6.47):

$$\frac{\partial E_{\mathbf{y},\boldsymbol{\lambda}}}{\partial\lambda_n} \simeq -\Delta_y\Delta_\lambda\sum_{i,j}^{M^2} h(i,j)\Phi\left(\frac{\Delta_y(i-n)}{\sigma_y^2}\right)\Phi'\left(\frac{\Delta_\lambda(j-n)}{\sigma_\lambda^2}\right) \tag{6.31}$$

which is $\mathcal{O}(2M^2\log M)$. For the semi-parametric model, expanding (6.18) one obtains an expression similar to (6.31), with computational complexity $\mathcal{O}(N_kM\log M)$ and $\mathcal{O}(N_k(1+R)M^{(1+R)}\log M)$ with R multi-spectral images.

6.3.2 Sensitivity and selection of the nuisance parameters

The prior based on Joint Entropy has a number of parameters: β, which controls the importance of the prior; the bandwidth of the Gaussian kernel for Parzen Windows estimation of the *joint pdf* σ_y and σ_λ (6.47); the size of the discretization grid for the joint entropy Δ_y and Δ_λ (6.47). The *bias/variance* curves were computed varying Δ_y and Δ_λ, with an arbitrary value of $\sigma_y = \sigma_\lambda$ and the activity estimate appeared to be largely independent upon Δx and Δy when the discretization grid for the *joint pdf* has more than about 200 points in y and λ, for any value of β. A discretization grid of size 400×400 was chosen. For what concerns the choice of σ_y and σ_λ, multiple reconstructions were performed again (with the grid 400×400) and the quality of the reconstruction in terms of *bias/variance* appeared to increase when σ_y and σ_λ decrease, and then to abruptly decrease when they are down to the order of Δ_y and Δ_λ. The following values were adopted $\sigma_y = 10 * \Delta_y$ and $\sigma_\lambda = 10 * \Delta_\lambda$. This procedure gave insight of the effect of the parameters and allowed us to consider β as the only parameter of the algorithm.

6.4 Experiments

The proposed semiparametric algorithm was applied to the reconstruction of synthetic brain perfusion data and to the reconstruction of clinical DaTSCAN SPECT.

6.4.1 Brain phantom experiments

Multi-modal synthetic phantom

Multiple instances of the sinogram were generated by applying Poisson noise to the ideal noise-free (high number of simulated photons) sinogram that includes scatter and attenuation, produced by the Monte Carlo simulator and by selecting photons in the energy window 126–154 KeV.

The reconstructions with the two regularisation methods were obtained with the NiftyRec GPU-accelerated reconstruction software [97]. Activity was estimated from each sinogram instance by executing 10000 iterations of the One Step Late MAP-EM algorithm of (6.19) with each of the three methods MLEM, JE, SP-JE. For SP-JE, the parameters of the Gaussians were initialised by repeating (6.10)–(6.14) until convergence with the initial uniform activity estimate, then one step of (6.10)–(6.15) was alternated with one update of the activity (6.19) throughtout the reconstruction ($T = 1$).

With access to multiple instances of the sinogram data, *bias/variance* characterization of the reconstruction algorithms was performed. Specifically, the activity was estimated from 10240 realizations of the sinogram.

Figure 6.8 reports the bias and variance norm at each iteration of the reconstruction algorithms for unconstrained MLEM and with the two anatomical priors JE and SP-JE. In presence of a prior, after a number of iterations, the curves tend to converge, while with unconstrained MLEM, the noise keeps increasing due to *dimensional instability* [93]. In order to compare the *bias/variance* curves of the three

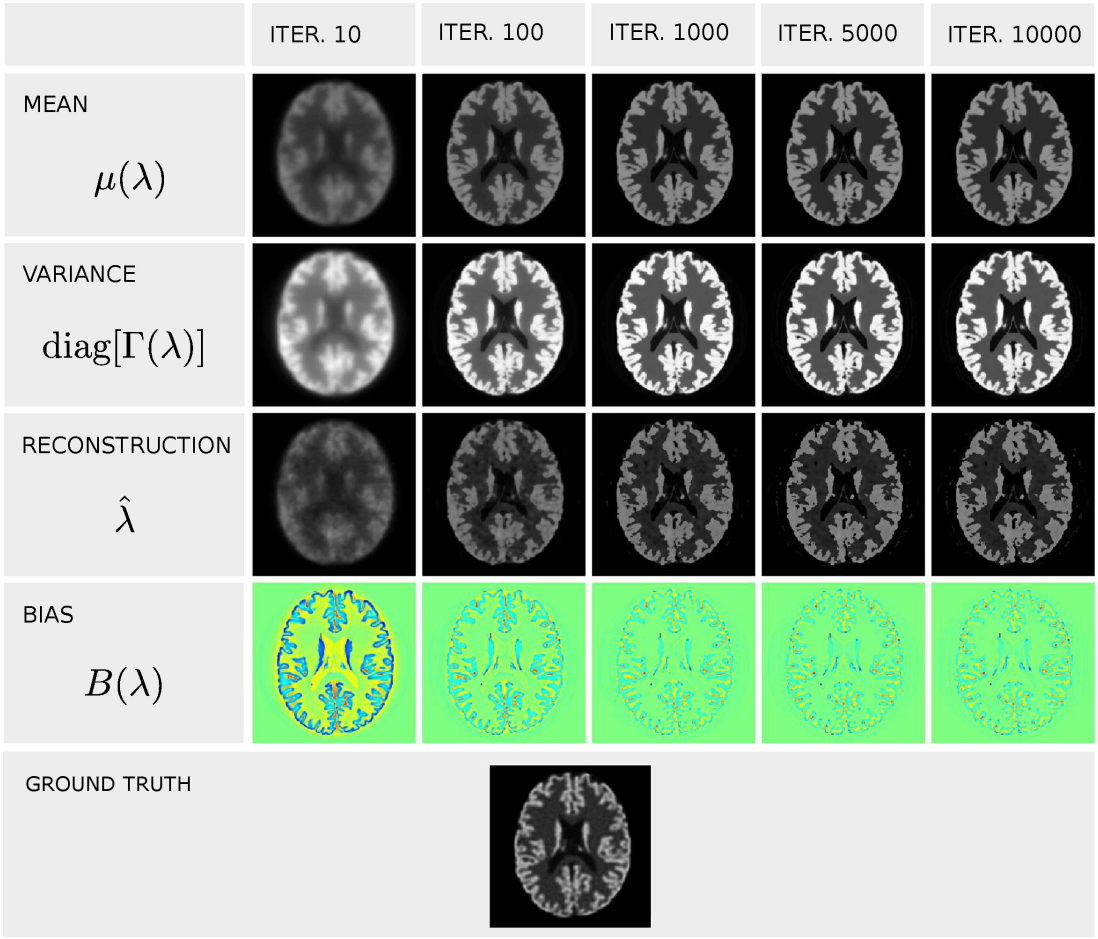


Figure 6.6: Non-parametric Joint Entropy. $N = 10240$ instances, 10 to 10000 iterations.

algorithms, the optimal values of the hyper-parameter for each reconstruction method was found by running the reconstructions of the multiple instances of noise for varying values of the hyper-parameters, obtaining the set of *bias/variance* curves in figure 6.8. The optimal value of β for each algorithm was chosen as the value that determines convergence of the *bias/variance* curve to a point closer to the axis origin (low *bias* and low *variance*).

The reconstructed images with optimum β are reported in figures 6.6 and 6.7. The reconstruction with the semi-parametric model produces images with lower *bias* when compared with the conventional non-parametric Joint Entropy method, while the noise is approximately unaffected. Consistently with the *bias/variance* curves, the images of the activity reconstructed with the two algorithms (figures 6.6 and 6.7) show approximately the same level of random variations and the image obtained with the semi-parametric model is more similar to the true activity at visual inspection. Remarkably, the images obtained with SP-JE present boundaries that are more resembling of the MR image.

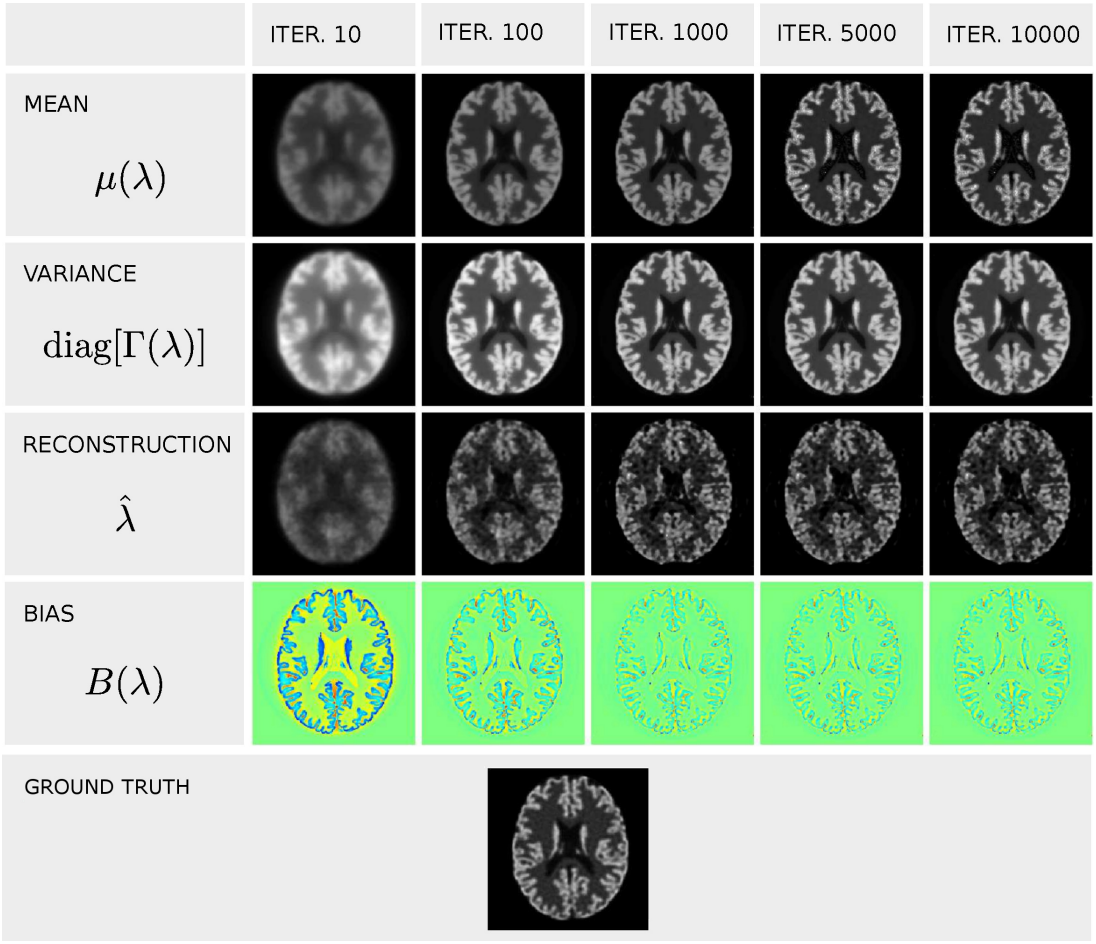


Figure 6.7: Semi-parametric Joint Entropy. $N = 10240$ instances, 10 to 10000 iterations. The third row (reconstruction) reports the reconstruction of a single instance of the measurement.

6.4.2 DaTSCAN SPECT experiments

Anatomical information from an intra-subject T1-weighted MR image was used to regularise the reconstruction of DaTSCAN emission images acquired from 13 subjects involved in a study for the diagnosis of Parkinson's Disease.

The DaTSCAN SPECT scans were acquired on a General Electric Infinia with Low Energy High Resolution (LEHR) collimator (spatial resolution of 7.30 mm Full Width at Half Maximum at 80 mm from the detector surface) with 120 camera positions regularly spaced between 0^o and 360^o. A low-dose computed tomography (CT) scan was acquired immediately before each SPECT study for attenuation correction. Subjects had the structural T1-weighted MR image acquired on a $1.5T$ MR scanner (GE Signa Excite).

The parameters of the three regularisation algorithms were chosen by optimising the reconstruction in terms of *bias/variance* with a digital phantom representing an average subject. This was obtained by segmenting the *striatum* region of the 13 subjects with Free Surfer [98] and applying uniform activity to

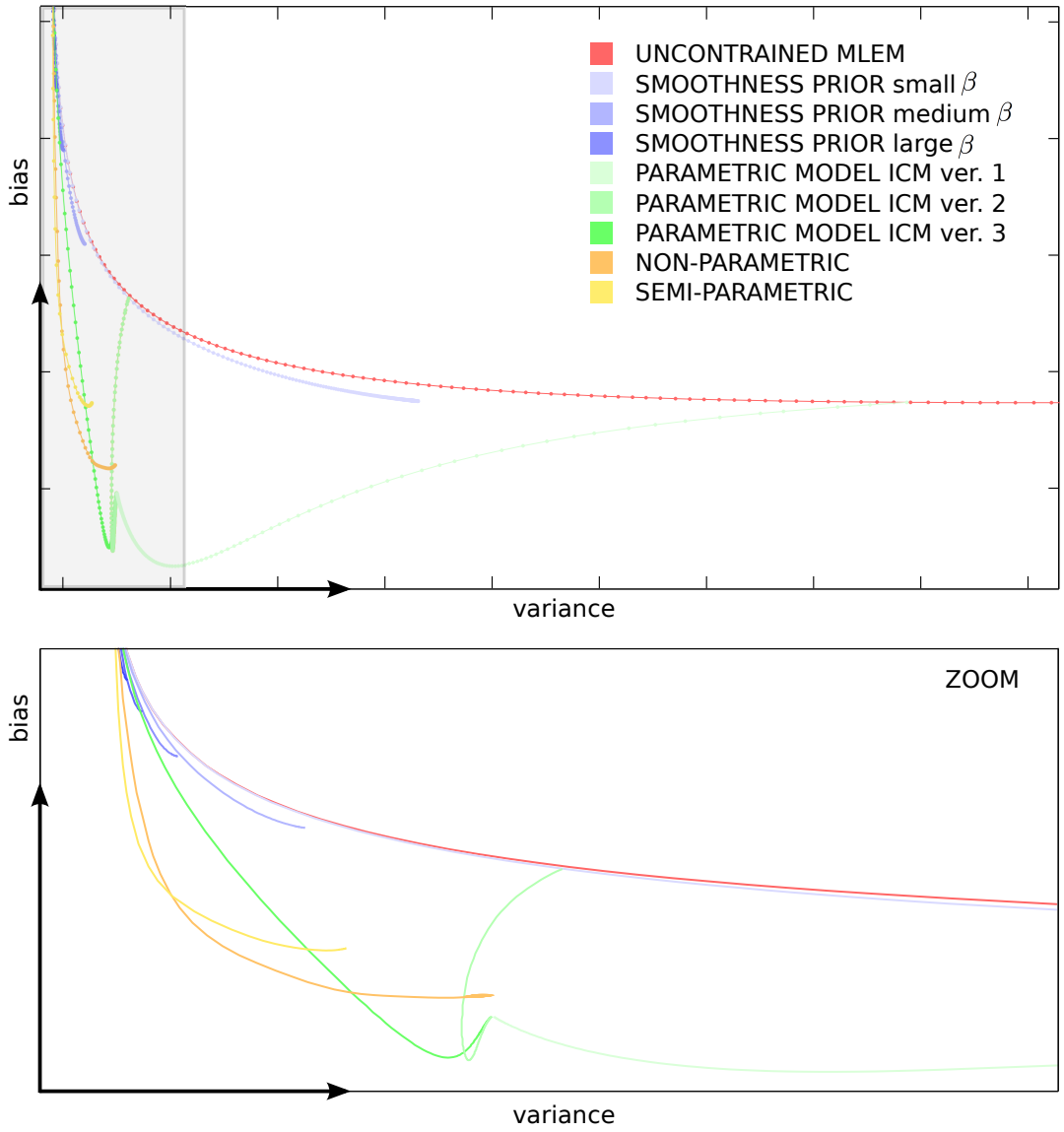


Figure 6.8: Bias variance norm (for the whole image) as a function of the number of iterations (1 to 10000 iterations). See equations (3.59) and (3.60) for the definition of bias and covariance norm (the variance is the diagonal of the covariance matrix). The curves are obtained according to (3.59) and (3.60) integrating over 10240 instances of the measurement. Red: unconstrained MLEM; blue: smoothing prior with varying smoothing parameter; green: parametric finite mixture model with different ICM schemes (see Fig. **??** in chapter 4); yellow: non-parametric Joint Entropy with optimum parameters; orange: semi-parametric Joint Entropy with optimum parameter.

the striatum and to the rest of the brain. The activity in the two regions was obtained by averaging the activity estimate given by the MLEM reconstructions of the 13 subjects. A SPECT scan of the phantom was simulated by projecting the phantom with NiftyRec with the parameters emulating the GE Infinia

LEHR scanner and by applying Poisson noise. The bias/variance curves were computed for varying values of the parameters for 30 iterations of each of the reconstruction methods. Optimum parameters were chosen by selecting the curve that ends closer to the axis origin, as described in section 6.4.1.

Finally, the spatial density of pharmaceutical was reconstructed from the 13 sinograms with unconstrained MLEM, JE and SP-JE executing 30 iterations for each algorithm with the optimum parameters. As reported in the pharmaceutical concentration image for one representative subject in Figure 6.9, the regularisation based on anatomical information produces images less noisy at visual inspection. The estimate produced by the anatomical regularisation based on the non-parametric model is more uniform throughout the brain than the semi-parametric model, which seems to allow for more differences of uptake in different tissues, while still reducing noise artefacts. The activity in the *striatum* appears sharper than with MLEM and with non-parametric regularisation.

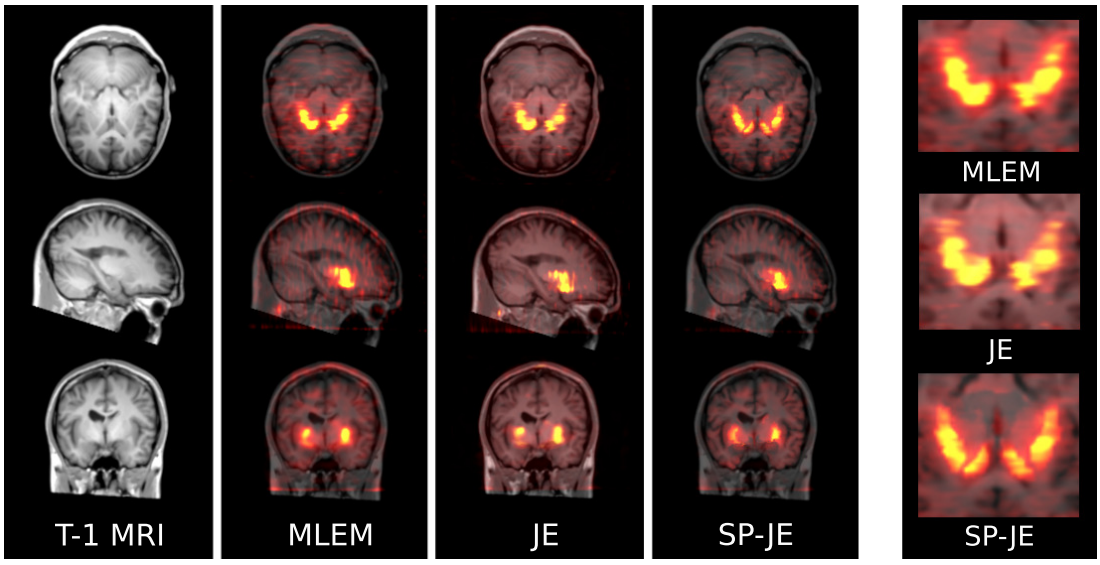


Figure 6.9: DaTSCAN SPECT of a control subject. For each reconstruction method, the images show the radio-pharmaceutical estimate after $N = 30$ iterations overlaid on the T1-weighted MR image. Nuisance parameters are chosen from optimisation of synthetic DaTSCAN data with characteristics obtained from a population of 13 subjects. The transaxial view on the right shows a close up of the striatal region.

When compared with the conventional non-parametric Joint Entropy prior, the proposed method provides lower *bias* of the activity estimate in synthetic reconstructions of brain perfusion data. At visual inspection the images are more similar to the true activity and present discontinuities more consistent with the anatomical image, thanks to the use of a more informed model of the MR acquisition system. The proposed semi-parametric model introduces a differentiable measure (6.18) which could be used for registration of multi-modal images, such as MR and SPECT or MR and PET by parametrising the semi-parametric entropy term with a deformation model.

6.A *The Principle of Maximum Entropy and the Gibbs Prior*

The Principle of Maximum Entropy, formulated by Jaynes [9], expresses a general criteria for the choice of a prior distribution, based on the principle that it should manifest maximum uncertainty in order not to assume knowledge that the observer does not posses. Uncertainty is measured by the amount if information necessary to communicate the state of the variable, expressed by Shannon's Information Entropy [91]. The amount of information necessary to communicate the state of the spatial density of the radio-pharmaceutical $\boldsymbol{\lambda}$ depends on the state (e.g. the amount of information necessary to communicate the state $\boldsymbol{\lambda}$ if $\boldsymbol{\lambda}$ is such that $y_1 = y_2 = \cdots = y_B$, is lower than if $\boldsymbol{\lambda}$ is more complex), let it be $E_{\boldsymbol{\lambda}}$. Let the probability distribution of the states be $p(\boldsymbol{\lambda})$. The average uncertainty is

$$E = \int p(\boldsymbol{\lambda})\ E_{\boldsymbol{\lambda}} \mathrm{d}\boldsymbol{\lambda} \qquad \int p(\boldsymbol{\lambda}) \mathrm{d}\boldsymbol{\lambda} = 1 \tag{6.32}$$

Applying the Principle of Maximum Entropy, the choice of $p(\boldsymbol{\lambda})$ that assumes the least amount of information is one that maximises the uncertainty about the system, expressed by the Entropy S of $\boldsymbol{\lambda}$

$$S = -\int p(\boldsymbol{\lambda}) \ln p(\boldsymbol{\lambda}) \mathrm{d}\boldsymbol{\lambda} \tag{6.33}$$

Maximisation of S subject to the constraints (6.32) can be achieved by the use of the Lagrange multiplier:

$$L = S + \alpha \left[\int p(\boldsymbol{\lambda}) \mathrm{d}\boldsymbol{\lambda} - 1\right] + \beta \left[\int p(\boldsymbol{\lambda}) E_{\boldsymbol{\lambda}} \mathrm{d}\boldsymbol{\lambda} - E\right] \tag{6.34}$$

$$\tfrac{\partial L}{\partial \alpha} = \int p(\boldsymbol{\lambda}) \mathrm{d}\boldsymbol{\lambda} - 1 = 0 \tag{6.35}$$

$$\tfrac{\partial L}{\partial \beta} = \int p(\boldsymbol{\lambda}) E_{\boldsymbol{\lambda}} \mathrm{d}\boldsymbol{\lambda} - E = 0 \tag{6.36}$$

$$\tfrac{\partial L}{\partial p(\boldsymbol{\lambda})} = \ln p(\boldsymbol{\lambda}) + \alpha + \beta E_{\boldsymbol{\lambda}} = 0 \tag{6.37}$$

Solving for α and substituting $\alpha = \ln \int e^{-\beta E_{\boldsymbol{\lambda}} \mathrm{d}\boldsymbol{\lambda}}$ in (6.37):

$$p(\boldsymbol{\lambda}) = \frac{e^{-\beta E_{\boldsymbol{\lambda}}}}{\int e^{-\beta E_{\boldsymbol{\lambda}}} \mathrm{d}\boldsymbol{\lambda}} = \frac{1}{Z} e^{-\beta E_{\boldsymbol{\lambda}}} \tag{6.38}$$

The complexity of the posterior in high dimensional problems such as Emission Tomography hinders the computation of β as it depends on the full posterior, though some algorithms have been proposed to approximate it by sampling the posterior (see [99]). Here β is considered a parameter and is optimised in order to maximise *bias/variance* for the reconstruction of synthetic data.

6.B *Non-parametric Joint Entropy*

For the Differential Entropy in (6.4) to be differentiable, the probability distribution of the symbols can be estimated, for a given state, by a non-parametric kernel density estimation method:

$$p_{\boldsymbol{\lambda}}(\lambda) = \frac{1}{N_v} \sum_{v=1}^{N_v} \Phi\left(\frac{\lambda - \lambda_v}{\sigma_\lambda^2}\right) \tag{6.39}$$

Where Φ is a kernel function with bandwidth σ_λ^2. By the chain rule of differentiation and approximating the integral with the sum of rectangles of base Δ_λ, the gradient of the Entropy associated to a state $\boldsymbol{\lambda}$ is:

$$\frac{\partial}{\partial \lambda_v} E_{\boldsymbol{\lambda}} \simeq -\Delta_\lambda \sum_{j}^{M} \left(1 + \log p_{\boldsymbol{\lambda}}(\lambda_j)\right) \frac{\partial p_{\boldsymbol{\lambda}}(\lambda_j)}{\partial \lambda_v} \tag{6.40}$$

where, by differentiating (6.39):

$$\frac{\partial p_{\boldsymbol{\lambda}}(\lambda)}{\partial \lambda_v} = \frac{1}{N_v} \Phi'\left(\frac{\lambda - \lambda_v}{\sigma_\lambda^2}\right) \tag{6.41}$$

In the context of emission tomography, letting $\boldsymbol{\lambda}$ represent the unknown activity, considering an intra-subject image $\boldsymbol{y}$, such as a magnetic resonance (MR) image, the amount of information necessary to communicate $\boldsymbol{\lambda}$ is expressed by the conditional entropy

$$E_{\boldsymbol{\lambda}|\boldsymbol{y}} = E_{\boldsymbol{y},\boldsymbol{\lambda}} - E_{\boldsymbol{y}} \tag{6.42}$$

(which equals $E_{\boldsymbol{\lambda}}$ if $\boldsymbol{\lambda}$ and $\boldsymbol{y}$ are independent). It follows from (6.32) and (6.33) that

$$p(\boldsymbol{\lambda}) = \frac{1}{Z} e^{-\beta E_{\boldsymbol{\lambda}|\boldsymbol{y}}} \tag{6.43}$$

For the purpose of MAP estimation, minimisation of the Conditional Entropy $E_{\boldsymbol{\lambda}|\boldsymbol{y}}$ and of the Joint Entropy $E_{\boldsymbol{\lambda},\boldsymbol{y}}$ are equivalent.

$$\frac{\partial \log p(\boldsymbol{\lambda})}{\partial \lambda_v} = -\beta \frac{\partial}{\partial \lambda_v} E_{\boldsymbol{\lambda}|\boldsymbol{y}} = -\beta \frac{\partial}{\partial \lambda_v} E_{\boldsymbol{y},\boldsymbol{\lambda}} \tag{6.44}$$

Where:

$$E_{\boldsymbol{y},\boldsymbol{\lambda}} = -\int p_{\boldsymbol{\lambda}}(\lambda, y) \log p_{\boldsymbol{\lambda}}(\lambda, y) \, \mathrm{d}y \, \mathrm{d}\lambda \tag{6.45}$$

In order for the Joint Entropy to be differentiable, the probability distribution of the symbols can be estimated, for a given state of $\boldsymbol{\lambda}$, by a non-parametric kernel density estimation method:

$$p_{\boldsymbol{\lambda}}(\lambda, y) = \frac{1}{N_v} \sum_{v=1}^{B_v} \Phi\left(\frac{y - y_v}{\sigma_y^2}\right) \Phi\left(\frac{\lambda - \lambda_v}{\sigma_y^2}\right) \tag{6.46}$$

Approximating the integral in (6.45) with the sum of rectangular parallelepipeds of base $(\Delta_y, \Delta_\lambda)$, the gradient of the prior is proportional to:

$$\frac{\partial}{\partial \lambda_v} E_{\boldsymbol{\lambda},\boldsymbol{y}} \simeq -\Delta_y \Delta_\lambda \sum_{i,j}^{M^2} \left(1 + \log p_{\boldsymbol{\lambda}}(y_i, y_j)\right) \frac{\partial p_{\boldsymbol{\lambda}}(y_i, y_j)}{\partial \lambda_v} \tag{6.47}$$

With:

$$\frac{\partial p_{\boldsymbol{\lambda}}(\lambda, y)}{\partial \lambda_v} = \frac{1}{N_v} \Phi\left(\frac{y - y_v}{\sigma_y^2}\right) \Phi'\left(\frac{\lambda - \lambda_v}{\sigma_\lambda^2}\right) \tag{6.48}$$

(6.47) time a factor β is the gradient of the prior for MAP estimation for the non-parametric Joint Entropy method implemented in [92, 65, 63].

Chapter 7

Modelling the imaging devices

This chapter describes the gamma photon detection process in the probabilistic graphical framework. The models defined in this Chapter will be employed in Chapter 8 to extend the model of the emission process described in Chapter 2. The joint model that emerges in Chapter 8 has several applications including the use of the energy information for the estimation of the activity and an algorithm for the auto-adaptation of the emission imaging systems to disrupted photo-detectors. The current Chapter, after describing the gamma photon detection process by means of a scintillating camera and its probabilistic model, introduces two algorithms for the automated calibration of SPECT and PET detectors.

Scanners for planar and tomographic nuclear emission imaging detect gamma photons, recording their location of interaction and eventually their energy. Owing to the relatively low probability of interaction of gamma photons with matter, characteristic that makes the gamma photons suited to tomographic imaging of (relatively) large volumes, the detectors for gamma radiation, in order to maximise the detection efficiency, make use of high density materials to intercept the radiation. The most widespread detector design for the detection of gamma photons is the scintillating camera. In a scintillating camera, a high density crystal is engineered to emit low energy (visible or nearly visible) light in consequence of the interaction of a gamma photon. The scintillation is detected by one or more photo-detectors, producing an indirect measure of the location of the interaction and of the energy of the gamma photon.

The problems of determining the location of the interaction of each gamma photon and its energy are inverse problems. Once again, the uncertainty of the measurement plays an important role, thus the probabilistic formulation is essential in order to make good use of the information acquired.

A few articles have recently focused on optimum estimation of the location of interaction and energy of the gamma photons (see [100, 101]) by posing the problem in the probabilistic framework. This chapter firstly frames the problem in the probabilistic (graphical) framework, deriving the well known expression for the maximum likelihood estimator (MLE) of the location of interaction and energy of a gamma photon. The second part of the chapter focuses on the problem of measuring experimentally the characteristics of the imaging device, especially focusing on devices designed to discriminate the

depth-of-interaction (DOI).

Two algorithms for the characterisation of scintillating imaging devices are introduced; the first arises from the generative modelling approach and is based on a finite mixture model and on the Expectation Maximisation optimisation strategy; the second arises from concepts of manifold learning, introducing a new approach to the characterisation of the imaging devices, which brings artificial intelligence near the hardware of PET and SPECT scanners.

7.1 The scintillating camera

The scintillating gamma-camera, whose first design was proposed by Anger, is composed of a monolithic scintillator crystal coupled with an array of photo-detectors (see schematic in Fig. 7.1). The interaction of a γ-photon with the crystal determines, through a cascade of radiative and non-radiative processes, the emission of secondary photons. These are captured by the photo-detectors of the gamma-camera, producing electrical signals that are amplified by the front-end electronics and, in modern designs, digitalised and streamed to a digital computer [102]. The set of signals acquired by the photo-detectors bears information about the energy of the γ-photon and the location of the interaction with the scintillator crystal. The reconstruction of the coordinates of the interaction and of the energy from the measurement of secondary photons can be described as an *inverse problem*, where the forward problem consists in predicting the set of measurements given the location of the interaction and the energy of the γ-photon.

A number of reconstruction algorithms have been proposed, including heuristic methods, such as the centroid algorithm initially proposed by Anger [103][104], and *maximum likelihood estimation* (MLE).

MLE arises from probabilistic formulation of the reconstruction problem. It has proven superior to heuristic algorithms, eliminating non-linear distortions, such as the *barrel effect* (see [105] and section 7.5 of this chapter). Furthermore, the probabilistic formulation enables the definition of optimality criteria; MLE provides in fact optimum reconstructions in the sense of the $L2$-norm of the expected error (Cramér-Rao lower bound) [106][107]. Another important advantage of maximum likelihood estimation is that it enables 3-dimensional reconstruction of the coordinates of interaction, while heuristic methods are often limited, as is the case for the centroid method, to in-plane 2-dimensional reconstruction.

Though the development of efficient reconstruction strategies and hardware acceleration have overcome initial limitations due to computational complexity of maximum likelihood estimation, probabilistic methods for the estimation of the reconstruction of the events of interaction still have limited diffusion. Despite the optimality of MLE reconstruction, it is common experience to obtain visually better results, as described in section 7.6 of this chapter, with the centroid method. The reason for the poor performance of MLE, as clarified in 7.6, is that errors in the forward model propagate to the reconstruction. MLE, one may say, attempts to fit the data closely to the model, while the inherent imprecision of the heuristic methods hides the distortions introduced by the use of inexact forward models.

In order to obtain accurate reconstructions of the coordinates of interaction and energy of the detected γ-photons and to achieve the optimum resolution dictated by the Cramér-Rao bound, it is crucial to construct an accurate forward model of the camera. The forward model can be obtained with numerical simulations or by exhaustive experimental characterisation of the device.

A number of numerical methods for the prediction of the response of gamma-cameras have been described, including the simple solid-angle model based on Huilier's formula [106] and more accurate Monte Carlo light transport solvers [108]. However the complexity of the light transport processes limits the accuracy of numerical simulators. Especially the uncertain parametrisation of the materials and the high complexity of surface effects (such as the coupling between the scintillator crystal and the array of detectors) render the simulations highly approximative.

Experimental characterisation of the detector may enable accurate estimation of the forward model. By using a collimated beam one can force γ-photons to interact at a specific location of the camera plane (Fig. 7.2), obtaining an exhaustive characterisation of the camera response to the interaction of photons at that location. However the electrical signals generated by the array of photo-detectors is also a function of the depth-of-interaction, which cannot be imposed in an experimental set-up. The 2-dimensional characterisation of the response of the camera, obtained by neglecting the dependence of the response on the depth-of-interaction, is highly inaccurate causing distortions in the reconstruction, as illustrated in section 7.6 of this chapter.

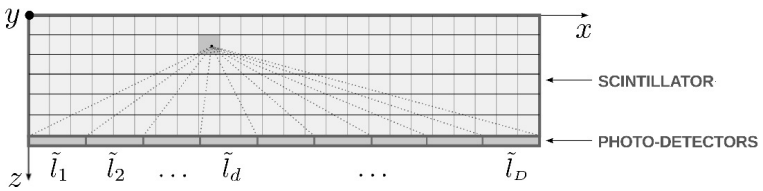


Figure 7.1: Schematic representation of the Anger gamma-camera. The monolithic scintillator is coupled with the array of photo-detectors indexed with $d = 1, \ldots, D$. The forward model of the camera is discretised in voxels indexed with j. The interaction of a γ-photon in j determines the noisy measurement vector $\mathbf{u} = \{u_1, \ldots, u_D\}$. The discrete forward model of the camera is characterised by the set of expected measurement vectors l_{dj} for γ-interaction in j.

3-dimensional characterisation is still an unsolved problem due to the computational challenges that is entails, associated with the unknown depth-of-interaction. In this chapter we describe a new methodology for exhaustive 3-dimensional characterisation of the forward model of gamma-cameras based on unsupervised learning techniques. Such powerful techniques, which have had large widespread in medical imaging applications, have not been applied to the reconstruction of the photon interaction events and to the characterisation of gamma-camera devices. One important advantage of the algorithm that we describe is that it is globally convergent, guaranteeing stability and fully automated operation.

While maximum likelihood estimation of the events of interaction may be considered *optimum*,

given that the forward model is accurate, Chapter 8 challenges this statement, integrating the generative model of the detection device described in the present chapter with the generative model of the tomographic system described in Chapters 2, 3, 4 in a unifying model.

7.2 Generative model of the imaging device

Let (x, y, z) be the 3-dimensional space coordinates with z being the direction normal to the camera imaging plane (Fig. 7.1) and let $\mathbf{u} = \{u_1, \ldots, u_d, \ldots, u_D\}$ be the measurement vector produced, in occurrence of a γ-interaction event, by the D photo-detectors, indexed by $d = 1, \ldots, D$. $\mathbf{u}$ is a random vector owing to the uncertainty associated with the measurement of the secondary photons by the D photo-detectors. We refer to the (continuous) forward model, denoted with $\tilde{l}_d(x, y, z)$, as the expected signal in d per unit of energy of the γ-photon, in case of interaction with the scintillator crystal in (x, y, z). Scintillator crystals present a linear relation between the average number of secondary photons emitted and the energy of the incoming gamma photon [109][106]. Therefore, by the thinning property of the Poisson distribution, the expected number of secondary photons in d for a gamma photon of energy E that interacts in (x, y, z) is $E\tilde{l}_d(x, y, z)$. In order to discretise the problems of reconstruction and model estimation, let us partition the spatial domain of the scintillator crystal in a regular grid of voxels, indexed with j, and assume that the response within each voxel is constant:

$$\mathbf{L} = l_{dj} \tag{7.1}$$

The elements of matrix $\mathbf{L} = \{l_{dj}\}$ constitute the parameters of the discrete forward model. Let us formulate the reconstruction of an event of interaction as an optimisation problem, entailing the maximisation of an energy functional parametrised by $\mathbf{L}$:

$$\hat{j}, \hat{E} = \arg\max_{j,E} \Psi_{\mathbf{L}}(j, E, \mathbf{u}) \tag{7.2}$$

Assuming that the main source of noise is the intrinsic uncertainty associated to the collection of secondary photons by the photo-detectors, the energy function can be written as a product of Poisson distributions (see 7.3). With this commonly employed model of the uncertainty, the MLE estimators (7.2) of the Poisson likelihood for j and E have the well known expressions (see derivation in paragraph 7.3):

$$\hat{j} = \arg\max_{j} \sum_{d}^{D} -l_{dj} + u_d \log l_{dj} \tag{7.3}$$

$$\hat{E} = \frac{\sum_d u_d}{\sum_d l_{d\hat{j}}} \tag{7.4}$$

The energy of the γ-photon (7.4) has a closed form expression dependent on $\hat{j}$. The location of interaction $\hat{j}$ (7.3) is independent of the energy but does not have a closed form solution. A number of algorithms for the maximisation of (7.3) have been described, including gradient free iterative optimisation [110] and (hardware accelerated) exhaustive search [106]. Details about specific implementations of

the maximisation algorithms are beyond the scope of this chapter; we are concerned here with obtaining accurate experimental measures of the parameters $\mathbf{L} = \{l_{dj}\}$ of the discrete forward model. In section 7.6 we use equations (7.3) and (7.4) with exhaustive search of j for reconstruction in order to compare different model parameters $\mathbf{L}$.

While it was convenient in the formulation of the reconstruction problem above to index all voxels with a single variable j, defining the matrix $\mathbf{L} = l_{dj}$ of model parameters, we use from here on a slightly different notation in order to simplify the indexes for the description of the algorithm for the estimation of the parameters of the forward model. The next section will focus on the estimation of the depth-dependent response of the camera at a single location (x^o, y^o). The parameters that describe the depth-dependent model in (x^o, y^o) correspond to a subset of the elements of $\mathbf{L}$ whose indexes depend on how the elements of $\mathbf{L}$ are mapped to spatial locations. In order to get rid of the specific indexing pattern, let us denote with l^o_{kd} the expected signal in d for γ-interaction at depth k, with $k \in \{1, \ldots, N_z\}$ indexing the voxels in (x^o, y^o) along axis z, as pictured in Fig. 7.2. $\mathbf{L^o} = l^o_{kd}$ is the set of parameters that describe the depth-dependent response of the camera in (x^o, y^o). The matrix $\mathbf{L}$ is filled by repeating the learning procedure described in the next section for each discrete location (x^o, y^o) according to the indexing pattern that maps the elements of $\mathbf{L}$ to spatial locations j.

7.3 Estimation of the events of interaction

The maximum likelihood estimators (MLE) of the location of interaction and energy of the γ-photon are the values $\hat{j}$ and $\hat{E}$ of j and E that maximise the log likelihood of the measurement $\log p(\mathbf{u}|E, j, L)$.

$$\Psi_{\mathbf{L}}(j, E, \mathbf{u}) = \log p(\mathbf{u}|E, j, L) \tag{7.5}$$

The likelihood function expresses the model of the uncertainty associated with the measurement. A common model for gamma-cameras is based on the assumption that the uncertainty associated with the discrete nature of light interaction dominates over other sources of noise. In this case the uncertainty is described by the Poisson distribution associated with counting the secondary photons. The measurements u_d are assumed to be independent conditionally to the location of interaction and energy. Such assumption, which corresponds to the following product, is essentially the statement that location of interaction and energy are the only causes of the measurements u_d.

$$p(\mathbf{u}|E, j, L) = \prod_{d=1}^{D} \frac{e^{-El_{dj}} {El_{dj}}^{u_d}}{u_d!} \tag{7.6}$$

$$\log p(\mathbf{u}|E, j, L) = E \sum_{d=1}^{D} -l_{dj} + \log E \sum_{d=1}^{D} u_d + \sum_{d=1}^{D} u_d \log l_{dj} \tag{7.7}$$

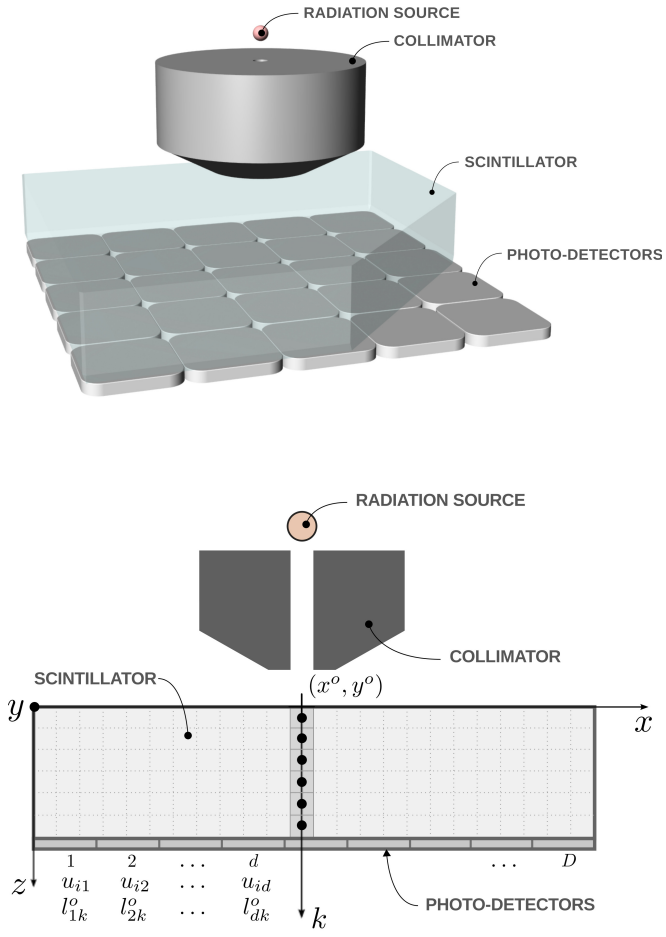


Figure 7.2: Schematic of the experimental set-up to learn the depth-dependent response of the gamma-camera. The collimated beam is focused at a given location (x^o, y^o). Though the depth-of-interaction z can not be imposed in the experiment, the depth-dependent response is learned from the signals produced by the photon interactions at (x^o, y^o).

Maximising with respect to j and E:

$$\frac{\partial}{\partial E} \log p(\mathbf{u}|E, \hat{j}, L) = \sum_{d=1}^{D} -l_{dj} + \frac{\sum_{d=1}^{D} u_d}{E} = 0 \tag{7.8}$$

$$\hat{E} = \frac{\sum_d u_d}{\sum_d l_{d\hat{j}}} \tag{7.9}$$

This expresses a closed form solution for the optimum energy. Notice that it depends on the optimum location of interaction $\hat{j}$. Let us express the maximiser with respect to j. Dropping the term that does not depend on j from equation (7.7), we can write:

$$\hat{j} = \arg\max_j E \sum_{d}^{D} -l_{dj} + u_d \log l_{dj} \tag{7.10}$$

For $E > 0$ the maximiser does not depend on E, hence:

$$\hat{j} = \arg\max_{j} \sum_{d}^{D} -l_{dj} + u_d \log l_{dj} \tag{7.11}$$

Notice that there is no closed form solution for the optimum location of interaction $\hat{j}$, however $\hat{j}$ is not a function of the optimum energy $\hat{E}$. As already boserved, the optimum energy (7.9) does have a closed form solution, dependent on $\hat{j}$. Therefore the optimum location of interaction and energy may be computed by finding the optimum location of interaction with (7.11) and then the optimum energy with (7.9). (7.11) can be optimised with gradient free optimisation algorithms, exhaustive search or more efficient direct search algorithms, such as the algorithm depicted in Fig. 7.3. Such algorithm and a version optimised for GPU implementation are described in [106]. With GPU acceleration, the 3-D (i.e. with DOI) reconstruction of the events of interaction can be performed in real-time at rates of 10^6 events per second.

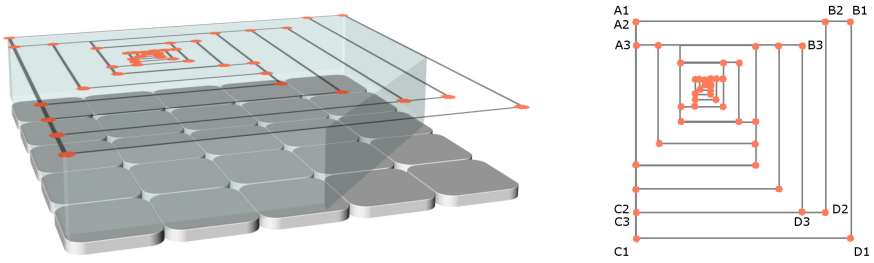


Figure 7.3: Fast globally-convergent algorithm for the direct search of the optimum location of interaction. The image depicts the algorithm described in [106] for the optimisation problem of (7.11) through direct search. At the first step, the algorithm computes the likelihood of the interaction at the four corners of the imaging device ($A1, B1, C1, D1$). The search grid is then refined selecting one of four rectangular regions depending on which of the four likelihoods is highest ($A1$ in this example). The algorithm proceeds until convergence to one point. The depicted algorithm searches the solution within a 2-D grid. The algorithm trivially extends to 3-D, as explained in [106].

7.4 Learning the characteristics of the imaging device: finite mixture model

7.4.1 Experimental set-up

Fig. 7.2 is a schematic representation of the set-up for the characterisation of the gamma-camera. The collimator generates a narrow beam that forces the γ-photons to interact with the crystal in (x^o, y^o). The signals produced by the photo-detectors for each photon interaction depend on the unknown depth-of-interaction. If one knew at which depth z each of the photon interactions occurs, it would be straightforward to characterise the average response of the camera to the interactions in a specific location

(x^o, y^o, z) (forward model) by averaging the response of a large number of interactions associated to (x^o, y^o, z). While x^o and y^o are imposed by the collimator, unfortunately the depth-of-interaction can not be forced. If one knew how the camera responds on average to the interaction in (x^o, y^o, z), the depth-of-interaction of each γ-photon could be easily estimated. However, when the forward model is unknown, the depth-of-interaction of the γ-photons can not be estimated. In order to estimate the forward model, in conclusion, it is required that the forward model is already known. In fact one way to proceed to estimate the forward model is to start with an approximate model, determine the depth-of-interaction of each photon according to the model and re-estimate the depth-dependent forward model according to the measurements now mapped to the depth-of-interaction. In 7.4.2 we derive an algorithm based on this iterative procedure and on a finite mixture probabilistic model. Such iterative procedures however are strongly dependent on the initial guess of the model parameters, posing the problem of establishing an initial value that is *close* to the solution. In other words, the optimisation problem associated with the estimation of the parameters of the forward model has multiple maxima. Owing to the problems associated with the initialisation of the aforementioned iterative procedure, in the following we adopt powerful unsupervised machine learning techniques to derive an algorithm with global convergence properties. The main idea is that though each measurement is a point in a high dimensional space, the measurements live on a low dimensional manifold. The algorithm learns the structure of the low dimensional embedding and exploits prior knowledge of the local density of data points along the manifold to learn the relation between the measurements and the depth-of-interaction. The resulting algorithm is surprisingly simple to implement and computationally efficient.

7.4.2 Finite mixture model

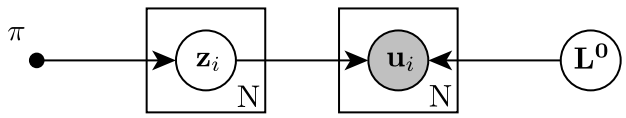


Figure 7.4: Probabilistic graphical representation of the mixture model for the set of N i.i.d. measurements $\{\mathbf{u}_i\}$ with corresponding latent locations of interaction $\{\mathbf{z}_i\}$, with $i = 1, \ldots, N$. For each measurement i, the latent location of interaction $\mathbf{z}_i$ is a 1-of-N_z unit vector. The matrix of parameters that characterise the imaging device $\mathbf{L^0}$ is the unknown variable of interest. $\pi = \{\pi_k\}$ is the prior probability associated to the latent variables, with $k = 1, \ldots, N_z$ being the running index over the elements of $\mathbf{z}_i$. The prior probability distribution π associated to the depth of interaction is identical for each measurement (therefore not a function of i) and given by Beer-Lambert exponential law, which expresses knowledge of the photon interaction process.

The model described in the following is an extension of the Poisson model described in 7.3. The

MLE estimators $\hat{j}$ and $\hat{E}$ of the location of interaction j and energy of the γ-photon E described in 7.3 maximise the Poisson likelihood, expressing the uncertainty associated with the measurement of secondary photons. The expressions of the MLE estimators found in 7.3 depend on the system model $\mathbf{L} = \{l_d j\}$. In the following the same model of the uncertainty is adopted for the purpose of estimating the parameters of the forward imaging model. Assuming that the locations of interaction along axis z are discrete, we obtain in the following a *finite mixture model* and derive the Expectation Maximisation (EM) optimisation algorithm for the parameters of the forward model. The probability distribution of the depth-of-interaction $p(z)$ is given by Beer-Lambert exponential law of absorption (7.28). Indexing with $k = 1, \ldots, N_z$ the allowed depths-of-interaction (Fig. 7.2), let us introduce a N_z dimensional binary random variable $\mathbf{z}$ having a 1-of-N_z representation in which a particular element z_k is equal to 1 and all other elements are equal to 0. The values of z_k therefore satisfy $\mathrm{z}_k \in \{0, 1\}$ and $\sum_k \mathrm{z}_k = 1$, and there are N_z possible states for vector $\mathbf{z}$ according to which element is non-zero. Let π_k be the prior probability that the interaction occurs in voxel k:

$$p(\mathrm{z}_k = 1) = \pi_k \tag{7.12}$$

Section 7.5.1 will describe one choice of prior probability distribution based on the mechanics of the photon interaction (Beer Lambert exponential law). Because $\mathbf{z}$ is a 1-of-N_z representation, we can also write this distribution in the form

$$p(\mathbf{z}) = \prod_{k=1}^{N_z} {\pi_k}^{\mathrm{z}_k} \tag{7.13}$$

The conditional distribution for $\mathbf{u}$ given a particular value for $\mathbf{z}$, adopting the Poisson model of the uncertainty described in 7.3, is a product of Poisson distributions:

$$p(\mathbf{u}|\mathrm{z}_k = 1) = \prod_{d=1}^{D} \mathcal{P}\left(u_{id}; \lambda_{k,d}\right) \tag{7.14}$$

which, similarly, can be written in the form

$$p(\mathbf{u}|\mathbf{z}) = \prod_{k=1}^{N_z} \left[\prod_{d=1}^{D} \mathcal{P}\left(u_{id}; \lambda_{k,d}\right) \right]^{\mathrm{z}_k} \tag{7.15}$$

Integrating over the depth-of-interaction, the likelihood of the measurement is

$$p(\mathbf{u}) = \sum_{k=1}^{N_z} \left[\pi_k \prod_{d=1}^{D} \mathcal{P}\left(u_{id}; \lambda_{k,d}\right) \right] \tag{7.16}$$

Considering N measurements $\mathbf{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_i, \ldots, \mathbf{u}_N\}$ at location (x^o, y^o), each data point is drawn independently from the distribution (7.16). We can express the mixture model for this i.i.d. data set using the graphical representation in 7.4. From (7.16) the $\log$ of the likelihood function for the training data $\mathbf{U}$ is given by

$$\ln p(\mathbf{U}|\mathbf{L^0}) = \sum_{i=1}^{N} \ln \sum_{k=1}^{N_z} \left[\pi_k \prod_{d=1}^{D} \mathcal{P}\left(u_{id}; l^0_{kd}\right) \right] \tag{7.17}$$

The summation that appears inside the logarithm produces a complicated expression for the maximum likelihood solution of the parameters l^o_{kd}, excluding the existence of a closed form solution for l^o_{kd} that maximises (7.17) (see [111] p.439). The EM algorithm, however, once again, provides an iterative update formula for l^o_{kd} which is free from nuisance parameters.

7.4.3 Expectation Maximisation

Considering $\mathbf{U}$ as the incomplete data of the complete data set $\{\mathbf{U}, \mathbf{Z}\}$, with $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_i, \ldots, \mathbf{z}_N\}$, let us consider the problem of maximising the likelihood of the complete data. From (7.13) and (7.15), this likelihood takes the form:

$$p(\mathbf{U}, \mathbf{Z}|\mathbf{L^0}) = \prod_{i=1}^{N} \prod_{k=1}^{N_z} \prod_{d=1}^{D} {\pi_k}^{z_{ik}} \mathcal{P}\left(u_{i,d}; l^0_{kd}\right)^{z_{ik}} \tag{7.18}$$

where z_{ik} denotes the k_{th} component of $\mathbf{z}_i$. Taking the logarithm, we obtain

$$\ln p(\mathbf{U}, \mathbf{Z}|\mathbf{L^0}) = \sum_{i=1}^{N} \sum_{k=1}^{N_z} z_{ik} \left[\ln \pi_k + \sum_{d=1}^{D} \ln \mathcal{P}\left(u_{id}; l^0_{kd}\right)\right] \tag{7.19}$$

The complete data $\log$ likelihood function could be maximised trivially in closed form if $\mathbf{Z}$ were known. However in the missing-data formulation, $\mathbf{Z}$ are unknown latent variables. As the EM algorithm prescribes, we consider the expectation, with respect to the posterior distribution of the latent variables, of the complete-data $\log$ likelihood: from (7.13) and (7.15) and applying the Bayes' theorem

$$p(\mathbf{Z}|\mathbf{U}, \mathbf{L^0}) \propto \prod_{i=1}^{N} \prod_{k=1}^{N_z} \left[\pi_k \prod_{d=1}^{D} \mathcal{P}\left(u_{id}; l^0_{kd}\right)\right]^{z_{ik}} \tag{7.20}$$

The expected value of the indicator variable z_{ik} under this posterior distribution is given by

$$\mathbb{E}[z_{i,k}] = \frac{\sum_{\mathbf{z}_i} z_{i,k} \prod_{k'=1}^{N_z} \left[\pi_{k'} \prod_{d=1}^{D} \mathcal{P}\left(u_{id}; l^0_{k',d}\right)\right]^{z_{ik'}}}{\sum_{\mathbf{z}_i} \prod_{k''=1}^{N_z} \left[\pi_{k''} \prod_{d=1}^{D} \mathcal{P}\left(u_{id}; l^0_{k''d}\right)\right]^{z_{ik''}}} = \frac{\pi_k \prod_{d=1}^{D} \mathcal{P}\left(u_{id}; \lambda_{kd}\right)}{\sum_{k'=1}^{N_z} \pi_{k'} \prod_{d=1}^{D} \mathcal{P}\left(u_{id}; \lambda_{k'd}\right)} \triangleq h_{ik} \tag{7.21}$$

referred to, in context of finite mixture models, as the *responsibility* of the mixture component k for data point $\mathbf{u}_i$. The expected value of the complete-data $\log$ likelihood function is therefore given by

$$\begin{aligned} Q(\mathbf{L^0}|\mathbf{L^0}^{\text{old}}) &= \mathbb{E}_{\mathbf{z}}\left[\ln p(\mathbf{Z}, \mathbf{U}|\mathbf{L^0}^{\text{old}})\right] \\ &= \sum_{i=1}^{N} \sum_{k=1}^{N_z} h_{ik} \left[\ln \pi_k + \sum_{d=1}^{D} \ln \mathcal{P}\left(u_{id}; l^0_{kd}\right)\right] \end{aligned} \tag{7.22}$$

This is maximum when the gradient with respect to L^0 is zero

$$\frac{\partial Q(\mathbf{L^0}|\mathbf{L^0}^{\text{old}})}{\partial l^0_{kd}} = \sum_{i=1}^{N} \left[h_{ik} \frac{\partial}{\partial l_{kd}} \ln \mathcal{P}\left(u_{id}; l^0_{kd}\right)\right] = 0 \tag{7.23}$$

solving for l^0_{kd}

$$l_{kd}^{0\ \text{new}} = \frac{\sum_{i=1}^{N} h_{ik} u_{id}}{\sum_{i=1}^{N} h_{ik}} \tag{7.24}$$

The EM algorithm for the estimation of the model parameters l_{kd}^0, in summary, consists in choosing an initial guess and iterating until convergence (7.24) (M-step), with h_{ik} expressed by eq. (7.21) (E-step):

$$h_{ik} = \frac{\pi_k \prod_{d=1}^{D} \mathcal{P}\left(u_{id}; \lambda_{kd}\right)}{\sum_{k'=1}^{N_z} \pi_{k'} \prod_{d=1}^{D} \mathcal{P}\left(u_{id}; \lambda_{k'd}\right)} \tag{7.25}$$

Starting from an initial guess, the EM algorithm of (7.24) and (7.25) provides a series of estimates of the parameters of the forward model with increasing likelihood (the increase of likelihood at each step of the EM algorithm is guaranteed to be ≥ 0 [12]). Therefore the algorithm converges to a local maximiser of the likelihood. While for other problems, such as the problem discussed in Chapter 4, the locally convergent algorithm is often sufficient given a reasonable initial guess, as discussed in the experiments section of this chapter 7.6, for this particular problem, the selection of the initial value of the parameters is critical, making the algorithm unstable. The next paragraph describes a different approach based on manifold-learning techniques.

7.5 Density of the locally linear embedding - DLLE

Let us consider the set of N measurements $\mathbf{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_i \ldots, \mathbf{u}_N\}$, with $\mathbf{u}_i \in \mathbb{R}^D$. Dimensionality reduction algorithms provide a reduced dimensional representation $\mathbf{R} = \{\mathbf{r}_1, \ldots, \mathbf{r}_i, \ldots, \mathbf{r}_N\}, \mathbf{r}_i \in \mathbb{R}^d$ of the original dataset $\mathbf{U}$. The underlying assumption behind these algorithms is that $\mathbf{U}$ lies on a manifold $\mathcal{M}$ embedded in $\mathbb{R}^D$ with intrinsic dimensionality d^*. The estimated dimension d of d^* is provided by the user as a parameter to the algorithm. We adopt here the LLE algorithm [112] to learn the structure of the manifold. Although the details of LLE can be found in [112], we provide a brief description in order to fully explain our method.

The first step of LLE is to solve for the location of each point $\mathbf{u}_i \in \mathbb{R}^D$ in terms of its n nearest neighbours. This step is performed simultaneously for every point $\mathbf{u}_i \in \mathbf{U}$ by selecting the n-nearest neighbours of each data point and solving

$$\hat{\mathbf{W}} = \arg\min_{\mathbf{W}} \sum_i \left| \mathbf{u}_i - \sum_j W_{ij} \mathbf{u}_j \right|^2 \tag{7.26}$$

subject to two constraints: first, that each data point $\mathbf{u}_i$ is reconstructed only from its neighbours, enforcing $W_{ij} = 0$ if $\mathbf{u}_j$ does not belong to the set of neighbours of $\mathbf{u}_i$; second, that the rows of the weight matrix sum to one $\sum_j W_{ij} = 1$ for every i. The minimisation (7.26) with these two constraints finds the

coefficients of the weighting matrix $\mathbf{W}$ that reconstruct optimally each data point as a linear combination of its neighbours. In the implementation of LLE that we adopt here, data points are reconstructed from their n nearest neighbours as measured by the Euclidean distance. For such implementation of LLE, the algorithm has only one free parameter: the number of neighbours n. The minimisation problem (7.26) has a closed form solution, as described in [113].

Once the reconstruction weights $\hat{\mathbf{W}}$ have been estimated, each high-dimensional observation $\mathbf{u}_i$ is mapped to a low-dimensional vector $\mathbf{r}_i$ representing global internal coordinates *on the manifold*. This is done by choosing d-dimensional coordinates $\mathbf{r}_i$ to minimise the embedding cost function

$$\hat{\mathbf{R}} = \arg\min_{\mathbf{R}} \sum_i \left| \mathbf{r}_i - \sum_j \hat{W}_{ij} \mathbf{r}_j \right|^2 \tag{7.27}$$

Such embedding cost, defining a quadratic form in the vectors $\mathbf{r}_i$, can be minimised by solving a sparse $D \times D$ eigenvalue problem [113], whose bottom d non-zero eigenvectors provide an ordered set of orthogonal coordinates centred on the origin. Implementation of the algorithm, detailed in [113], is straightforward. We set here $d = 1$ in order to capture the 1-dimensional variations associated with the depth-of-interaction. The scalar measure r_i denotes the location of data point i on the manifold.

7.5.1 Partitioning of the manifold

In order to establish a relation between the internal coordinates of the manifold and the depth-of-interaction, we exploit the prior information about the density of data points on the manifold.

Beer-Lambert exponential law of absorption expresses the probability of interaction as a function of z for the collimated photons travelling in the direction of the z axis

$$p(z) = \alpha_{\tilde{E}} e^{-\alpha_{\tilde{E}} z} \tag{7.28}$$

with $\alpha_{\tilde{E}}$ being the attenuation coefficient of the scintillator crystal for the γ-source of energy $\tilde{E}$ utilised for the experimental characterisation. Though a number of algorithms may be devised for the purpose of matching the prior density of z with the density of points on the manifold, let us introduce here a simple algorithm based on the discretisation of the depth-of-interaction in N_z locations. Indexing with $k = 1, \ldots, N_z$ the voxels along axis z (see Fig. 7.2), and letting Δz be the size of the voxels along direction z, the probability that a γ-photon interacts in voxel k is

$$\pi_k = \frac{\int_{(k-1)\Delta z}^{k\Delta z} p(z)\mathrm{d}z}{\sum_{k=1}^{N_z} \int_{(k-1)\Delta z}^{k\Delta z} p(z)\mathrm{d}z} = \frac{e^{-\alpha_{\tilde{E}}(k-1)\Delta z} - e^{-\alpha_{\tilde{E}} k \Delta z}}{1 - e^{-\alpha_{\tilde{E}} N_z \Delta z}} \tag{7.29}$$

The manifold is subdivided in k partitions corresponding to the depth-of-interaction by matching the density of data points on the manifold with the expected a-priori number of data points in each voxel

k. As the scalars r_i represent a measure of the location of point i on the manifold, the manifold is partitioned by finding $N_z + 1$ thresholds $t_h, h = 0, \ldots, N_z$ for the internal measure on the manifold in order for the density of points within each interval $[t_{k-1}, t_k]$ to match the prior probability distribution π_k for all voxels $k = 1, \ldots, N_z$. The matching condition has the following expression in terms of unit step functions $H(x)$:

$$t_h = \min r_j : \left[\frac{1}{N} \sum_{i=1}^{N} H(r_i - r_j) \right] > \pi_h, h = 1, \ldots, N_z \tag{7.30}$$

with $t_0 = \min r_i$. Such threshold values t_h are computed simply by sorting the values of r_i, setting $t_0 = \min r_i$, then t_1 to the value r_i such that the number of points that are smaller than r_i is $floor(N \frac{\pi_1}{\sum_k \pi_k})$, t_2 to the value of r_i such that the number of points that are smaller than r_i is $floor(N \frac{\pi_1 + \pi_2}{\sum_k \pi_k})$, and so on. Finally, given the partition of the manifold $t_h, h = 0, \ldots, N_z$, each data point i is assigned to location k according to the value r_i of its low dimensional representation. Let us introduce a N_z dimensional binary variable $\mathbf{z}$ having a 1-of-N_z representation in which a particular element z_k is equal to 1 and all other elements are equal to 0. The values of z_k therefore satisfy $\mathrm{z}_k \in \{0, 1\}$ and $\sum_k \mathrm{z}_k = 1$, and there are N_z possible states for vector $\mathbf{z}$ according to which element is non-zero. $\mathbf{z}$ may be considered as a hidden label indicating the discrete depth-of-interaction of measurement $\mathbf{u}$. Let $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_i, \ldots, \mathbf{z}_N\}$ with $\mathbf{z}_i = \{\mathrm{z}_{ik}\}$ be the set of hidden labels for the N measurements and $\hat{\mathrm{z}}_{ik}$ their estimates according to the partition of the manifold:

$$\hat{\mathrm{z}}_{ik} = H(r_i - t_{k-1}) H(t_k - r_i) \tag{7.31}$$

The forward model l^o_{kd} is calculated as the expectation vector for depth-of-interaction k according to the estimated depth-of-interaction labels:

$$l^o_{kd} = \frac{\sum_{i=1}^{N} u_{id} \hat{\mathrm{z}}_{ik}}{\sum_{i=1}^{N} \hat{\mathrm{z}}_{ik}} \tag{7.32}$$

7.5.2 Summary of the algorithm

For each discrete location (x^o, y^o)

- Acquire N data points.
- Learn the structure of the 1-dimensional manifold (7.26).
- Project the data on the manifold (7.27).

- Partition the manifold in order to match the prior probability distribution of the depth-of-interaction with the density of data points (7.30).
- Estimate the depth-of-interaction of each γ-photon based on its position on the manifold (7.31).
- Estimate the depth-dependent response of the gamma-camera by averaging the signals produced by all photons with the same label (7.32).

7.6 Experiments

7.6.1 Analytical model of the gamma-camera

In order to evaluate the algorithm described in the previous section, we make use in the following of an analytical ground-truth forward model of the gamma-camera. We adopt the simple solid angle model often used as the forward model for reconstruction [106][100]. Though such simplistic model neglects scatter, reflection and refraction at the boundaries of the scintillator crystal, it allows us to generate efficiently the expectations for continuously varying depth-of-interaction z. The number of expected secondary photons detected by each photo-detector d for an interaction in (x, y, z) is considered to be proportional to the solid angle $\theta_d(x, y, z)$ subtended by the surface of the photo-detector to the interaction point in (x, y, z). The solid angle is calculated exactly with Huilier's formula for the triangular planes obtained by triangular meshing of the surface of the photo-detectors [106]. Emission of secondary photons is assumed to be isotropic: the expected photon counts at each location d are proportional to the solid angle $\theta_d(x, y, z)$, to the light yield of the scintillator ξ and to the energy of the γ-photon E:

$$\tilde{l}_d(x, y, z) = \frac{\theta_d(x, y, z)\xi E}{4\pi} \tag{7.33}$$

7.6.2 Learning the forward model

The parameters of the simulation emulate the characteristics of a small gamma-camera composed of $D = 25$ photo-detectors of size $10mm \times 10mm$ displaced in a 5 by 5 square array and coupled with a $50mm \times 50mm \times 10mm$ NaI(T1) scintillator crystal (Fig. 7.7(A) and Fig. 7.1). The scintillator crystal is partitioned into $128 \times 128 \times 10$ non-isotropic voxels. For each of the 128×128 locations (x_o, y_o), $N = 10000$ events of interaction of γ-photons with energy $E = 140\ KeV$ are simulated by sampling the depth-of-interaction according to (7.28) with $\alpha_{140Kev} = 2.4\ cm^{-1}$ (see [114] for typical parameters of NaI(T1)). The signals u_{id} for each of the i events are obtained by calculating the ideal response with the simulator described in the previous section, with scintillator light yield $\xi = 38\, photons/KeV$ [114], and applying Poisson noise. For each (x_o, y_o) location, the response of the gamma-camera is estimated with two methods:

1. *2-d averaging*: The responses for the N photon interactions are averaged, producing a 2-dimensional model of the camera. This may corresponds to applying (7.32) with $N_z = 1$.

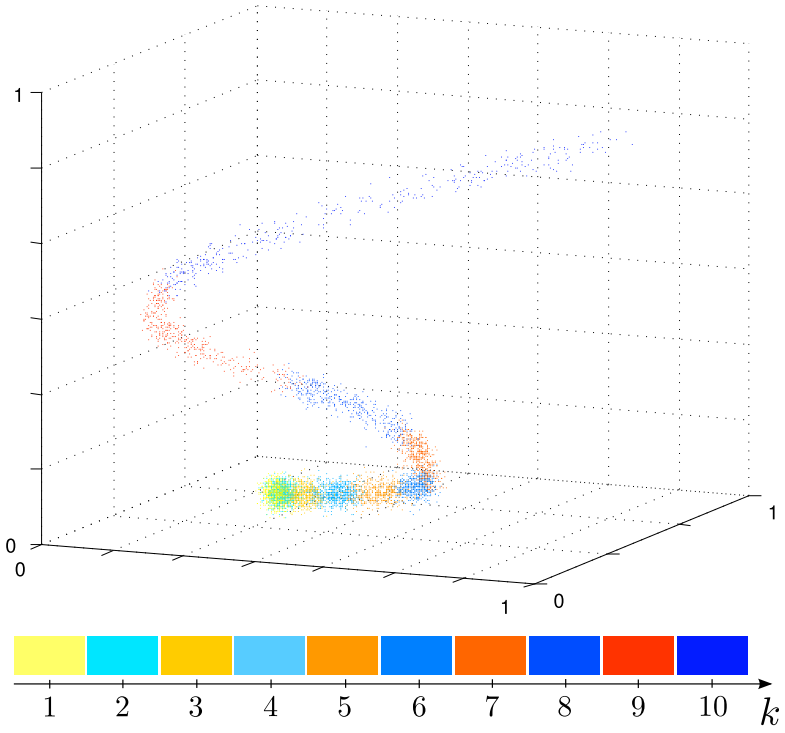


Figure 7.5: Scatter plot of the data points projected on the manifold of dimension $d = 3$. Data points are coloured according to the ground-truth depth-of-interaction label k. From this visualisation of the data projected on the 3-dimensional manifold it can be observed that the data live on a 1-dimensional manifold. The density of data points decreases exponentially with increasing depth-of-interaction. Figure 7.6 represents the projection for $d = 1$

.

2. *Unsupervised learning*: The 3-dimensional response of the camera is learned from the N measurements with the algorithm described in 7.5. The number of neighbours is set to $n = 12$.

Fig. 7.5 shows the scatter plot of the N data points projected on the 3-dimensional manifold $\mathbf{r}_i$ ($d = 3$). Each point is coloured according to the depth-of-interaction k at which it interacts, obtained by thresholding the ground-truth continuous depth-of-interaction. From the projection on the 3-dimensional manifold in Fig. 7.5, the manifold appears to be, as expected, intrinsically 1-dimensional and the density of data points decreases as the depth-of-interaction increases. Fig. 7.6 reports the scatter plot of the projection of the data points on the manifold of dimension $d = 1$ and the thresholds $t_h = \{t_0, t_1, \ldots, t_{N_z}\}$ calculated according to (7.30). Data points in Fig. 7.6(B) are coloured according to the ground-truth depth-of-interaction while in Fig. 7.6(A) according to the value of r_i relative to $t_h = \{t_0, t_1, \ldots, t_{N_z}\}$ - equation (7.31). As described in section 7.5, the parameters of the forward model are estimated by averaging all data points assigned to the same depth-of-interaction - equation (7.32).

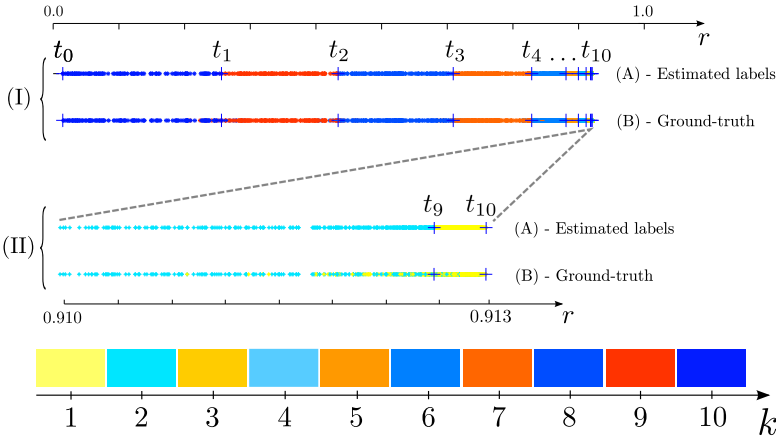


Figure 7.6: Scatter plot of the data points projected on the manifold of dimension $d = 1$. The location of each data-point i on the manifold is characterised by scalar r_i. (I): All N training data points. (II): Close up for r near 1, corresponding to depth-of-interaction $k = 9$ and $k = 10$ voxels. $t_h = \{t_0, t_1, \ldots, t_{N_z}\}$ partition the manifold in $N_z = 10$ regions according to (7.30). In plot (A) the data points are coloured according to the value of r_i (estimated depth-of-interaction). In plot (B) data points are coloured according to the true depth-of-interaction.

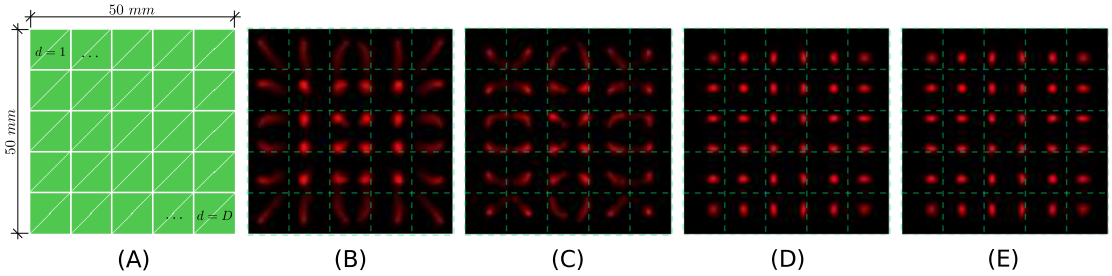


Figure 7.7: (A) The experiments are based on a $50mm \times 50mm \times 10mm$ scintillator crystal coupled with an array of $D = 25$ square photo-detectors of size $10mm \times 10mm$. The ground-truth response of the gamma-camera is obtained with the solid angle model and computed by triangulating the photo-detector elements. Simulation of $5M$ events of interaction at 36 discrete locations. (B) Reconstruction with the centre-of-mass algorithm with forward model obtained by experimental characterisation of the camera. (C) MLE reconstruction with 2-D forward model obtained by experimental characterisation of the camera. (D) MLE reconstruction with the 3-D forward model learned with the manifold learning algorithm. (E) MLE reconstruction with the 3-D forward model learned with the manifold learning algorithm and refined with the EM algorithm (7.21)(7.24).

7.6.3 MLE reconstruction and model mismatch

In this last section we evaluate, through numerical simulation, the effect of using the learned 3-dimensional forward model and the approximated 2-dimensional model for the reconstruction of the

events of interaction. $5\ Million$ interactions are simulated by generating random (x, y, z) locations of interaction as follows:

While less then $5\ M$ events of interaction:

1. draw (x, y) location from uniform distribution on the surface of the gamma-camera;
2. discard the event if outside of the mask defined by the white pattern in figure 7.7(B);
3. draw (z) according to the Beer-Lambert distribution (7.28);
4. compute the noiseless measurement with the simulator 7.6.1;
5. apply Poisson noise.

Reconstructions of the $5\ M$ events of interaction are visualised in Fig. 7.7. The intensity of the images in 7.7 represents the number of events associated to each of the 128×128 pixels. The events are reconstructed with MLE (7.3) (7.4) using the ground-truth forward discrete model (Fig. 7.7(C)) and the 2-dimensional (Fig. 7.7(D)) and 3-dimensional (Fig. 7.7(F)) estimated forward models. For reference Fig. 7.7(E) reports the reconstruction obtained with the centroid algorithm [104] with the 2-dimensional estimated forward model. Finally the parameters of the 3-dimensional model estimated with the learning algorithm are refined with 100 iterations of the EM algorithm described in 7.4.2 and Fig. 7.7(G) reports the reconstruction obtained with the refined model.

Comparison of the reconstruction obtained with the centroid algorithm in Fig. 7.7(E) with the reference reconstruction in 7.7(C) shows the distortion produced by the centroid method, underlined by the two circular dark regions that are shifted towards the centre of the image. MLE reconstruction with the 2-dimensional model artificially assigns events to the regions located near the centre of the photo-detectors, as highlighted by Fig. 7.7(D). The reconstruction obtained with the 3-dimensional estimated forward model shows how the recovery of the forward model is highly accurate, producing a reconstruction that is visually undistinguishable from the MLE reconstruction obtained with the ground-truth forward model in Fig. 7.7(C).

Finally, the refinement of the forward model obtained with the EM algorithm does not produce visible improvements in the reconstruction.

In summary, the first part of this chapter has framed in the probabilistic graphical framework the inverse problem of estimating the coordinates of interaction and the energy of the gamma photons detected with a scintillating camera. The second part of the chapter has presented two algorithms for the estimation of the depth-dependent characteristics of monolithic gamma-cameras. The first algorithm, based on the mixture of Poisson distributions model and the EM algorithm, is the natural solution that arises from the formulation of the problem in the probabilistic framework. However such algorithm suffers from local minima. The second algorithm, based on concepts of manifold learning, overcomes the limitations of the EM-based algorithm. The LLE manifold learning algorithm is employed to learn

the characteristics of the camera and the density of data points on the manifold is matched to the prior probability distribution of the depth-of-interaction in order to establish a relationship between the position of the data points on the manifold and the depth-of-interaction. This new concept shows promising results in the simulation study that is presented here, recovering the forward imaging model with high accuracy. The unique global convergence properties of the algorithm make it stable and reliable. The absence of nuisance parameters (exception given for the number of neighbours n, which has been found to be non-influential as long as the number of training photons N is sufficiently large) and the stability of the algorithm make it suitable for fully automated characterisation of the cameras.

The inverse problems described in this chapter and the inverse problem of tomographic image reconstruction are inherently coupled, but often considered in separate contexts. The description of the imaging device in the probabilistic graphical framework enables us to consider the two problems jointly. This is the focus of the next chapter.

Chapter 8

Unified computational model of scatter

The elements of the system matrix of the emission imaging system $A = \{a_{vd}\}$, as described in chapter 2-paragraph 2.3, express the probability that a photon emitted from voxel v is detected in detector bin d in SPECT or by the pair of detectors corresponding to the line of response d in PET. The system matrix encompasses the geometry of the system (position of the detectors relative to the imaging volume; collimator), the attenuation of γ-rays throughout the body of the patient, scattering of photons and randoms. As a photon emitted in v propagates through the body, its direction can be altered by random scattering events. The photon may then be detected in d after one or more scattering events, though having lost part of its energy at the scattering sites. If one had sufficient computational resources, scatter and randoms would be accounted for by expressing in each element of P the probability that a photon emitted in v is detected in d accounting for scatter events. This involves integrating over all possible scattering sites and accounting for multiple scatter events. It is immediately understood that the computational complexity is too high.

Using a projector and back-projector that only account for un-scattered photons, the image quality is degraded. In fact each detected pair of photons, regardless of whether one of the two or both have scattered or not, is considered, in PET, to be emitted somewhere along the line of response corresponding to the detector pair d excited by the two photons. Similarly, in SPECT, a photon detected in detector bin d is assumed to be emitted from the cone of response centred in d (with probability weighted by the Point Spread Function - PSF) regardless of whether the photon has scattered or not. However, often the γ-cameras measure not only the coordinates of interaction of the γ-photons, but also their energy (though both measurements are subject to uncertainty). The energy information can be used to discriminate and discard scattered photons. The simplest algorithm to reduce the degradation of image quality due to the use of simple forward models that do not account for scatter, consists in applying a *threshold* to the measured energy, discarding photons with energy that falls below a threshold (Fig. 8.5 reports in blue a typical measured energy spectrum, the green continuous line is the spectrum of un-scattered photons and the red continuous curve is the spectrum of scattered photons). However, due to the noise associated to the measurement of energy, some of the scattered photons are considered un-scattered and vice versa.

In order to reduce the effect of confounded photons, the Triple Energy Window (TEW) algorithm uses empirical formulae to correct the number of un-scattered photons obtained for each detector location by energy thresholding. In both algorithms, the correction factor for each detector bin is derived from the number of counts associated to photons with energy below a lower threshold and above a higher threshold (the higher threshold is adopted only in case of multi-spectral radio-ligands with higher energy yield). The rationale of this methods is that the number of photons with detected energy falling outside of the energy window for the un-scattered photons is related to the number of photons that have scattered but whose measured energy falls within the un-scattered region due to uncertainty in the measurement of energy. The empirical correction factors reduce the counts in the un-scattered region accordingly. However such algorithms are not derived from a model.

The question that arises is, if one could compute the system matrix that accounts for the scattered photons, what would be the optimal use of the energy information from energy resolving gamma cameras. The model described in the following addresses this question, though, on the side, also suggesting algorithms for scatter correction that do not require the computation of the forward projection for the scattered photons.

The trivial answer to the question of how make use the energy information, is to extend the model of the emission imaging system described in chapter 7, considering the energy as an additional dimension of the measurement and discretising it just like the coordinates of interaction, let's say in N_E bins. This would require a slight modification of the model described in chapter 2-paragraph 2.3 (equation (2.45)). The number of detector bins would be $N_E \times N_d$ and the system matrix, of size $[N_v \times N_d N_E]$ would express the probability that a photon emitted in v is detected in d and the measured energy is E_e, with index $e = 1, 2, \ldots, N_E$. The complexity of such model is dramatically higher than the complexity of the model that does not account for the energy measurements. One may say that this is a case of the curse of dimensionality. However the model, as expressed, hides certain factorisations, as discussed in this chapter.

The chapter is divided in two parts. The first paragraph describes the joint model that occurs when joining the probabilistic model of the tomographic imaging system of chapter 7 with the probabilistic model of the gamma detectors of chapter 7. At this stage, scatter photons are disregarded. The joint model suggests an algorithm for the automatic compensation of disrupted detectors.

The second paragraph extends the joint model described in the first paragraph, accounting for scattered photons. While the model described in this chapter may be applied, nearly unchanged, to PET and SPECT, it is described in the following in the context of SPECT only.

8.1 Joint estimation of the activity and of the events of interaction

As described in Chapter 7, a monolithic γ camera is composed of a collimator, a scintillator crystal, and a planar array of photo-detectors. When a γ photon annihilates in the scintillator crystal, a cascade of decay

processes determines the emission of low energy photons (secondary photons), whose number is of the order of a few thousand (yield of the scintillator crystal). The secondary photons are detected by the array of photo-detectors, which produce electrical signals in proportion to the number of detected photons. Such signals are correlated to the location and energy of the γ interaction. By employing a model of the response of the camera, one may infer the location of the interaction event and the energy of the γ photon, computing task often referred to as *reconstruction of the event of interaction*. Reconstruction of the events of interaction has been formulated in the probabilistic framework in chapter 7, modelling the uncertainty associated to the measurement of the secondary photons produced by the scintillator crystal by means of the Poisson distribution. The coordinates of interaction and energy are commonly estimated with the Maximum Likelihood Estimator (MLE) for such Poisson model, as described in chapter 7-paragraph 7.3. The position of iteration is a continuous function over the plane of the camera, however the computational model described in chapter 2-paragraph 2.3 for the purpose of tomographic reconstruction of the activity assumes discrete locations of interaction. Such model consists again of a product of Poisson distributions. Both the models described in chapter 2 and in chapter 7 are non-strictly concave and can be optimised with gradient ascent type algorithms such as MLEM. Due to the statistics associated with the detection of secondary photons, the locations of interaction of the γ photons with the camera have a degree of uncertainty (the scintillator crystals produce typical yields of a few thousand secondary photons for the energies of interest in emission computed tomography, determining typically around $5mm$ full-width-half-maximum uncertainty). Such uncertainty can be accounted for in the reconstruction of the activity, and embedded in the system matrix P. In fact the probability that a photon emitted in v is *detected* in d is now not only dependent on the attenuation and collimator, but also on the probability that it *interacts* somewhere nearby d but it is *detected* in d. It is often assumed that the uncertainty about the location of interaction is a *normal distribution* over the plane of the camera centred in the MLE estimate d. The probability distribution of the location of interaction is in fact asymptotically (high number of secondary photons) normal [115], with variance equal to the inverse of the Fisher Information Matrix (FIM) associated with the camera model. Assuming normality, the uncertainty about the location of the interaction events may be accounted for simply by adding a convolution step in the projection and back-projection; the kernel of the convolution operator obtained by inverting the FIM and is often referred to as *intrinsic resolution*. The uncertainty is, however, in general strongly non-uniform across the camera due to the finite size of the secondary photo-detectors and normality only holds asymptotically.

An important result of probability theory is the concept of *complete statistics*, which holds for normal distributions. In our case the results of *complete statistics* tell us that if the probability distribution of the location of interaction of an event, associated to the measurement of secondary photons is normal, then the parameters of the normal are sufficient to describe the measurement. However, since the Poisson model does not have sufficient statistics [116] (except in the asymptotic case), the probability distribution

of the location of interaction does not have a synthetic representation, thus some information is lost when one accounts for the MLE estimate of the location of interaction forgetting about the actual counts of the secondary photons. Rather than considering the MLE estimates of the locations of the interaction events, in the following we express explicitly the joint probability distribution of the activity, and of the location of all events of interaction.

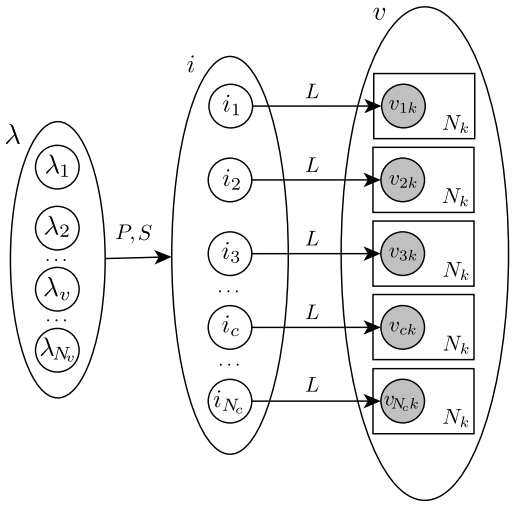


Figure 8.1: DAG of the joint model of the tomographic system and of the imaging device. Observed variables are shaded. Photons are indexed by $c = \{1, \dots, N_c\}$. The variables i_c are 1-of-N_d indicators representing the location of interaction of photon c, with $d = 1, \dots, N_d$ indicating the allowed discrete locations of interaction. Activity $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_v, \lambda_{N_v}]$ is the unique cause of the N_c photon interactions. The location of the interaction of photon c is the unique cause of the secondary photons v_c. v_c is a vector of length N_k equal to the number of secondary photo-detectors (photo-multiplier tubes or solid state detectors). $A = a_{vd}$ describes the geometry of the system (collimator, attenuation map) and L the system model of the gamma camera, relating the allowed locations of interaction to the expected count of secondary photons.

The SPECT acquisition produces N_c photon interactions. Each interaction generates a vector of electrical signals $v_c = [v_{c1}, v_{c2}, \dots, v_{ck}, \dots, v_{cN_k}]$ of length N_k, where k indexes the secondary photo-detectors. Let $v = \{v_1, v_2, \dots, v_c, \dots, v_{N_c}\}$ be the set of all the measurements; let $i = \{i_1, \dots, i_c, \dots, N_c\}$ be a set of N_c indicator functions, one for each photon interaction and N_d be the number of allowed discrete locations of interaction, indexed by $d = 1, \dots, N_d$. Each indicator i_c takes N_d possible values $i_c = \{0000 \dots 001, 0000 \dots 010, \dots\}$. Let $Sysmat = a_{vd}$ be the probability that a photon emitted in v interacts with the crystal in location d (notice that while in the previous formulation (2.45) this was *is detected*, here it's *interacts*; though subtle, this is the key difference). A encompasses again the characteristics of the collimator, position of the cameras and the attenuation map.

Let $L = l_{dk}$ be the probability that a secondary photon emitted from location d is detected by photo-detector k. Assuming that all the detected gamma-photons have the same energy and letting E be the product of the energy of the gamma-photon multiplied by the yield of the scintillator crystal and by the quantum efficiency of the secondary photo-detectors, the photon counting processes that describe the emission and interaction of the gamma photons and the detection of the secondary photons in the camera are the following:

$$p(i|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} \mathcal{P}(\sum_{c=1}^{N_c} i_c \cdot e_d; \sum_{v=1}^{N_v} a_{vd}\lambda_v) \tag{8.1}$$

$$p(v_c|i_c) = \prod_{k=1}^{N_k} \mathcal{P}(v_{ck}; [L \cdot i_c]_k E) \tag{8.2}$$

$$p(v|i) = \prod_{c=1}^{N_c} p(v_c|i_c) \tag{8.3}$$

where $\mathcal{P}$ is the Poisson distribution and e_d is a unit vector of length N_d with the element indexed by d equal to 1. Recognising that the photon counts in detector bin are given by:

$$q_d = \sum_{c=1}^{N_c} i_c \cdot e_d \tag{8.4}$$

equation (8.1) is the Poisson model described in chapter 2-paragraph 2.3. Equations (8.2) and (8.3) express the model of the imaging device described in chapter 7-paragraph 7.2. the joint probabilistic model of equations (8.1), (8.2) and (8.3) is represented graphically in Fig. 8.1. Let us indicate with $\bar{q_d}$ the expected photon counts in d given the activity:

$$\bar{q_d} = \sum_{v=1}^{N_v} a_{vd}\lambda_v \tag{8.5}$$

On the optimum estimation of the locations of interaction

The probabilistic model of the imaging device described in chapter 7-paragraph 7.2 and expressed here in equation (8.2), expresses the likelihood of the measurement obtained from the array of secondary photo-detectors (for the photon interaction c) v_c given the location of interaction of the photon. One may, according to such model, estimate the location of interaction i_c by maximum likelihood (ML). The maximum likelihood estimator (MLE) for the location of interaction may be shown to be *optimum* in the sense the $L2$-norm of the expected error. That is, if one measures v_c, the best estimate that one can obtain of the location of interaction i_c is given by the MLE estimator.

However, though this appears to be the maximum resolution of the locations of interaction, the joint model described in equations (8.1)-(8.3) and in Fig. 8.1, exposes that the location of interaction can be measured with higher accuracy. Consider the following example. A small source of radiation, small enough to be considered to be a point source, is placed in front of a planar monolithic gamma-camera

equipped with parallel-hole collimator. The source is located at a given distance and the characteristics of the camera and of the collimator are well known. One now collects a number of measurement vectors $v_1, v_2, \ldots, v_c, \ldots, v_{N_c}$. The location of interaction of each of the N_c photons is random, though concentrated around the point corresponding to the intersection of the line normal to the camera plane that passes through the point source. One may therefore guess the location of interaction of photon c without even looking at v_c. If one considers the experiment of measuring v_c alone, the fact that photon was certainly generated by the point source at the known location, and the knowledge of the characteristics of the imaging system, determine prior knowledge about the location of interaction. If the location of the point source is not known, one may at first estimate the location of the source, and this can be estimated with high accuracy from the multiple interactions, then re-estimate the location of interaction of each photon at the light of the acquired (i.e. computed) information. When the source is more complex than a point, the problem is unchanged. Referring to the DAG of the joint model of Fig. 8.1, if one disregards $\boldsymbol{\lambda}$, the interactions are independent $i_c \perp i_{c'}, c \neq c'$. However, when the model includes $\boldsymbol{\lambda}$, as in figure, the interactions are no longer independent, although they are independent when the value of $\boldsymbol{\lambda}$ is given. This is the same type of relation that holds between the activity and the photon counts, discussed in chapter 2 paragraph 2.3.2. Since the locations of interaction are no longer independent when one considers $\boldsymbol{\lambda}$, the measurement v_c produces information also regarding the location of interaction of another photon. Considering the tomographic, which includes multiple cameras, the photon interactions measured by one of the cameras, provides information about the location of interaction of the photons in other cameras.

Joint estimation of the activity and of the events of interaction

Let us consider the problem of optimising jointly the model of equations (8.1)-(8.3) and Fig. 8.1 with respect to $(\boldsymbol{\lambda}, i)$.

$$\hat{\boldsymbol{\lambda}}, \hat{i} = \arg\max_{\boldsymbol{\lambda}, i} p(\boldsymbol{\lambda}, i|v) \tag{8.6}$$

One approach to the joint optimisation is alternated optimisation (Iterated Conditional Modes), with the following subsets:

- $\mathcal{S}_1$: $\boldsymbol{\lambda}$
- $\mathcal{S}_2$: i

Therefore ICM consists in alternating the optimisation with respect to $\boldsymbol{\lambda}$ given provisional estimates of the locations of interaction with the estimation of the locations of interaction given the provisional estimate of the activity:

$$\hat{i}^{(n+1)} = \operatorname*{argmax}_i p(i|v, \boldsymbol{\lambda}^{(n)}) = \operatorname*{argmax}_i p(i|\boldsymbol{\lambda}^{(n)}) p(v|i) \tag{8.7}$$

$$\hat{\boldsymbol{\lambda}}^{(n+1)} = \operatorname*{argmax}_{\boldsymbol{\lambda}} p(\boldsymbol{\lambda}|v, i^{(n+1)}) \tag{8.8}$$

The locations of interaction are independent conditionally to the activity: $i_c \perp i_{c'}|\boldsymbol{\lambda}, c \neq c'$. Therefore (8.7) splits into N_c optimisations. Taking the *log* of (8.7), and considering that it factorises in N_c terms, the optimisation regarding photon c consists in maximising the sum of two terms. One term is the likelihood of the measurement, expressed by (8.2), and the other term is the prior probability that the interaction happened in $l_c == e_d$, given by the activity and is expressed by the marginalisation of (8.1) with respect to $i_{c'}, c' \neq c$. This marginalisation is non trivial, however, it is simple to understand intuitively, considering a simple example. Given $\boldsymbol{\lambda}$, the number of expected photon counts in d is $\bar{q_d} = \sum_{v=1}^{N_v} a_{vd}\lambda_v$. Then, if one considers a single photon interaction what is the probability that this happened in d without looking at the measurement v_c? It is proportional to the rate of events in d, therefore:

$$p(i_c = e_d|\boldsymbol{\lambda}) = \frac{\sum_v^{N_v} a_{vd}\lambda_v}{\sum_{d=1}^{N_d}\sum_v^{N_v} a_{vd}\lambda_v} = \frac{\bar{q_d}}{\sum_{d=1}^{N_d}\bar{q_d}} \tag{8.9}$$

Substituting in (8.7):

$$\hat{i_c}^{(n+1)} = \underset{i_c}{\operatorname{argmax}} \log \frac{\sum_v^{N_v} a_{vd}\lambda_v}{\sum_{d=1}^{N_d}\sum_v^{N_v} a_{vd}\lambda_v} + \sum_k - [[L \cdot i_c]_k + v_{ck} \log[L \cdot i_c]_k] \tag{8.10}$$

This can be maximised with the direct search algorithm described in chapter 7 paragraph 7.3 (Fig. 7.3). Notice that ML estimation of the location of interaction, described in 7 paragraph 7.3 consists in maximising only the second term of (8.10). The computation of the first term consists in a single projection.

The optimisation of the activity given the provisional estimates of the locations of interaction (equation (8.8)), considering that $p(\boldsymbol{\lambda}|i^{(n+1)}, v) = p(\boldsymbol{\lambda}|i^{(n+1)}) \propto p(i^{(n+1)}|\boldsymbol{\lambda})$, can be performed with the EM algorithm:

$$\lambda_v^{(n+1)} = \lambda_v^{(n)} \frac{1}{\sum_{d=1}^{N_d} a_{vd}} \sum_{d=1}^{N_d} a_{vd} \frac{\sum_{c=1}^{N_c} i_c^{(n+1)} e_d}{\sum_{v'=1}^{N_v} a_{v'd}\lambda_v^{(n)}} \tag{8.11}$$

8.2 Compensation of disrupted detectors

It is not uncommon that one or more photo-detectors of a γ-camera stop working due to hardware failure. In this condition, the resolution of the camera in the region near the disrupted detector is lower, as visualised in Fig. 8.2-left. We adopt the joint model for the estimation of the activity and the events of interaction in order to compensate for the disrupted detector. A SPECT perfusion study is simulated with the SIMIND Monte Carlo simulator with 50×10^6 counts and ideal, infinite resolution, gamma camera (8.2-left). Events of interaction are filtered based on simple energy thresholding, in order to reduce the effect of scattered photons. The γ interactions are simulated according to the solid angle model described

in chapter 7 paragraph 7.6.1 for an array of 25 square photo-detectors of size $4\times4cm$, quantum efficiency $\nu = 0.8$ and $1cm$ thick NaI(Tl) crystal with yield of 53200 photons. The same simulation is performed disabling two of the photo-detector.

Joint reconstruction of the activity and the events of interaction with 100 iterations of the ICM algorithm show, at visual inspection, recovery of the lost resolution (Fig. 8.2-right). This is explained by the flow of information amongst the cameras through the iterative process. Since the uncertainty at the centre of the disrupted camera is large, the location of interaction of the photons is decided, as a consequence of the joint model, in accordance to the observations of the other cameras.

8.3 Unified computational model of scatter

A common use of the information produced by the gamma cameras is to estimate the location of interaction (regardless of the underlying activity), of each photon and to disregard the energy. A second common use, especially in SPECT, is to select the photons that, according to the measured energy, are likely to have scattered, and to disregard the rest (energy thresholding). Energy thresholding, or energy windowing is a scatter rejection algorithm, or more appropriately, a reconstruction algorithm that accounts for scattering and for the energy of the detected gamma photons, perhaps the simplest algorithm of such sort. Let us extend the model described in the previous paragraph, to account for scattered photons. Let us consider the discrete variable $s_c = \{0, 1\}$ that indicates whether a photon has scattered on its way to the detector, or not. One could consider other variables, such as an integer variable that indicates the number of scattering events, but let us consider here the binary case. The DAG of the measurement system is represented in Fig. 8.3. There is no energy windowing, a photon c detected by the gamma camera has energy E_c. Let $S = \{\sigma_{vd}\}$ be the probability that a photon emitted in v scatters and is detected in d and let $A = \{a_{vd}\}$ be the probability that a photon emitted in v is detected in d un-scattered. (The probability that a photon emitted in v is detected in d is given by the sum $\sigma_{vd} + a_{vd}$).

$$p(i, s|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} \mathcal{P}(\sum_{c=1}^{N_c} s_c i_c \cdot e_d; \sum_{v=1}^{N_v} \sigma_{vd}\lambda_v)\mathcal{P}(\sum_{c=1}^{N_c} (!s_c) i_c \cdot e_d; \sum_{v=1}^{N_v} a_{vd}\lambda_v) \tag{8.12}$$

Let:

$$q_d^P \triangleq \sum_{c=1}^{N_c} s_c i_c \cdot e_d \tag{8.13}$$

$$q_d^S \triangleq \sum_{c=1}^{N_c} (!s_c) i_c \cdot e_d \tag{8.14}$$

As described in chapter 7 paragraph 7.2, the probability distribution associated to the measurement v_c when the energy and the location of interaction is known, is expressed by:

$$p(v_c|i_c, E_c) = \prod_{k=1}^{N_k} \mathcal{P}(v_{ck}; [L \cdot i_c]_k E_c) \tag{8.15}$$

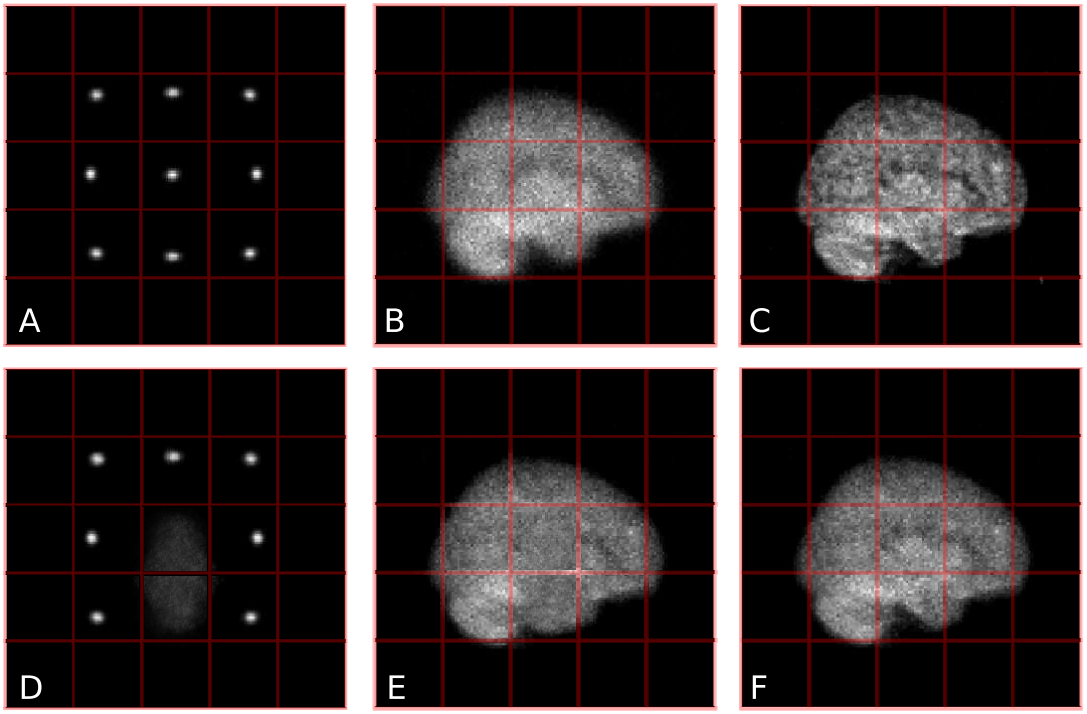


Figure 8.2: The red boxes represent the secondary photo-detectors. A) Visualisation of the uncertainty in different locations of the detector obtained by reconstructing 10000 events for each of 9 locations and histogramming the reconstructed locations of interaction on the detector plane. B) Histogram of the ML reconstructions of the locations of interaction for one position of the gamma camera. C) Histogram of the ideal locations of interaction for one position of the gamma camera, obtained under the assumption of perfect reconstruction of the locations of interaction (no uncertainty associated to the measurement of location of interaction of the gamma photons). D) Visualisation of the uncertainty in different locations of the detector obtained again by reconstructing 10000 events for each of 9 locations and histogramming the reconstructed locations of interaction on the detector plane. Two of the photodetectors are disrupted. Uncertainty is much greater for photons that interact in regions near the disrupted photodetector. E) Histogram of the ML reconstructions of the locations of interaction for one position of the gamma camera with two disrupted photodetectors. Notice the the effect of the disrupted photodetectors in comparison with B. F) Histogram of the reconstructions of the locations of interaction obtained, in the case of the two disrupted photodetectors, by iterative joint reconstruction of the activity and of the locations of interaction. Notice the recovery of resolution in the region of the gamma camera corresponding to the disrupted photodetectors.

(notice that this is a slight modification of (8.3)). The conditional probability distribution of the measurement given the location and scatter labels is expressed by the following integral (see the graph in Fig. 8.3):

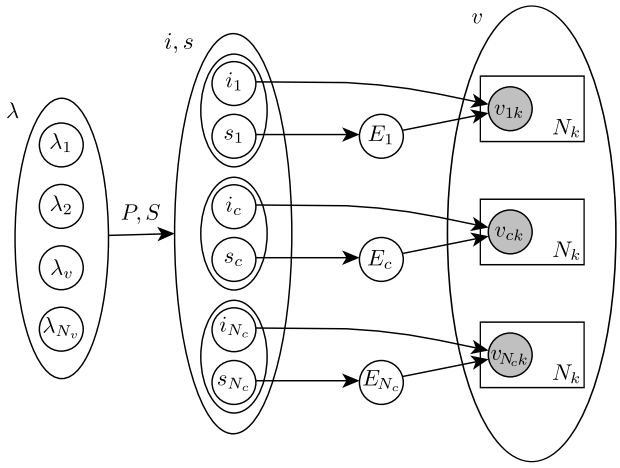


Figure 8.3: DAG of the unified scatter model. Observed variables are shaded. Photons are indexed by $c = \{1, \ldots, N_c\}$. i_c are 1-of-N_d indicator functions representing the location of interaction of photon c, with $d = 1, \ldots, N_d$ indicating the allowed discrete locations of interaction. $s_c = \{0, 1\}$ is a binary variable indicating whether photon c has scattered or not. Activity $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_v, \lambda_{N_v}]$ is the unique cause of i_c, s_c. The location of the interaction and the scatter label of photon c are the unique cause of the secondary photons v_c. v_c is a vector of length N_k equal to the number of secondary photo-detectors (photo-multiplier tubes or solid state detectors). The energy of the photon E_c depends on whether is has scattered or not. If one knew the exact location of interaction and the energy of photon c, the scatter label s_c would not add information about v_c. $A = a_{vd}$ and $S = \sigma_{vd}$describe the geometry of the system (collimator, scatter coefficients).

$$p(v_c|i_c, s_c) = \int p(v_c|i_c, E_c)p(E_c|s_c)\mathrm{d}E_c \tag{8.16}$$

Joint estimation of the activity and of the scattered photons Let us consider the problem of optimising jointly the model of equations (8.1)-(8.3) and Fig. 8.1 with respect to $(\boldsymbol{\lambda}, i, s)$.

$$\hat{\boldsymbol{\lambda}}, \hat{i}, \hat{s} = \arg\max_{\boldsymbol{\lambda}, i, s} p(\boldsymbol{\lambda}, i, s|v) \tag{8.17}$$

Let us alternate the optimisation of the following two subsets:

- $\mathcal{S}_1$: $\boldsymbol{\lambda}$
- $\mathcal{S}_2$: i, s

The optimisation of the activity, given the provisional estimates of i and s, consists in maximising $p(\boldsymbol{\lambda}|i, s) \propto p(i, s|\boldsymbol{\lambda})$, expressed by (8.12). $p(i, s|\boldsymbol{\lambda})$ can be written as $p(q_d^P, q_d^S|\boldsymbol{\lambda})$, with q_d^P and q_d^P defined in (8.13) and (8.14). The Let us differentiate the *log* of (8.12):

$$\frac{\partial}{\partial \boldsymbol{\lambda}_v} \log p(q_d^P, q_d^S|\boldsymbol{\lambda}) = -\sum_{d=1}^{N_d} (a_{vd} + \sigma_{vd}) + \sum_{d=1}^{N_d} \left(\frac{q_d^P}{\sum_{v'=1}^{N_v} a_{v'd}\lambda_{v'}} + \frac{q_d^S}{\sum_{v'=1}^{N_v} \sigma_{v'd}\lambda_{v'}} \right) \tag{8.18}$$

Therefore, rearranging the gradient, as described in chapter 3 paragraph 3.1.1, the MLEM algorithm for the optimisation of the activity is the following:

$$\lambda_v^{(n+1)} = \lambda_v^{(n)} \frac{1}{\sum_{d=1}^{N_d} (a_{vd} + \sigma_{vd})} \sum_{d=1}^{N_d} \left(a_{vd} \frac{q_d^P}{\sum_{v'=1}^{N_v} a_{v'd}\lambda_{v'}} + \sigma_{vd} \frac{q_d^S}{\sum_{v'=1}^{N_v} \sigma_{v'd}\lambda_{v'}} \right) \tag{8.19}$$

The optimisation of i, s given the provisional estimate of the activity factors in N_c optimisations. Let us indicate with Φ the cost function for the optimisation relative to photon c:

$$\hat{i_c}, \hat{s_c} = \underset{i_c, s_c}{\operatorname{argmax}}\, \Phi(i_c, s_c) \tag{8.20}$$

$$\Phi(i_c, s_c) = \log p(i, s|\boldsymbol{\lambda}) + \log \int p(E_c|s_c) p(v_c|E_c, i_c) \mathrm{d}E_c \tag{8.21}$$

Discretising the gamma camera in, e.g. $256 \times 256 \times 10$ pixels, and considering that i_c is binary, the the dimensionality of the search space (i_c, s_c) is not large. We could therefore proceed by direct search. However, the evaluation of the cost function requires the evaluation of the integral over the energy reported on the left had side of (8.21). This may appear problematic for the fast estimation of the location of interaction and scatter labels. However, considering, as explained in the following, that the maximiser of the probability distribution of the location of interaction conditional to the scatter label is independent of the energy, the integrals can be pre-computed and the optimisation can be performed with a simple direct search algorithm of the type described in 7 paragraph 7.3 (Fig. 7.3). In order to select the most probable value of i_c and s_c, we wish to compute the cost function $\log p(i, s|\boldsymbol{\lambda}) + \log p(v_c|i_c, s_c)$ for every value of i_c and s_c. Computing the cost function for all the values of i_c and s_c is trivially equivalent to computing the cost function for all the locations i_c for $s_c = 1$ and then for all the values of i_c for $s_c = 0$.

$$\Phi^{\text{no-scatter}}(i_c) = \Phi(i_c, s_c = 0) = \log p(i, s|\boldsymbol{\lambda}) + \log \int p(E_c|s_c = 0) p(v_c|E_c, i_c) \mathrm{d}E_c \tag{8.22}$$

$$\Phi^{\text{scatter}}(i_c) = \Phi(i_c, s_c = 1) = \log p(i, s|\boldsymbol{\lambda}) + \log \int p(E_c|s_c = 1) p(v_c|i_c, E_c) \mathrm{d}E_c \tag{8.23}$$

As one observes in equation (8.10), the maximiser of $p(v_c|E_c, i_c)$ with respect to i_c does not depend on E_c. Therefore, moving this term out of the integral, the maximisers of (8.22) and (8.23) with respect to the location of interaction do not depend on E_c:

$$\hat{i_c}^{\text{no-scatter}} = \underset{i_c}{\operatorname{argmax}}\, \Phi^{\text{no-scatter}}(i_c) = \underset{i_c}{\operatorname{argmax}} \log \frac{q_d^P}{\sum_{d=1}^{N_d} q_d^P + q_d^S} + \sum_k - [[L \cdot i_c]_k + v_{ck} \log[L \cdot i_c]_k] \tag{8.24}$$

$$\hat{i_c}^{\text{scatter}} = \underset{i_c}{\operatorname{argmax}}\, \Phi^{\text{scatter}}(i_c) = \underset{i_c}{\operatorname{argmax}} \log \frac{q_d^S}{\sum_{d=1}^{N_d} q_d^P + q_d^S} + \sum_k - [[L \cdot i_c]_k + v_{ck} \log[L \cdot i_c]_k] \tag{8.25}$$

The value of i_c that maximises jointly the cost function with respect of i_c, s_c is one of these two. (8.24) and (8.25) can be maximised by applying twice the direct search algorithm described in chapter 7 paragraph 7.3 (Fig. 7.3). The joint maximiser of the cost function is either $(\hat{i_c}^{\text{no-scatter}}, s_c = 0)$, or $(\hat{i_c}^{\text{scatter}}, s_c = 1)$. depending on which of the two gives a higher value of the cost function 8.21. The evaluation of the cost function, however, involves the integration over the energy:

$$\log \int p(E_c|s_c = 1) p(v_c|i_c, E_c) \mathrm{d}E_c \tag{8.26}$$

With $p(v_c|i_c, E_c)$ expressed by (8.15):

$$p(v_c|i_c, E_c) = \prod_{k=1}^{N_k} \mathcal{P}(v_{ck}; [L \cdot i_c]_k E_c) \tag{8.27}$$

This, in the integration in (8.22) and (8.23), is a function of E_c with given parameters v_c and i_c The integral (8.26) appears complicated due to the term of the integrand (8.27) being a product of functions (Poisson distributions). However, applying the sum property of the Poisson function, (8.27) can be expressed as a function of E_c parametrised by the sum of the elements of vector v_c, rather than a function of E_c parametrised by v_c. Since, as expressed by (8.27), the conditional probability distributions associated to v_{ck} are independent Poisson distributions, by the sum rule, the conditional probability distribution associated to $\sum_{k=1}^{N_k} v_{ck}$ is a Poisson distribution, whose expectation is the sum of the expectations of v_{ck}:

$$V_c \triangleq \sum_{k=1}^{N_k} v_{ck} \tag{8.28}$$

$$p(V_c|i_c, E_c) = \mathcal{P}(V_c; \sum_{k=1}^{N_k} [L \cdot i_c]_k E_c) \tag{8.29}$$

This is equivalent to (8.27), but expresses explicitly the dependence on the sum of the elements of v_{ck}, rather than on the values of v_{ck} considered one by one. Then (8.26) can be written as:

$$\log \int p(E_c|s_c = 1)\mathcal{P}(V_c; \sum_{k=1}^{N_k} [L \cdot i_c]_k E_c) \mathrm{d}E_c \triangleq f(s_c, i_c, V_c) \tag{8.30}$$

$f(s_c, i_c, V_c)$ can be pre-computed and stored in a look-up table as V_c is an integer between 0 and a few thousand (maximum yield of the gamma-camera), s_c is binary and the number of discrete locations of interaction i_c is of the order of a few thousand (e.g. $256 \times 256 \times 10$). Each value in the table is obtained by performing the integration (8.30). Substituting (8.30) in

$$\Phi(i_c, s_c = 0) = \log \frac{q_d^P}{\sum_{d=1}^{N_d} q_d^P + q_d^S} + f(0, \hat{i_c}^{\text{no-scatter}}, V_c) \tag{8.31}$$

$$\Phi(i_c, s_c = 1) = \log \frac{q_d^S}{\sum_{d=1}^{N_d} q_d^P + q_d^S} + f(1, \hat{i_c}^{\text{scatter}}, V_c) \tag{8.32}$$

In summary, the estimation of the location of interaction and scatter labels consists in 1 pre-computation step and 5 run-time steps:

- Pre-computation
  1. Evaluate $f(s_c, i_c, V_c)$ (the integral of eq. (8.30)) for all values of i_c, s_c, V_c and store in a loo-kup table.
- Run-time:
  1. Compute the most probable location of interaction in case the photon has not scattered $\hat{i_c}^{\text{no-scatter}}$ (eq. (8.24));
  2. Compute the most probable location of interaction in case the photon has scattered $\hat{i_c}^{\text{scatter}}$ (eq. (8.25));
  3. Compute the integral of the measurement V_c (eq. (8.28));
  4. Extract from the look-up table $f(0, \hat{i_c}^{\text{no-scatter}}, V_c)$ and $f(1, \hat{i_c}^{\text{scatter}}, V_c)$;
  5. Compute the cost function (eq. (8.31)-(8.32)) and decide between $(\hat{s_c} = 0, \hat{i_c} = \hat{i_c}^{\text{no-scatter}})$ and $(\hat{s_c} = 1, \hat{i_c} = \hat{i_c}^{\text{scatter}})$.

8.4 Continuous-energy scatter-rejection algorithm

The algorithm described in the previous paragraph iteratively updates the activity, making use of the un-scattered and of the scattered photons. The decision of whether a photon has scattered or not is re-instantiated at every iteration and it is based on the detected energy and on the expected number of scattered photons at the location of interaction, relative to the total expected number of photons (scattered and un-scattered). Both steps of the algorithm (update of the activity and update of the location of

interaction and scatter labels) require the computation of the un-scattered projection $A\boldsymbol{\lambda}$ and of the scattered projection $S\boldsymbol{\lambda}$. Additionally, update of the activity requires the computation of the un-scattered back-projection $A^T(\cdot)$ and of the scattered back-projection $S^T(\cdot)$.

Computing the scatter projection and back-projection at every iteration is, in general, computationally demanding. This paragraph investigates the simplest forward model of scatter: the scatter operator S is assumed to be a *matrix of ones* scaled by a known scaling factor η:

$$S = \eta \cdot \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & 1 & \dots & 1 \\ \dots & \dots & \dots & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} \tag{8.33}$$

With such imaging matrix, the projection is identical for every value of d and equal to the integral sum of the activity scaled by η. The use of scatter matrix (8.33) in the model described in the previous paragraph has two implications:

1. Scattered photons are nearly not informative of the activity. The expression of the conditional probability distribution of i, s of equation 8.12 simplifies to:

$$p(i,s|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} \mathcal{P}(\sum_{c=1}^{N_c} s_c i_c \cdot e_d; \eta \sum_{v=1}^{N_v} \lambda_v) \mathcal{P}(\sum_{c=1}^{N_c} (!s_c) i_c \cdot e_d; \sum_{v=1}^{N_v} a_{vd}\lambda_v) \tag{8.34}$$

   The observation of the scattered photons ($i_c = 1$) defines a probability distribution over the sum of the activity. The scattered photons are therefore informative only of the integral sum of the activity. Differentiating the expression (8.34), the MLEM algorithm of equation (8.19), in the case of scatter matrix (8.33) is:

$$\lambda_v^{(n+1)} = \lambda_v^{(n)} \frac{1}{\sum_{d=1}^{N_d} (a_{vd} + \eta)} \sum_{d=1}^{N_d} \left(a_{vd} \frac{q_d^P}{\sum_{v'=1}^{N_v} a_{v'd}\lambda_{v'}} + \frac{q_d^S}{\sum_{v'=1}^{N_v} \lambda_{v'}} \right) \tag{8.35}$$

2. The expected number of scattered photons given the activity is identical for all the detector locations:

$$\bar{q}_d^S = \nu \sum_{v=1}^{N_v} \lambda_v \tag{8.36}$$

   Notice that the decision of whether a photon has scattered or not still depends on the current estimate of the activity. In fact the decision of whether the photon has scattered or not depends on q_d^S and on q_d^P (see eq. (8.31)-(8.32)). Adopting the matrix-of-ones approximation, q_d^S is constant, but q_d^P is not.

In order to evaluate the algorithm, a synthetic brain perfusion phantom simulating the uptake of Technetium-99m was projected with the Simind Monte Carlo simulator, with the parameters of a standard SPECT scanner with LEHR parallel-hole collimator. The data was collected in list mode, recording the location of the annihilation and the energy of each event of interaction for 50×10^6 events. The Simind simulator also records the number of scatter events for each photon. The interaction of the gamma photons with the gamma camera was simulated according to the solid angle model described in chapter 7 paragraph 7.6.1, for an array of 25 square photo-detectors of size $40 \times 40mm$, quantum efficiency $\nu = 0.8$ and $10mm$ thick NaI(Tl) crystal with yield of $38photons/KeV$ (corresponding to 5320 average secondary photons for the un-scattered gamma photons emitted by Technetium-99m at $140KeV$). Each interaction was simulated by computing the 25 expected detector read-out signals and adding Poisson noise.

In order to achieve convergence, the MLEM algorithm was replaced by the OSL-MAPEM algorithm with smoothness prior. The level of smoothness was set empirically to achieve convergence within 100 iterations. The joint reconstruction was then obtained by alternating for 100 iterations the update of the activity and the update of the labels. The parameter η, expressing the ratio between the scattered photons and the un-scattered photons, was assumed to be known:

$$\eta = \frac{\sum_d \sum_v \tilde{\sigma} \boldsymbol{\lambda}_v}{\sum_d \sum_v \tilde{a} \tilde{\boldsymbol{\lambda}}_v} \tag{8.37}$$

Where $\tilde{a}$ and $\tilde{\sigma}$ indicate the exact system matrices utilised in the Monte Carlo simulation and $\tilde{\boldsymbol{\lambda}}$ the true activity. Fig. 8.4 reports the binned photon counts obtained for the camera at location 0^o (i.e. the viewgram at angle 0^o). Image (A) displays the photon counts binned regardless of their energy (scattered and un-scattered). Images B and C display the photon counts obtained by counting the photons with identical location labels i_c for $s_c = 0$ (B) and for $s_c = 1$ (C) at convergence. Fig. 8.5 displays the histogram of the integral number of secondary photons.

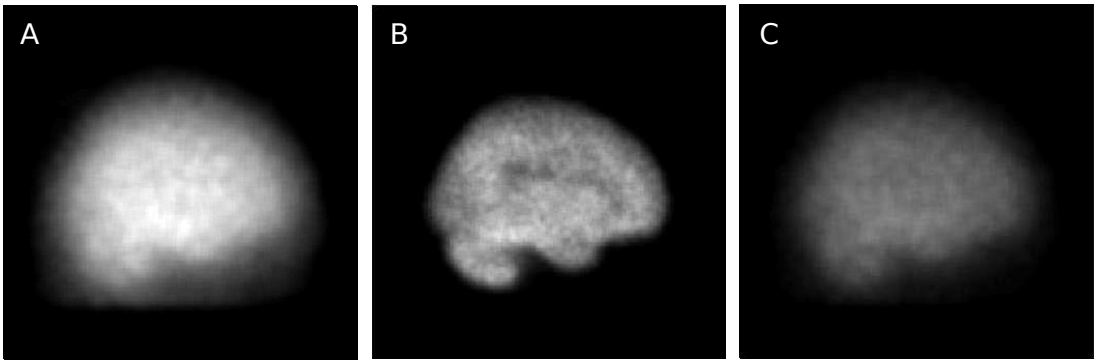


Figure 8.4: Synthetic brain perfusion SPECT imaging. A) viewgram considering all photons (energy is discarded); B) reconstructed viewgram of un-scattered photons; C) reconstructed viewgram of scattered photons.

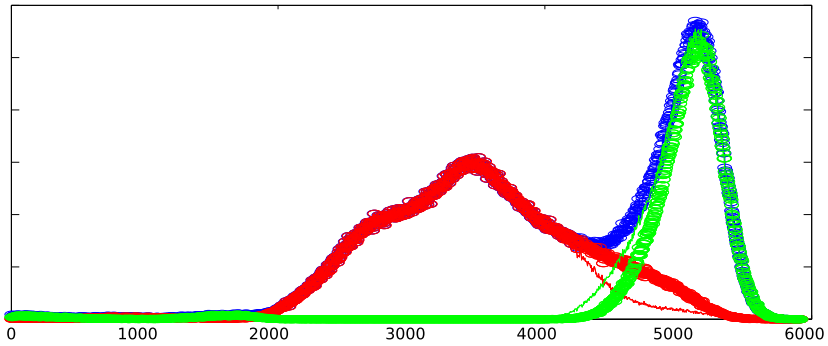


Figure 8.5: Histogram of the integral sum of the secondary photons. The histogram obtained only with the scattered photons is coloured in red; the histogram obtained with the un-scattered photons is coloured in green. Circles indicate the histogram obtained with the ground truth labels (from the Simind Monte Carlo simulator). Solid lines indicate the histograms obtained according to the estimated labels.

Chapter 9

Estimation of the information retrieval

Probabilistic models for emission tomography have proven more effective than the early deterministic models as they allow us to account for the intrinsic uncertainty of the emission measurements and to employ accurate models of the imaging systems. Furthermore sources of information other than the photon counts may be embedded in the reconstruction process. Another important advantage of the probabilistic formulation is that by describing the measurements as random variables, one may establish the relationship between the measurements and the uncertainty in the reconstructed images. The work described in this chapter addresses the estimation of the uncertainty of the reconstruction and the optimisation of the design of the imaging systems under the information theoretic pespective.

ML and MAP are point estimators of the posterior probability distribution that one associates to the unknown variables (the concentration of radio-pharmaceutical in each voxel). However, in the fully Bayesian apporach, one is interested in the full *posterior probability distribution* of the unknown variables, rather than the point that maximises it. The posterior probability distribution of the unknown $\boldsymbol{\lambda}$ is readily expressed analytically by equation (3.19) (and equations (2.44), (2.45) for the Poisson model). Such expression of the posterior represents the joint probability distribution over all the components of vector $\boldsymbol{\lambda}$ when photon counts $\boldsymbol{q}$ are observed; this expression can be marginalised with respect to all the components of vector-valued $\boldsymbol{\lambda}$ except for the v-th, obtaining the probability distribution $p(\lambda_v|\boldsymbol{q})$ associated to the uptake of pharmaceutical in v. A *synthetic* representation of the uncertainty about the uptake in v, such as the second moment (variance) of $p(\lambda_v|\boldsymbol{q})$ could be used to inform decisions related to the uptake of radio pharmaceutical in voxel v, either visually or in an automated decision process. More in general, the posterior probability distribution $p(\boldsymbol{\lambda}|\boldsymbol{q})$ can be integrated in order to obtain the probability distribution not only of the uptake in a voxel, but of derived measurements, such as the integral uptake in a region of interest. Such measure of uncertainty may be employed to inform design choices for new imaging systems. One family of algorithms for the integration of the posterior that has become popular in other fields in the last few years is based on *posterior sampling* techniques (Markov Chain Monte Carlo - MCMC). However neither sampling techniques nor deterministic integration schemes are viable due to the high dimensionality of $p(\boldsymbol{\lambda}|\boldsymbol{q})$. Instead, this chapter describes the use of the Laplace approxi-

mation for the estimation of the posterior probability distribution of the activity and of its marginals and an algorithm based on the sparsification of the Hessian of the log-posterior for the efficient computation of information measures.

9.1 Information Retrieval

Let us consider an imaging system whose response is a function of a set of parameters ξ.

$$p_\xi(\boldsymbol{q}|\boldsymbol{\lambda}) \tag{9.1}$$

In the case of the Poisson model 2.3, the probabilistic interaction between $\boldsymbol{\lambda}$ and $\boldsymbol{q}$ is regulated entirely by the system matrix. (Notice that this is not the case e.g. with the homoscedastic Gaussian model 3.3, where $p_\xi(\boldsymbol{q}|\boldsymbol{\lambda})$ is parameterised by the imaging matrix A and by the parameter σ). Let us denote by A_ξ the system matrix, dependent on the parameters of the acquisition system, then we can write:

$$p_\xi(\boldsymbol{q}|\boldsymbol{\lambda}) = \prod_{d=1}^{N_d} \mathcal{P}\left(q_d; [A_\xi \boldsymbol{\lambda}]_d\right) \tag{9.2}$$

Upon observing $\boldsymbol{q}$ with parameters ξ, the probability distribution associated to $\boldsymbol{\lambda}$ is:

$$p_\xi(\boldsymbol{\lambda}|\boldsymbol{q}) = \frac{p_\xi(\boldsymbol{q}|\boldsymbol{\lambda})p(\boldsymbol{\lambda})}{p_\xi(\boldsymbol{q})} \tag{9.3}$$

with

$$p_\xi(\boldsymbol{q}) = \int p_\xi(\boldsymbol{q}|\boldsymbol{\lambda})p(\boldsymbol{\lambda})\mathrm{d}\boldsymbol{\lambda} \tag{9.4}$$

One can then express information measures such as (see [117] and [91]):

$$E_{\boldsymbol{\lambda}|\boldsymbol{q}}(\xi) = \int p_\xi(\boldsymbol{\lambda}|\boldsymbol{q}) \log p_\xi(\boldsymbol{\lambda}|\boldsymbol{q})\mathrm{d}\boldsymbol{\lambda} \tag{9.5}$$

and

$$E_{\lambda_v|\boldsymbol{q}}(\xi) = \int p_\xi(\lambda_v|\boldsymbol{q}) \log p_\xi(\lambda_v|\boldsymbol{q})\mathrm{d}\lambda_v \tag{9.6}$$

with

$$p(\lambda_v|\boldsymbol{q}) = \int p(\boldsymbol{\lambda}|\boldsymbol{q})\mathrm{d}\lambda_{v'\neq v} \tag{9.7}$$

Equation (9.5) expresses the information that one would have about $\boldsymbol{\lambda}$ upon observing $\boldsymbol{q}$ and (9.6) the information that one would have about the unknown variable in voxel v. Measures of information related to functions of $\boldsymbol{\lambda}$, such as the integral uptake in a region of interest, can eventually be expressed via the distribution function method. Let us denote with $\varkappa$ the function of $\boldsymbol{\lambda}$ that we are interested in:

$$E_{\varkappa|\boldsymbol{q}}(\xi) = f(p_\xi(\boldsymbol{\lambda}|\boldsymbol{q})) \tag{9.8}$$

$E_{\varkappa|\boldsymbol{q}}$ encompasses (9.5) and (9.6) for $\varkappa = \boldsymbol{\lambda}$ and $\varkappa = \lambda_v$.

If the variable of interest is $\varkappa$, the system should be designed in order to retrieve maximum information about $\varkappa$. Equation (9.8) expresses the information that one would have about $\varkappa$ upon observing $\boldsymbol{q}$. Integrating over the probability distribution associated to $\boldsymbol{q}$ (9.4):

$$E_{\varkappa}(\xi) = \sum_{\boldsymbol{q}} p_{\xi}(\boldsymbol{q}) E_{\varkappa|\boldsymbol{q}}(\xi) \tag{9.9}$$

The global measure of information about $\boldsymbol{\lambda}$ of (9.6) (i.e. for $\varkappa = \boldsymbol{\lambda}$), is expressed by:

$$\begin{aligned} E_{\boldsymbol{\lambda}}(\xi) &= \sum_{\boldsymbol{q}} p_{\xi}(\boldsymbol{q}) \int p_{\xi}(\boldsymbol{\lambda}|\boldsymbol{q}) \log p_{\xi}(\boldsymbol{\lambda}|\boldsymbol{q}) \mathrm{d}\boldsymbol{\lambda} = & (9.10) \\ &= \sum_{\boldsymbol{q}} p_{\xi}(\boldsymbol{q}|\boldsymbol{\lambda}) p(\boldsymbol{\lambda}) \log p_{\xi}(\boldsymbol{\lambda}|\boldsymbol{q}) \mathrm{d}\boldsymbol{\lambda} & (9.11) \end{aligned}$$

9.1.1 Bayesian versus frequentist design optimisation

In the frequentist framework, as discussed in 3, an optimum design is one that produces, over many observations, the least error in the representation that one constructs about the true values (low bias), while minimising the difference across observations (low variance) and making the observations statistically independent (diagonal covariance matrix). The frequentist framework emphasises the existence of a true underlying image. In order to select an optimum design, one would start with a known value of $\boldsymbol{\lambda}$ (a phantom), and search, through the numerical simulation and reconstruction of multiple instances of the measurement for each candidate design parametrisation, for the design parameters that produce the best combination of bias and covariance. The difficulty, in the frequentist framework, is to establish a principled design cost function that expresses how favourable is a certain combination of bias and covariance.

In the Bayesian framework described in the previous paragraph, there is no concept of *true image*. A given measurement of the photon counts provides information about the underlying activity, the posterior distribution of the activity expressing the state of knowledge that one has about the activity after the observation. A set of design parameters is considered optimal if, considering all possible values of the underlying activity and all possible measurements that the machine may consequently produce, the information that the system retrieves about the variable of interest is maximum. The variable of interest is not necessarily the entire vector valued variable $\boldsymbol{\lambda}$, but perhaps the activity in one voxel, or a function of $\boldsymbol{\lambda}$.

It is somewhat simpler to define optimality criteria in the Bayesian framework. If one is interested in the integral uptake in the grey matter region, then the measurement should maximise the information associated to such scalar variable, which is a linear function of $\boldsymbol{\lambda}$. However, direct implementation of the Bayesian formulation of the previous paragraph is unfeasible due to computational complexity. It is clearly unthinkable to integrate over all possible values of $\boldsymbol{\lambda}$ and all possible values of q.

While the frequentist approach focuses on the true value of $\boldsymbol{\lambda}$, suggesting, in the basic formulation, to optimise the imaging system for the observation of one given object, the Bayesian approach, in the

basic formulation, suggests to integrate over all possible values of $\boldsymbol{\lambda}$. The integration over $\boldsymbol{\lambda}$ and over $\boldsymbol{q}$ that occurs in the Bayesian approach constitutes a tremendous challenge, but there is a second, more subtle difficulty. As expressed by equations (9.3)-(9.11), the optimum design depends (through equation (9.3)) on the prior belief about $\boldsymbol{\lambda}$. The probability distribution $p(\boldsymbol{\lambda})$ expresses the state of knowledge that one has about $\boldsymbol{\lambda}$ a priori. However the choice of a mathematical model, i.e. a functional form of $p(\boldsymbol{\lambda})$, is affected by considerations of computational complexity and of complexity of the mathematical formulation. The smoothing prior, though convenient for the simple formulation and for the simplicity of the algorithms that it yields, does not express the prior belief that one truly has about $\boldsymbol{\lambda}$. Therefore, the optimum design expressed by the formulae of the previous paragraph is the optimum design that one would have to use if $p(\boldsymbol{\lambda})$ were her prior belief about $\boldsymbol{\lambda}$. In brain imaging, one expects the activity to resemble the shape of a brain. However such qualitative prior knowledge is difficult to encode. In the frequentist framework, such knowledge appears to be expressed, to some extent, by the choice of the phantom; though optimising the system for the observation of a single object is a limitation. In the Bayesian framework described in the previous paragraph, there does not seem to be a way to express such knowledge. If one assigns $p(\boldsymbol{\lambda}) = \delta(\boldsymbol{\lambda} - \check{\boldsymbol{\lambda}})$, the posterior probability distribution (9.3) becomes a delta function and information is infinite for any choice of the design parameters.

The next paragraph addresses this difficulty, describing an algorithm that learns the optimum design parameters from random observations. Incidentally, the formulation of the next paragraph addresses the computational complexity of the Bayesian design optimisation problem.

9.2 Information Gain

Bob constructs the machine pictured in Fig. 9.1 and gives it to Alice as a gift. The machine is controlled by two buttons (N and M) and a knob and it displays information through two numerical displays A and $\boldsymbol{q}$. When you press button N, Bob tells Alice, a new value of vector-valued variable $\boldsymbol{\lambda}$ is generated according to a secret set of rules. However, the machine does not display $\boldsymbol{\lambda}$, it displays $\boldsymbol{q}$, which is equal to $\boldsymbol{\lambda}$ multiplied by A, plus a random number. The random numbers are generated according to the Poisson distribution: $p(q_d|\boldsymbol{\lambda}) = \mathcal{P}\left(q_d; [A\boldsymbol{\lambda}]_d\right)$. The random numbers make it difficult for you to guess $\boldsymbol{\lambda}$, says Bob, however, button M generates a new value of $\boldsymbol{q}$, while $\boldsymbol{\lambda}$ remains unchanged. The knob regulates, according to a deterministic set of rules, the values of A.

Alice is determined to understand how the machine generates $\boldsymbol{\lambda}$. She decides to write down the values of A for every possible position of the knob. Then she starts playing with the buttons and she writes down all the numbers that are generated by the machine. After a while, she has reason to believe that the values of $\boldsymbol{\lambda}$ are related spatially. On average, if the value of $\boldsymbol{\lambda}$ is large in location d, it seems to be large in the nearby display locations; if it is small, it appears to be small in the nearby locations. Alice realises that there is more; there seem to be certain patterns in the data, which she cannot yet describe.

At some point, Alice, frustrated by the attempt of understanding the set of rules that produces $\boldsymbol{\lambda}$,

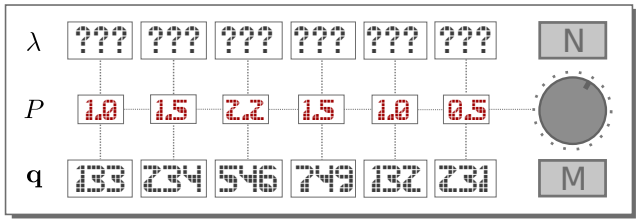


Figure 9.1: Machine for iterative design experiments. Button N generates a new value of $\boldsymbol{\lambda}$, button M generates a new value of $\boldsymbol{q}$, which is a noisy realisation of $A\boldsymbol{\lambda}$. The knob regulates the parameters of A. The machine does not display $\boldsymbol{\lambda}$. The set of rules that generates $\boldsymbol{\lambda}$ when N is pressed is unknown.

considers that if she kept pushing M infinite times, she would eventually know what is the true value of $\boldsymbol{\lambda}$ as long as all values of A are non-zero, which is the case for all knob positions. Then, she wonders, given the numbers that the machine is currently displaying ($\boldsymbol{q}_1$), obtained with the knob in the current position (ξ_1), in which position should she set the knob in order to acquire as much knowledge as possible about $\boldsymbol{\lambda}$ the next time she presses M. She decides to use the information that she has gained about $\boldsymbol{\lambda}$ and encodes it with the assumption that λ_{d+1} must be normally distributed if λ_d is known. She parametrises the normal distribution using the values that she had noted down. Her state of knowledge about $\boldsymbol{\lambda}$ is expressed by:

$$p(\boldsymbol{\lambda}|\boldsymbol{q}_1) = \frac{p_{\xi_1}(\boldsymbol{q}_1|\boldsymbol{\lambda})p(\boldsymbol{\lambda})}{p_{\xi_1}(\boldsymbol{q}_1)} \tag{9.12}$$

$$p_{\xi_1}(\boldsymbol{q}_1) = \int p_{\xi_1}(\boldsymbol{q}_1|\boldsymbol{\lambda})p(\boldsymbol{\lambda})\mathrm{d}\boldsymbol{\lambda} \tag{9.13}$$

Her state of knowledge after performing the measurement with the new parameters ξ_2 will be expressed by:

$$p(\boldsymbol{\lambda}|\boldsymbol{q}_1, \boldsymbol{q}_2) = \frac{p_{\xi_1}(\boldsymbol{q}_1|\boldsymbol{\lambda})p_{\xi_2}(\boldsymbol{q}_1|\boldsymbol{\lambda})p(\boldsymbol{\lambda})}{p_{\xi_1,\xi_2}(\boldsymbol{q}_1, \boldsymbol{q}_2)} \tag{9.14}$$

$$p_{\xi_1,\xi_2}(\boldsymbol{q}_1, \boldsymbol{q}_2) = \int p_{\xi_1}(\boldsymbol{q}_1|\boldsymbol{\lambda})p_{\xi_2}(\boldsymbol{q}_2|\boldsymbol{\lambda})p(\boldsymbol{\lambda})\mathrm{d}\boldsymbol{\lambda} \tag{9.15}$$

Integrating over all possible values of $\boldsymbol{q}_2$:

$$E_{\boldsymbol{\lambda}}(\xi_2) = \sum_{\boldsymbol{q}_2} p_{\xi_2}(\boldsymbol{q}_2|\boldsymbol{q}_1) \int p(\boldsymbol{\lambda}|\boldsymbol{q}_1, \boldsymbol{q}2) \log p(\boldsymbol{\lambda}|\boldsymbol{q}_1, \boldsymbol{q}2)\mathrm{d}\boldsymbol{\lambda} \tag{9.16}$$

$$p_{\xi_2}(\boldsymbol{q}_2|\boldsymbol{q}_1) = \int p_{\xi_2}(\boldsymbol{q}_2|\boldsymbol{\lambda})p(\boldsymbol{\lambda}|\boldsymbol{q}_1)\mathrm{d}\boldsymbol{\lambda} \tag{9.17}$$

$E_{\boldsymbol{\lambda}}(\xi_2)$ expresses the information that Alice expects to gain from the measurement performed with parameters ξ_2. Notice that this expression coincides with (9.11) if $p(\boldsymbol{\lambda}|\boldsymbol{q}_1)$ is replaced by $p(\boldsymbol{\lambda})$. In fact, according to the Bayesian formulation of paragraph 9.1, the optimum design depends on the prior probability distribution assigned to $\boldsymbol{\lambda}$. In the iterative design experiment performed by Alice, the first measurement defines the prior probability distribution of $\boldsymbol{\lambda}$ for the second measurement. Therefore

$p(\boldsymbol{\lambda})$ is replaced by $p(\boldsymbol{\lambda}|\boldsymbol{q}_1)$. The estimation of the information gain (9.16), therefore, entails the same computational complexity as the estimation of the information retrieval (9.11). However, after observing q_1, the posterior probability $p(\boldsymbol{\lambda}|\boldsymbol{q}_1)$ is reasonably narrow (while $p(\boldsymbol{\lambda})$ in (9.11) is very broad in case of smoothing prior). This consideration suggests to approximate $p_{\xi_2}(\boldsymbol{q}_2|\boldsymbol{q}_1)$:

$$p_{\xi_1}(\boldsymbol{\lambda}|\boldsymbol{q}_1) \approxeq \delta(\boldsymbol{\lambda} - \hat{\boldsymbol{\lambda}_1}) \tag{9.18}$$

$$\hat{\boldsymbol{\lambda}_1} = \underset{\boldsymbol{\lambda}}{\operatorname{argmax}}\, p(\boldsymbol{\lambda}|\boldsymbol{q}_1) \tag{9.19}$$

$$p_{\xi_2}(\boldsymbol{q}_2|\boldsymbol{q}_1) \approxeq \delta(\boldsymbol{q}_2 - A_{\xi_2}\hat{\boldsymbol{\lambda}_1}) \tag{9.20}$$

Substituting in (9.16):

$$E_{\boldsymbol{\lambda}}(\xi_2) \approxeq \int p(\boldsymbol{\lambda}|\boldsymbol{q}_1, A_{\xi_2}\hat{\boldsymbol{\lambda}_1}) \log p(\boldsymbol{\lambda}|\boldsymbol{q}_1, A_{\xi_2}\hat{\boldsymbol{\lambda}_1}) \mathrm{d}\boldsymbol{\lambda} \tag{9.21}$$

The integration over $\boldsymbol{\lambda}$ is computationally complex. One way to greatly simplify the problem is the Laplace approximation, discussed in the next paragraph (9.3). The posterior distribution of $\boldsymbol{\lambda}$ is approximated with a normal probability distribution, that is the logarithm of $p(\boldsymbol{\lambda}|\boldsymbol{q}_1, \boldsymbol{q}_2)$ is approximated with a second order polynomial. Such approximation reveals especially convenient, making the integration of $p(\boldsymbol{\lambda}|..) \log p(\boldsymbol{\lambda}|..) \mathrm{d}\boldsymbol{\lambda}$ trivial. The Laplace approximation of (9.14) is:

$$p(\boldsymbol{\lambda}|\boldsymbol{q}_1, \boldsymbol{q}_2) \approxeq \mathcal{N}\left(\boldsymbol{\lambda}; \hat{\boldsymbol{\lambda}_1}, \Gamma_2\right) \tag{9.22}$$

$$\Gamma_2 = [F_1 + F_2 + R]^{-1} \tag{9.23}$$

with F_1 being the Hessian of $\log p(\boldsymbol{q}_1|\boldsymbol{\lambda})$, F_2 the Hessian of $\log p(\boldsymbol{q}_1|\boldsymbol{\lambda}|)$ and R the Hessian of $\log p(\boldsymbol{\lambda})$, as explained in detail in the next paragraph. Shannon information (9.16) of a multivariate normal distribution is the determinant of the covariance matrix. If the measurement (button M) is repeated several times, the n-th optimum parameter is expressed, under the Laplace approximation, by:

$$E_{\boldsymbol{\lambda}}(\xi_n) = \frac{1}{2} \log 2\pi e \,\mathrm{DET}\left[\left[F_{\xi_1}(\boldsymbol{q}_1) + \sum_{i=2}^{n} F_{\xi_i}(A_{\xi_i}\hat{\boldsymbol{\lambda}_{i-1}}) + R\right]^{-1}\right] \tag{9.24}$$

$$\hat{\boldsymbol{\lambda}_n} = \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \sum_{i=1} n \log p(\boldsymbol{q}_i|\boldsymbol{\lambda}) + \log p(\boldsymbol{\lambda}) \tag{9.25}$$

With normal posterior, it is straightforward to compute the marginals. The information about the activity in voxel v is:

$$E_{\lambda_v}(\xi_n) = \frac{1}{2} \log 2\pi e \left[\mathrm{diag}\left[\left[F_{\xi_1}(\boldsymbol{q}_1) + \sum_{i=2}^{n} F_{\xi_i}(A_{\xi_i}\hat{\boldsymbol{\lambda}_{i-1}}) + R\right]^{-1}\right]\right]_v \tag{9.26}$$

where diag[] is the vector of the diagonal elements of the square matrix given as argument.

9.3 Laplace approximation and sparse Fisher Information Matrix

The posterior probability of the activity given the photon counts is expressed by the product of the Poisson likelihood and the prior. Though we can write the analytical expression of the posterior distribution, calculating its statistical properties such as the variance, its marginals and the information associated to the distribution and to the marginals is very computationally complex due to the large dimensionality of the problem. However such properties are trivial to compute for normal distributions. If the posterior probability distribution is twice differentiable, one may compute the parameters of the normal distribution that approximates the posterior. The derivatives of order greater than 2 of the log of a normal distribution are zero, as the log of the normal distribution is a polynomial of order 2. The posterior probability distribution may then be approximated with the probability distribution whose log is the second order polynomial with coefficients given by the Taylor expansion of the log of the posterior around its mode $\hat{\boldsymbol{\lambda}}$, truncated to the second order. Notice that the first derivatives vanish around the mode, therefore leaving only the second derivatives. The approximation is exact if the derivatives of the posterior of order greater than 2 are zero (in which case the posterior is normal). The matrix of the second order derivatives of the log of a normal distribution (the Hessian H) is the inverse of the covariance matrix, indicated with Γ. Therefore:

$$p(\boldsymbol{\lambda}|\boldsymbol{q}) \approxeq \mathcal{N}\left(\boldsymbol{\lambda}, \hat{\boldsymbol{\lambda}}, \Gamma\right) \tag{9.27}$$

$$\Gamma = H^{-1} \tag{9.28}$$

The Hessian of the log posterior is the sum of the Hessian of the log likelihood (F) and the Hessian of the log prior (R):

$$H = F + R \tag{9.29}$$

The Hessian of the log likelihood may be referred to as Fisher Information Matrix (FIM). In the case of Poisson likelihood, differentiating (2.45), F has the following expression:

$$F_{ij} = \frac{\partial^2}{\partial\lambda_i \partial\lambda_y} \log p(\boldsymbol{q}|\boldsymbol{\lambda})|_{\hat{\boldsymbol{\lambda}}} = \sum_{d=1}^{M} \frac{a_{id}a_{jd}}{\sum_{v=1}^{N} a_{vd}\hat{\lambda}_v} \tag{9.30}$$

Paragraph 9.3.2 describes how the elements of the FIM can be computed efficiently with a highly parallel computing system. Computing the parameters of the normal distribution and, consequently the information expressed in (9.24) and (9.26), requires the inversion of $H + R$.

9.3.1 Sparse Fisher Information Matrix

In the previous section we have described how the Fisher Information Matrix can be employed to characterize the uncertainty of the reconstruction. Unfortunately computing the FIM inverse is intractable since we are dealing with a large matrix of size $N \times N = (N_x \times N_y \times N_z)^2$.

A computationally efficient approximation for the calculation of the inverse of the FIM has been previously proposed in the context of the design of space-variant penalties that yield to space-invariant

impulse response functions [35] [36] [44]. Qi et al. [44] argued that if it is reasonable to assume that the FIM varies slowly with position and if one is interested in calculating the effects of a quadratic prior in terms of bias and variance in a voxel i, then it is acceptable to ignore the shift-variance of the FIM. The FIM is then replaced with a circulant matrix obtained by shifting the row corresponding to one voxel location (i.e. one value of i in (9.30)). This local approximation of the FIM is obtained by replacing all rows of the FIM with the shifted version of its $i-th$ row and then by inverting this shift-invariant matrix in order to estimate the variance associated to voxel i. This approximation is referred to as the *circulant approximation*, since it simply reduces the FIM to a circulant matrix. This makes the computation of the information tractable as the inverse of a circulant matrix is a circulant matrix whose row i can be computed using the Discrete Fourier Transform (DFT) (see e.g. [35]).

Such approximation, while appearing well suited for the estimation of the uncertainty for systems whose response can be approximated as shift-invariant, does not account for the global interdependence between the estimates in all the voxels and therefore can not incorporate the effects of long-range correlations (e.g., evaluation of the effects of data truncation or missing data).

This thesis proposes a different approach for calculating the inverse of the FIM. This formulation reduces the computational complexity in inverting the FIM but nevertheless accounts for the global interdependence between the variables. The FIM is calculated over a subset of the voxel indexes $\boldsymbol{G} \subset [1,\ldots,N]$ arranged in a grid that covers the whole volume. We define a sparse version of the FIM calculated over a subset $\boldsymbol{\lambda_G}$ of the full set of parameters $\boldsymbol{\lambda}$:

$$F_{ij}^{G} = \sum_{d=1}^{M} \frac{a_{id}a_{jd}}{\sum_{v=1}^{N} a_{vd}\lambda_v} \quad \text{with } i,j \in G \tag{9.31}$$

where a_{id} is the element of index (i,d) of the system matrix: $A = \{a_{id}\}$. The expression in equation (9.31) is equivalent to saying that, in the estimation of the covariance, we are accounting for the interdependence between a subset of voxels $i,j \in \boldsymbol{G}$ only, assuming that, for the remaining voxels $i,j \notin \boldsymbol{G}$, the MAP estimate $\hat{\boldsymbol{\lambda}}$ is equal to the true value of $\boldsymbol{\lambda}$. The inverse of such kind of sparse matrix is a matrix with the same structure. Therefore, the inverse of the covariance matrix is obtained by inverting $F_{ij}^{G} + R_{ij}^{G}$:

$$\Gamma(\hat{\lambda}_G) = [F^G + R^G]^{-1} \tag{9.32}$$

The number of elements in the full FIM equals N^2, whereas the number of elements of the sparse FIM equals N_G^2; therefore reducing the computational burden in inverting the sparse FIM.

The Hessian of the quadratic penalty $\boldsymbol{R}$ is not dependent on $\boldsymbol{\lambda}$ and therefore can be precalculated. Analogously to F, the sparse version of the quadratic penalty $\boldsymbol{R}^G$ is obtained by selecting the elements of the matrix R that correspond to the points in the grid.

Two examples of grids are pictured in Fig. 9.2, for a small imaging volume of $6 \times 6 \times 6$ voxels. In Fig. 9.2-A the grid accounts for the interdependence between every point in the imaging volume, in Fig. 9.2-B, the grid accounts for the interdependence between half of the voxels in the imaging volume.

This model allows the user to design the grid and therefore to define the degree of approximation in the calculation of the FIM.

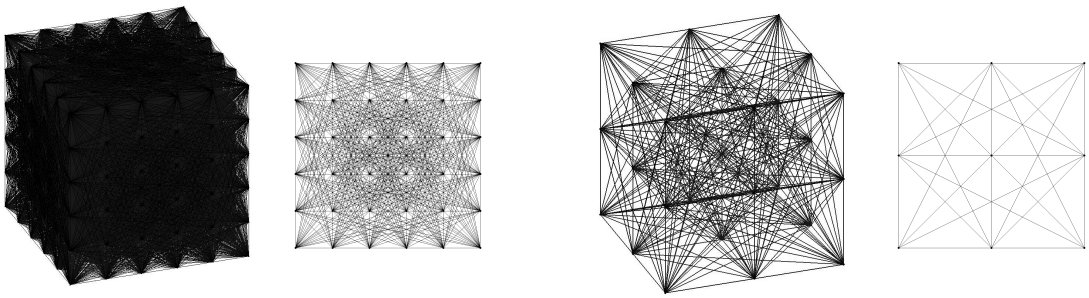


Figure 9.2: Example of grids for the estimation of the uncertainty. A (top left) the grid accounts for the correlation between every point in the imaging volume (full FIM). B (top right) the grid accounts for the correlation between $1/8$ of the voxels in the imaging volume (sparse FIM). C (Bottom right) The central plane of the grid displayed in A. D (Bottom left) The central plane of the grid displayed in B.

9.3.2 GPU Implementation

Every element of the FIM at the grid points is calculated exactly, accounting for the acquisition geometry and the object without further approximation. If the grid has N_G nodes, the FIM is of size $N_G \times N_G$ and symmetrical, so filling the matrix requires the computation of $\frac{1}{2}N_G^2 + \frac{1}{2}N_G$ elements. Naive computation of the FIM requires one projection for the denominator of (9.31) and M sums of products (SOPS) for each of the $\frac{1}{2}N_G^2 + \frac{1}{2}N_G$ elements of the half FIM. The proposed algorithm is inspired by the rotation-based algorithm proposed by Zeng and Gullberg [118]. The collimator-detector response is captured by a depth dependent Point Spread Function (PSF). Information being additive over the detector bins, the FIM element F_{ij} is the sum of F_{ij}^m contributions from the M camera positions indexed with $m = 1, \cdots, M$. The algorithm is based on interpolation of the activity and of the FIM grid on a regular grid aligned with each camera. By re-interpolating the activity and the FIM grid on a regular grid, the PSF can be applied more efficiently in the frequency domain as all points that are at a given distance lie on the same plane. The PSF is non-zero within a box X (see Fig. 9.3). The algorithm for the evaluation of the elements of the FIM consists of the following steps:

1) Compute projection of $\boldsymbol{\lambda}$ for each camera position.

2) Compute FIM elements for each camera position:

a. Re-sample the FIM grid positions on the voxel grid parallel to the camera by tri-linear interpolation.

b, For each pair of points i, j in the FIM grid

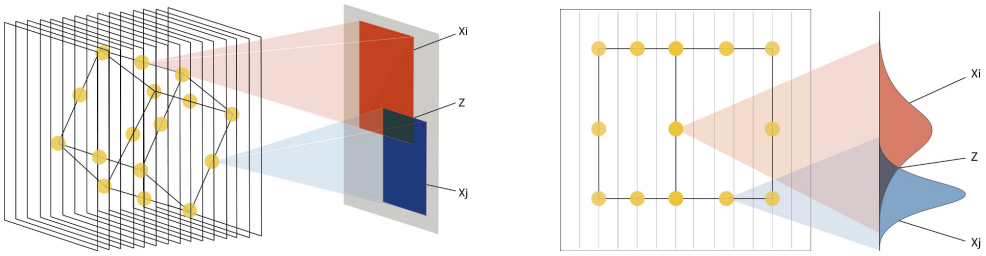


Figure 9.3: Rotation-based algorithm for fast computation of the Fisher Information Matrix: 3-D schematic representation. Algorithm: 1) For each Gamma camera (grey plane) position, the activity $\boldsymbol{\lambda}$ is resampled on a regular grid parallel to the camera plane and projected; 2) The grid points (yellow spheres) for the Fisher Information Matrix (FIM) are reinterpolated on the same parallel grid; 3) For each pair of points in the FIM grid, the FIM element is updated with the information relative to the current camera, which only depends on the region of the projection Z (black square) where the two PSFs X_i (red square) and X_j (blue square) intersect.

i. Compute coordinates of the box Z, on the camera plane, where the two PSFs X_i X_j intersect (if they intersect).

ii. If X_i and X_j intersect, update the FIM element by integrating (9.31) over the intersection box Z.

The algorithm is implemented in the CUDA programming language for parallel execution on Graphics Processing Units (GPU). Tri-linear re-sampling is performed in hardware by the texture fetch unit of the GPU at the cost of a single memory access. Coalesced memory access is achieved by partitioning the memory transfers in blocks. The convolutions are calculated with the 2D-FFT and IFFT routines included in the NVidia CUFFT library. A tailor made GPU kernel computes the projection (sum of planes) with high device occupancy and maximizes memory coalescing. A second kernel computes the integral in each intersection box (2-b-ii): each GPU thread computes the integral (for the current camera position), for a pair of points in the grid, so that the integrals for multiple pairs are evaluated concurrently on the multi-processors of the GPU. Each thread decides if the two PSFs intersect, then it loads from the global memory of the GPU device the sections of the PSFs that intersect and the projection data in the area of intersection Z (see Fig. 9.3). Finally, the thread computes the integral (2-b-ii) in the intersection box. After completion of the partial FIM for a single camera, the process is repeated for another camera, accumulating the elements of the FIM, as, according to (9.31), information is additive. Computation times are reported in table 9.1. The GPU-accelerated algorithm for the computation of the FIM has been integrated in the Niftyrec reconstruction software toolbox and has Matlab and Python interfaces which enable real time scripting interaction and full flexibility in the definition of the grid.

Table 9.1: Computation times for the calculation of the FIM and its inverse.

	g1	g2	g3
Grid Size	27648	6912	3072
NVidia GeForce GTX-285	348s	30s	19s

Multiple instances of noise	18 hours

9.3.3 Effect of the approximation of the FIM

In the following we explore the use of the approximate analytical method described in section 9.3.1 for the estimation of the uncertainty in emission imaging systems.

In order to evaluate the method described in the previous paragraph, one may proceed, as reference method, by sampling the posterior probability distribution with a posterior sampling technique. However, posterior sampling is very computationally expensive. An alternative to evaluate the effect of the sparse FIM is to employ the FIM to estimate the covariance matrix of the MAP estimator. The covariance of the MAP estimator, defined in 3 ((3.34)), is the matrix that describes how spread are the estimates of $\boldsymbol{\lambda}$ on average (over many repeated measurements) for a given true value of the activity. The covariance matrix of the MAP estimator, which we denominate in the reminder of this chapter Γ_{MAP} is (subtly) related to the covariance matrix that describes the uncertainty about $\boldsymbol{\lambda}$ from the single experiment Γ, defined in paragraph 9.3. While there is no explicit expression of the covariance matrix of the MAP estimator for the Poisson model, Fessler has derived the first order approximate expression:

$$\Gamma_{MAP}(\hat{\boldsymbol{\lambda}}) \approx [F+R]^{-1} \cdot F \cdot [F+R]^{-1} \tag{9.33}$$

The covariance matrix of the MAP estimator can be computed by performing the reconstruction, until convergence, of multiple instances of the measurements for a given true value of the activity, as described in chapter 3. In order to evaluate the effect of the sparse approximation, therefore we proceed by comparing Γ_{MAP} obtained with the analytical expression 9.33 with the covariance obtained with the multiple noise instances.

For the reference method, a series of independent noise realisations were computed using a pseudo-random Poisson noise generator (from the IRT toolbox [60]), based on the rejection sampling algorithm described at page 293 of [61]. The noisy data sets were reconstructed using the GPU-accelerated implementation of the One Step Late algorithm for MAP estimation, implemented as part of the NiftyRec toolbox. 10000 iterations were performed. A smoothing prior with a small weight was included in the cost function. The value of the smoothing parameter was chosen after trial and error, as a minimum value that guarantees convergence within 10000 iterations. The calculation of the variance was based on 10240 instances of the measurement (the number of noise instances is a multiple of 1024 as NiftyRec can

process concurrently up to 1024 reconstructions in order to make efficient use of the GPU. As discussed in chapter 3, the number 10240 was chosen in order to obtain visually satisfactory plots of the covariance matrix). Though often variance is calculated with much smaller sample size and number of iterations, we found that such large numbers are necessary to obtain estimates of the covariance comparable with the estimates obtained with the analytical methods based on the inversion of the Fisher Information Matrix.

The sparse version of the FIM has been calculated over three different grids of $g_1 = 27648$, $g_2 = 6912$ and $g_3 = 3072$ points equally distributed over the central axial slice and the two neighbouring slices. The volume was discretised in $96 \times 96 \times 12$ voxels, therefore grid g_1 samples fully over the three slices of interest $(27648 = 96 \cdot 96 \cdot 3)$. In the following, the FIM for grid g_1 will be referred to as the full FIM.

The variance images in Figures 9.4 (B-C-D), 9.5 (B-C-D) were obtained by re-ordering the diagonal of the covariance matrix Γ_{MAP} calculated as in (9.32). For the full FIM (g_1), the diagonal of Γ_{MAP} is simply reshaped to a three dimensional matrix, whereas for grids g_2 and g_3 every point of the diagonal of Γ_{MAP} is allocated to the respective points of the grid in the imaging volume. A tri-linear interpolation is then performed in order to facilitate the visual comparison between the variance images obtained with the different grid strides. It should be noticed that a direct interpolation on Γ_{MAP} can not be performed.

A more closely spaced grid gives a more precise estimation of the variance but increases the computational complexity of the estimation. The computation time needed to calculate the FIMs and their inverse are presented in table 9.1. Variance images obtained with the circulant approximation are also presented for comparison.

We performed two different sets of experiments with different software phantoms and different system models. First, the calculation of the variance obtained with the approximated analytical method based on the sparse FIM is validated for a realistic phantom. Secondly, in order to emphasise the benefits of the sparse approximation of the FIM with respect to the circular approximation, we show how it can be employed to calculate the reconstructed image quality in the case of region-of-interest reconstruction from truncated projection data.

Both experiments are based on the simulation of the response of a standard SPECT system with parallel-hole collimator.

NCAT Phantom The SPECT system is based on a detector system of size $236.16\ mm \times 236.16\ mm$. The detector rotates over 360° at a regular angular step of 2° around the centre of the imaging volume. The imaging volume dimensions are $96 \times 96 \times 12$ cubic voxels of $2.4\ mm$. Photon counts are binned on a grid of 96×96 pixels of $2.46\ mm$ in size and the detector is placed at a distance of $133\ mm$ from the centre. We consider a low energy high resolution (LEHR) parallel hole collimator, characterised by point spread function (PSF) with full width at half maximum $FWHM = 7.89\ mm$ and efficiency 1.55%. In this study we assume that the detectors have a perfect absorption efficiency and a perfect intrinsic detector resolution.

The first experiment is performed simply to validate the analytical method with a realistic phantom. The phantom used for this experiment was a heart phantom (NCAT) [119]. The activity within the phantom was $\lambda = 8kBq/cm^3$ in the left and right ventricle myocardium, $\lambda = 3.4kBq/cm^3$ in the left and right ventricle chamber, $\lambda = 0.9kBq/cm^3$ in the lungs and $\lambda = 0.6kBq/cm^3$ in the background. The variance images for the NCAT phantom are shown in Fig. 9.4. Fig. 9.4-A shows the variance image obtained from the reconstruction of 10240 noisy projection data sets. Fig. 9.4 B-C-D show the corresponding images predicted with the full FIM for grid $g_1 = 27648$ and with the method based on the sparse Fisher Information matrix for grid $g_2 = 6912$ and $g_3 = 3072$ respectively. Fig. 9.4-E shows the variance image predicted with the circulant approximation of the FIM. Fig. 9.4-F shows the horizontal profiles. From these images we can see how both the analytical calculation method based on the sparse Fisher Information Matrix and the circulant approximation of the FIM approximately predict the variance of the MAP estimator, presenting minor, but obvious, differences with respect to the variance obtained with the reference method. The variance image obtained from the reconstruction of multiple noise realisations is rather noisy, due to the limited number of repeated experiments. The variance images predicted with the FIM method appear smooth as neighbouring voxels suffer from similar levels of noise. However a sparser grid gives a more approximated estimation, as we can see in Fig. 9.4-F. In fact, a fundamental limitation of the sparse FIM approach is that fine detail is being lost as the grid becomes more sparse. However, the algorithm enables the degree of approximation in the estimation to be defined by the user. Therefore a trade-off between computational complexity and reliability of the estimation of the covariance matrix arises. The minimum number of grid points necessary to obtain a reliable estimation of the covariance depends on the characteristics of the system under investigation. Both the method based on the sparse FIM and the method based on the circulant approximation of the FIM are somewhat less accurate near the edge of the finite support used in image reconstruction, for unknown reasons. This effect (which has been reported also in another study [120]) can lead to a discrepancy exceeding 10% and it is more noticeable, in case of low level of activity, in the off-centre voxels of the phantom [50].

Truncated Projection Data

In this experiment we investigate the effect on image variance for region-of-interest reconstruction from truncated projection data. Truncation is caused by a limited detector size. Only a certain number of detector bins w are used to measure the emission data. The Field Of View (FOV), in this condition of truncated emission data, is defined as the cylindrical region, whose radius depends on the level of truncation, obtained by rotating the truncated camera. The phantom was a uniform sphere positioned in the centre of the image space, with diameter $D = 24.6\ mm$. The activity in the sphere was set to $\lambda = 8kBq/cm^3$ and to $\lambda = 4.4kBq/cm^3$ in the background. The size and the position of the sphere has been chosen in order to have the sphere always in the FOV, whereas a larger and larger part of the background activity falls in the region outside the FOV for larger levels of truncation. During the

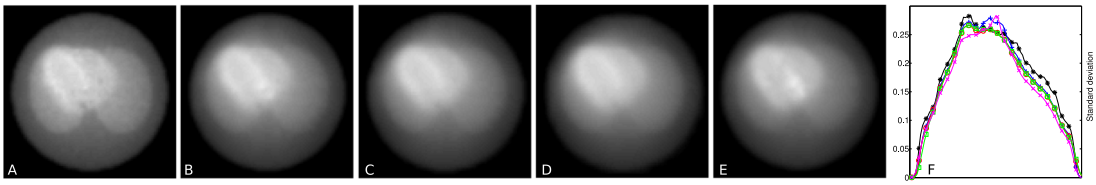


Figure 9.4: Variance images for the NCAT phantom obtained with a standard SPECT system. A (first column) - reference method (variance image obtained from the reconstruction of 10240 noisy projection data sets), B (second column) - Full Fisher Information-based method with grid g_1, C (third column) - Sparse Fisher Information based method with grid g_2, D (fourth column) - Sparse Fisher Information based method with grid g_3, E (fifth column)- Variance image obtained with the Circulant Approximation method. F (sixth column)- Variance profiles at the centre of the heart: reference method (black line - $*$), grid g_1 (blue line - $+$), grid g_2 (red line - $\circ$), grid g_3 (green line - $\square$), circulant (magenta line - $\times$).

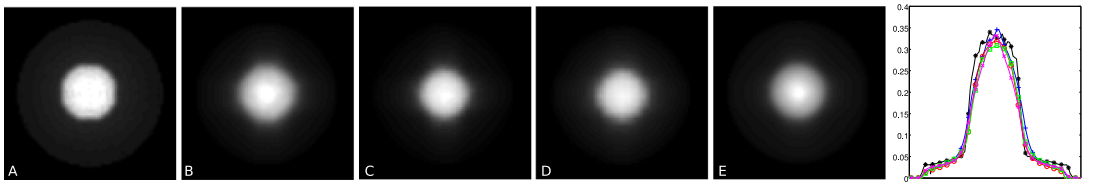


Figure 9.5: Variance images of a uniform sphere obtained with a standard SPECT system. A (first column) - reference method (variance image obtained from the reconstruction of 10240 noisy projection data sets), B (second column) - Full Fisher Information-based method with grid g_1, C (third column) - Sparse Fisher Information based method with grid g_2, D (fourth column) - Sparse Fisher Information based method with grid g_3, E (fifth column)- Variance image obtained with the Circulant Approximation method. F (sixth column)- Variance profiles at the centre of the heart: reference method (black line - $*$), grid g_1 (blue line - $+$), grid g_2 (red line - $\circ$), grid g_3 (green line - $\square$), circulant (magenta line - $\times$).

experiment, the FOV diameter varies from $w = 96$ to $w = 16$ (from 236.16 mm to 39.36 mm).

Fig. 9.6 displays variance images from truncated projection data with a FOV diameter $w = 36$. The variance image obtained from multiple noisy data sets and the variance images obtained from the full FIM and the sparse FIM method with different grids are in good agreement. For the voxels outside the FOV, the variance increases considerably in respect to the non truncated case. Outside the FOV, in fact, we do not have full sampling, since we acquire data from that region only at certain angular positions of the camera. The variance image obtained with the circulant approximation of the FIM method is displayed in Fig. 9.6 - E. From this image, we can notice an increase in variance in the voxels outside the FOV with respect to the non-truncated case. However the aforementioned effect is less accentuated with respect to the increase in variance estimated with full FIM method and with the sparse FIM method in the same region. The horizontal profiles are shown in Fig. 9.6-F.

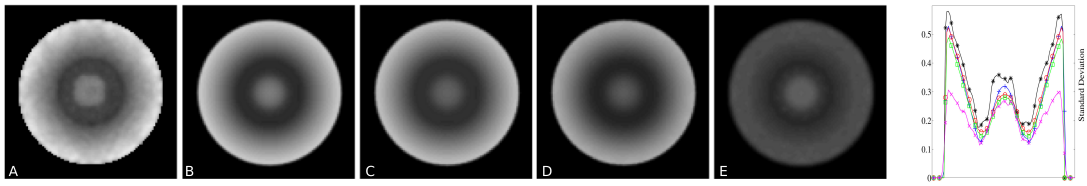


Figure 9.6: Interior tomography: Variance images for a uniform sphere phantom obtained with truncated projection data with FOV diameter $w = 36$. A (first column) - reference method (variance image obtained from the reconstruction of 10240 noisy projection data sets), B (second column) - Full Fisher Information-based method with grid g_1, C (third column) - Sparse Fisher Information based method with grid g_2, D (fourth column) - Sparse Fisher Information based method with grid g_3, E (fifth column)- Variance image obtained with the Circulant Approximation method. F (sixth column)- Variance profiles at the centre of the heart: reference method (black line - $*$), grid g_1 (blue line - $+$), grid g_2 (red line - $\circ$), grid g_3 (green line - $\square$), circulant (magenta line - $\times$).

In Fig. 9.6, the variance associated to the voxel at the centre of the sphere is plotted for different FOV diameters $w = 96, \cdots, 16$. The calculation of the variance is obtained with the reference method (reconstruction of 1024 noisy data sets), with the sparse FIM approach for (with grid of $g_1 = 27648$ (full FIM), $g_2 = 6912$, $g_3 = 3072$ points) and, for comparison, with the circulant approximation based on a single row of the FIM. Even if Region Of Interest (ROI) reconstruction from truncated projections data can lead to nearly unbiased reconstruction in a well-sampled ROI (as demonstrated in [18] [121]), we noticed that a decrease in FOV size leads to an increase in variance, not only outside the FOV but also inside it. For this specific experiment we observe an overall decrease in variance of 10% compared to the non-truncated case, using the reference method based on multiple noise realisations for the calculation of the variance. An important observation is that we see no effect due to truncation with the circulant approximation of the FIM (as stated in [43]) whereas with the sumbsampled FIM, since we account for the interdependence between the voxels, we see an increase of the variance with increased level of truncation. This is an important feature of the method that we have introduced, because it enables the optimisation of systems for interior imaging and of imaging systems that depart from the uniformly sampled circular trajectory of SPECT.

The circulant FIM is generally considered to yield a good approximation of the covariance matrix for nearly shift-invariant systems, however 3-D imaging systems are inherently shift variant, presenting a block-circulant FIM even in case of ideal uniform object in the FOV. Comparison of the variance (the diagonal of the covariance matrix) obtained from the circulant approximation, with the full FIM and with the reference statistical method, has highlighted certain pitfalls of the circulant approximation. In the following, the link between the shift-variance of the system and the different approximations of the FIM is illustrated with two examples. Figure 9.8 shows the full Fisher Information Matrix (Figure 9.8

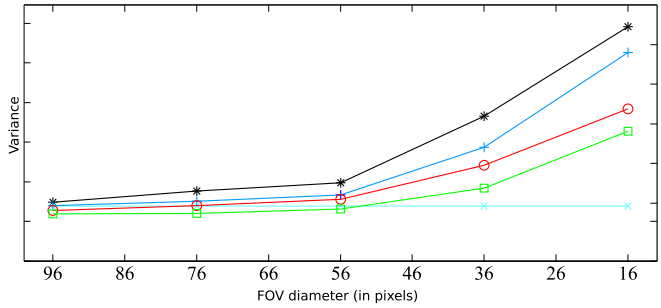


Figure 9.7: Interior tomography: variance for different levels of truncation for a voxel in the centre of the sphere. FOV diameter (from left to right) $w = 96, \cdots, 16$. Black line $(*)$ - variance obtained with the reference method. Blue line $(+)$ - variance obtained with full FIM method with grid g_1. Red line $(\circ)$ - variance obtained with sparse FIM method with grid g_2. Green line $(\square)$ - variance obtained with sparse FIM method with grid g_3. Magenta line $(\times)$ - variance obtained from the circulant approximation method.

A-D) a sparse Fisher Information Matrix with grid g_2 (Figure 9.8 B-E) and the Circulant Fisher Matrix (Figure 9.8 C-F) for the experiment with the Truncated Projection Data. It should be noticed that Fig. 9.8 C-F shows the circulant Fisher Matrix for the calculation of the variance of a voxel of interest i in the centre of the FOV; whereas Fig. 9.8 A-D shows the FIM for all the points in the FOV and Fig. 9.8 B-E shows the FIM for the points of grid g_2.

As already described in section 9.3.2, every element of the sparse FIM at the grid points is calculated exactly, accounting for the acquisition geometry and for the object. We can see, in fact, how in the sparse FIM we account for the system response and for the object dependency, whereas with the circulant approximation method we make the assumption that the FIM (and therefore the system response) is shift invariant. The number of elements in the full FIM and in the circulant FIM equals N^2, whereas the number of elements of the sparse FIM equals $(N/2)^2$; therefore highly reducing the computational burden in inverting the sparse FIM. Since the FIM is a huge matrix and therefore difficult to display, we show the FIMs for the 2-D case. Therefore the full FIM has $g_1 = 9216$ points and the sparse FIM has $g_2 = 2304$ points. However, what is ultimately of interest is the inverse of the FIM (the Covariance matrix) Figure 9.9 shows the inverse of the full FIM (Figure 9.9 A) the inverse of the sparse FIM with grid g_2 (Figure 9.9 D) and the inverse of the FIM calculated with the circulant approximation method (Figure 9.9 B) for the Truncated Projection Data experiment. Fig. 9.9 B shows a circulant FIM inverse, where each column of the matrix is evaluated independently. The resulting circulant FIM inverse is spatially variant (non circulant) but clearly does not show the same structure as the full FIM inverse in Fig. 9.9 A. We can therefore deduce that the circulant FIM can not incorporate the effects of shift-variant systems (Fig. 9.9) and also can not account for the effects of data truncation or missing data. A direct visual comparison between the inverse of the full FIM and the inverse of the sparse FIM is arduous, because of

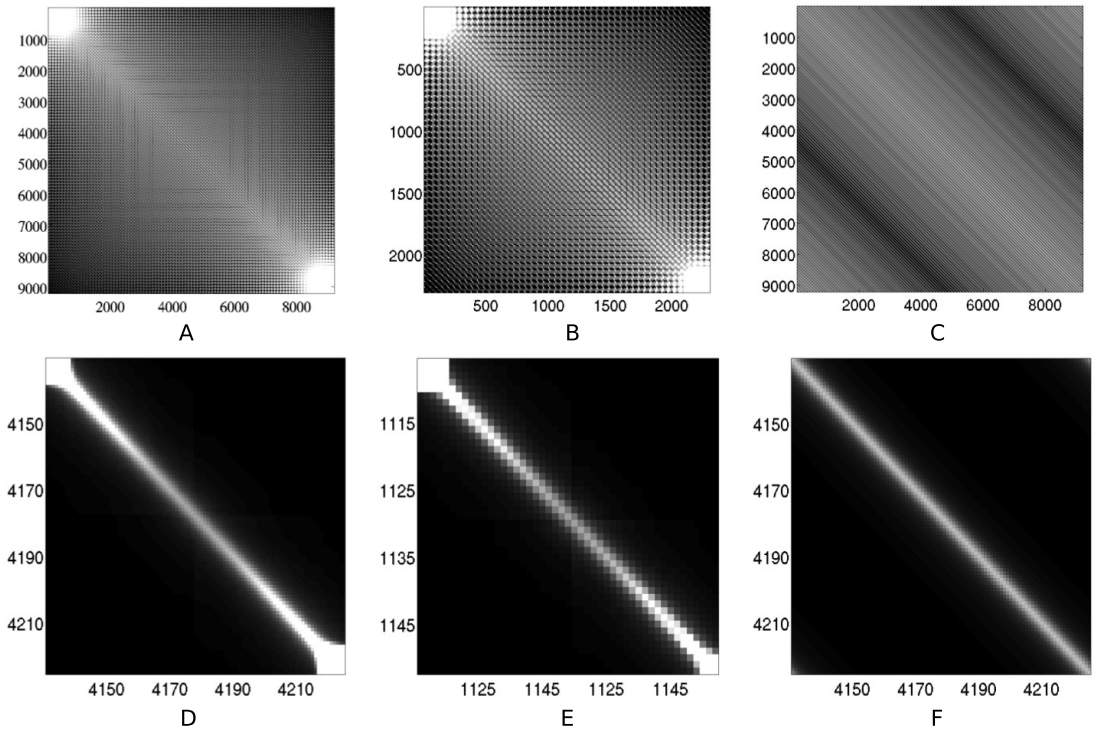


Figure 9.8: Fisher Information Matrix for the Truncated Projection Data experiment. Comparison between full FIM, sparse FIM and Circulant FIM. A (top left) - Full FIM, grid $g_1 = 9216$ points. B (top central) - sparse FIM, grid $g_2 = 2304$ points. C (top-right) - Circulant FIM for a voxel of interest i in the centre of the FOV only. D (bottom left) - A zoom-in part of A. E (bottom central) - A zoom-in part of B. F (bottom right) - A zoom-in part of C.

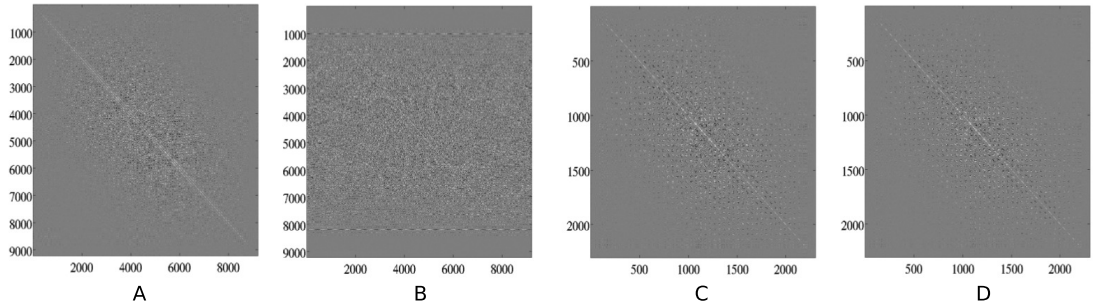


Figure 9.9: Covariance matrix for the Truncated Projection Data experiment. Comparison between inverse of full FIM, sparse FIM and Circulant FIM. A - Inverse of the Full FIM, grid $g_1 = 9216$ points. B - Inverse of the FIM using the circulant approximation method. C - obtained selecting the voxels at the locations of the full FIM that corresponds to locations of the elements of the grid for the sparse FIM. D - Inverse of the sparse FIM, grid $g_2 = 2304$ points.

the different size of the two matrices. Hence we show, in Figures 9.9 C, two matrices which are obtained selecting the voxels at the locations of the inverse of the full FIM that corresponds to locations of the elements of grid g_2 and then rebinning the selected voxels in a smaller matrix of size $(N/2)^2$. Clearly, the inverse of the sparse FIM will not be exact at the grid points because of the missing off-diagonal FIM entries between the grid points, however the matrices in 9.9 C and 9.9 D exhibit the same structure. This encompass the capability of the method to incorporate non-stationary system functions and effects of long-range correlations.

The sub-sampled FIM trades off computational complexity and accuracy of the estimation, enabling an adaptation of the accuracy of the estimation based on the available computational resources. When sufficient resources are available, the GPU-accelerated software described in section 9.3.2 can compute the covariance matrix exactly on a grid g_1. One important advantage of the scalable sub-sampled FIM approximation is that the algorithm provides an estimate of the full covariance matrix, though sub-sampled, accounting for the global interdependence between the variables of the tomogram. This enables the use of global metrics for system design optimisation.

Evaluation of the effect of sub-sampling, however, is complicated by the trade-off that arises. It is not possible to define an absolute criterion for the choice of the sub-sampling scheme. The contributions to the FIM at a given camera position, for a given pair of grid points, arise only from the overlap in the projected PSF from those points (as shown graphically in Figure 9.3). The implication is that grid points must be close enough in order to have non-zero overlap between those projected PSFs. This condition depends on many factors, among which: the image volume size, the voxel size, the size of the PSF, the camera trajectory etc etc. This condition refers to the accuracy of the FIM entry for those two points, but does not apply to accuracy of its inverse, which will suffer from missing points even if the overlapping PSF condition is met. Therefore, a general criterion to define a relationship between the grid stride and the obtainment of a reliable estimation of the covariance matrix, can not be provided. This criterion depends in fact on the properties of the specific system. The sparse FIM formulation and the software tool described in this chapter may be employed for the optimisation of a range of design parameters of emission imaging systems. However only two guidelines can be given, so far, for the choice of the sub-sampling scheme. The first is trivially to adopt the most dense grid for the available computational resources. The second is to restrict the grid volume to a specific region of the FOV, in case we know in advance that activity is present only in that region of interest.

9.4 Active Learning Tomography

The measure of uncertainty based on the FIM accounts for the complex interdependence of the information about the activity through the imaging volume. We can then explore new geometries that depart from the rotation at regular steps around an axis. In the concept camera pictured in 9.12, the algorithm for the estimation of the uncertainty has been coupled with a random walk algorithm in order to compute adaptively an optimum acquisition strategy. The camera performs an initial scan on a circular trajectory, then a new trajectory that maximises the information about the activity in the voxel at the centre of the volume is calculated with a random walk algorithm. Trajectories are sampled from a b-spline curve with random walk over the nodes of the spline, as depicted in Fig. 9.10. Trajectories are parametrised as b-splines. The random trajectory sampler of Fig. 9.10 returns a new random trajectory, given the previous trajectory, by drawing the locations of the nodes of the spline from independent normal distributions centred at the previous locations of the nodes. Given the current trajectory, the design optimisation algorithm performs a random walk by drawing randomly a series of trajectories. For each new random candidate trajectory, the information gain is computed according to (9.26) and the candidate trajectory is accepted (yellow trajectories in Fig. 9.11) if the information gain is positive. The process is repeated, until N_{opt} successive samples do not increase the information gain. At this point the trajectory is accepted (red trajectories in Fig. 9.11) and new emission data is acquired according to the new trajectory.

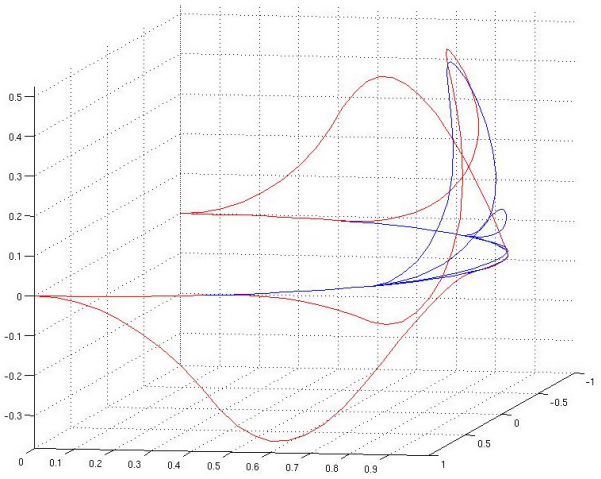


Figure 9.10: Active learning SPECT: random trajectory sampler.

Fig. 9.11 and Fig. 9.12 display the results of a synthetic experiment. The object is composed of a hot sphere at the centre of the imaging volume and of a second sphere in the periphery. With the initial path of the SPECT camera (circular trajectory between 0^o to 180^o), the outer sphere and the inner sphere overlap along the path of the camera. The design optimisation algorithm was parametrised for the maximisation of the information about the central voxel (equation (9.26)). The volume was discretised in $128 \times 128 \times 128$ voxels and the FIM was sampled on a grid every other 4 voxels along each axis.

N_{opt} was set to 10. The design optimisation algorithm, whose random candidates and final path are reported in Fig. 9.11, selected a trajectory that is close to the circular trajectory but does not intersect the peripheral sphere. Fig. 9.12 reports the viewgrams for the initial trajectory and for the final trajectory.

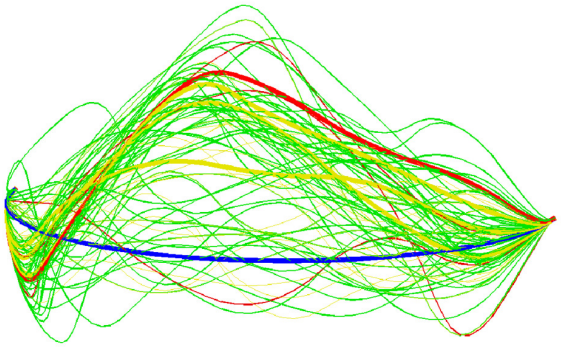


Figure 9.11: Active learning SPECT. The green lines are the randomly sampled trajectories. For each trajectory, the expected uncertainty is calculated with the algorithm based on the sub-sampled FIM. The yellow curves are the new candidates that increase the information gain. The red curves are the trajectories that maximise the information gain. A new scan is performed for each red curve. The image displays 3 iterations of the design optimisation algorithm. Fig. 9.12 reports the viewgrams for the initial trajectory and for the final trajectory.

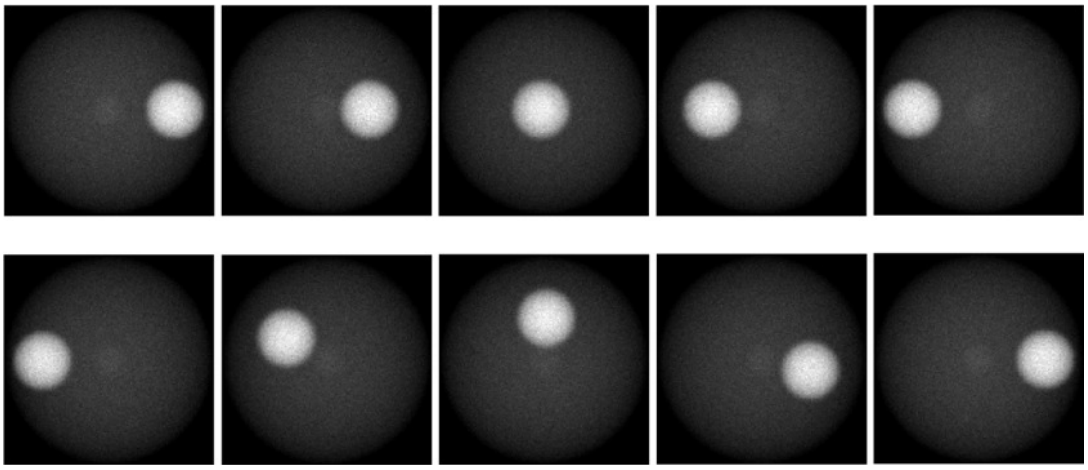


Figure 9.12: Active learning SPECT. The top images show the projections at 0^o,45^o,90^o,135^o,180^o for the initial circular trajectory. After 3 iterations of design optimisation, the trajectory avoids the intersection with the sphere in the peripheral region of the imaging volume.

Chapter 10

NiftyRec: unifying tomography toolkit

The work described in this thesis is embodied in the NiftyRec software toolkit for tomographic imaging. NiftyRec is *unifying* in three ways:

- integrating tomographic reconstruction, registration and segmentation in a unified mathematical and software framework
- supporting multiple tomographic imaging modalities through a unifying software interface
- providing a unified interface for third-party open source tomographic reconstruction software

The modular software design of NiftyRec is inspired by the graphical models described in Chapters 4 and 5, enabling the tight integration of tomographic reconstruction with image analysis, registration and segmentation in a unified framework. The multiple projectors and back-projectors, at the core of NiftyRec, enable multiple imaging modalities, including PET, SPECT, transmission tomography with cone-beam geometry (X-Ray CT and Tomosynthesis), fan-beam geometry (fan-beam X-Ray CT) and parallel beam geometry (Synchrotron X-Ray Tomography, Neutron Tomography and Optical Projection Tomography (OPT).

In order to address the near real-time demands of clinical applications, NiftyRec is designed for execution on the massively parallel computing architectures of off-the-shelf Graphics Processing Units (GPUs). The unprecedented performance of NiftyRec enables the exploration of complex models, analysis of the properties of optimisation algorithms at convergence and four-dimensional tomography. The modular design and the live scripting interfaces for Matlab and Python foster fast prototyping and development.

Wrapper interfaces for the open source reconstruction software Image Reconstruction Toolbox (IRT) [60] and the Software for Tomographic Image Reconstruction (STIR) provide unified access to projection and back-projection routines, fostering interoperability [122].

10.1 A software design pattern for the integrated modelling paradigm

Unifying mathematical models are delineating a paradigm shift away from sequential processing towards integrated modelling of the imaging process. The integrated modelling approach shifts the focus of the imaging process from the creation of images to direct extraction of clinically-relevant information and visualisations from the raw data produced by the imaging hardware.

Such paradigm shift is permeating the field of tomographic reconstruction, with the introduction of models that join information from multiple modalities, informing the reconstruction with data previously considered exogenous to the standalone tomographic imaging systems. Numerous instances of the integrated modelling approach can be found in the field of nuclear imaging. Examples are the use of the X-Ray transmission data for attenuation-aware reconstruction of the emission tomograms in PET/CT (see e.g. [123]), and the use of complementary imaging modalities to constrain the tomographic reconstruction, discussed in chapter 4. In the field of image analysis methodology, the trend towards integrated modelling has been outlined in seminal works such as in the Unified Segmentation framework [5] and in the work by Fischl et al. [124]. Both papers combine, in unifying mathematical frameworks, the three processing steps of tissue classification, bias field correction and non-linear warping, otherwise commonly performed in sequence. Tomographic reconstruction, however, along with eventual de-noising and other corrections, is still considered a pre-processing stage. This thesis has explored the convergence of the two trends towards integrated modelling observed in the fields of tomographic reconstruction and image analysis methodology and the combination of information from multiple tomographic imaging modalities within such framework. The integrated modelling paradigm, as observed in [5], generally makes better use of the information available from the data. The perceived disadvantages of the integrated approach, on the other hand, are the increased complexity of the software and the increased demand of computing resources.

- *Computation time:* In the context of tomographic reconstruction, while iterative algorithms have proven superior, in many applications, to direct inversion methods, their computational complexity is still largely a limit to widespread use in research and clinical settings. The even greater complexity of the algorithms devised within the integrated modelling approach thus constitutes a major development challenge.

- *Software complexity:* In the integrated modelling approach a sequence of simple computer programs is replaced by highly specialised monolithic programs. This makes it more difficult to write re-usable software and to mix and match different programs.

- *Accessibility:* The monolithic nature of the algorithms that emerge from the integrated modelling paradigm calls for software design patterns that address re-usability. Free availability of such

software is essential in order to foster experimentation, development and assessment of models and algorithms.

The large computation time of tomographic reconstruction algorithms has been addressed with solutions based on: *1)* numerical representations of the imaging data and discretisation schemes that promote computational efficiency [125, 126]; *2)* the use of dedicated high performance processors [127] and distributed computing systems [128]; *3)* optimisation algorithms with fast convergence rates [31]; *4)* use of priors in order to accelerate convergence [129]. Though a number of algorithms based on 1),2),3), 4) have been described in the literature, only a few open source software tools for tomographic image reconstruction are currently available. These include the Matlab based Image Reconstruction Toolbox (IRT)[60] and the Software for Tomographic Image Reconstruction (STIR) [122]. IRT provides a large collection of functions for the reconstruction of SPECT, PET, CT and MRI tomograms, making it useful tool for the exploration of new algorithms. IRT however is based on Matlab and therefore unsuited for the deployment of stand-alone applications. STIR is a C++ based software for the reconstruction of PET tomograms, designed for the deployment in research and clinical applications. Reconstruction algorithms are applied to the raw data files from a scanner or to simulated data by means of command-line tools. STIR implements state of the art iterative reconstruction algorithms (MLEM, OSEM) and handles file formats for several commercial PET scanners. The monolithic nature of the command-line interface and the steep learning curve for the development through the C++ API limit the freedom to experiment with new algorithms.

The design of NiftyRec addresses the challenges of the integrated modelling paradigm, conjugating computational efficiency and accessibility of the code. Bayesian Networks inspire the design of the modular application programming interface, providing a unifying interface for multiple imaging modalities, including Single Photon Emission Computed Tomography (SPECT), Positron Emission Tomography (PET) and transmission tomography with multiple geometries. Computational efficiency is achieved by making use of the Single-Instruction-Multiple-Data (SIMD) architecture of commercial off-the-shelf Graphics Processing Units (GPUs).

10.1.1 Design Principles

The shift of paradigm of the imaging process mandates a shift of paradigm in the software design. The sequence of simple computer programs of the sequential approach is replaced by monolithic programs in the integrated modelling approach, posing the challenge of writing modular and re-usable software components.

The sequential approach finds its natural expression in the *pipeline* design pattern, which inspires many software frameworks and common practice. Notable instances of the pipeline design pattern are the filter update mechanism of the Insight Toolkit (ITK) [130] and the widespread adoption of command line tools. Tools for medical image processing are often intended as binary programs that operate on input

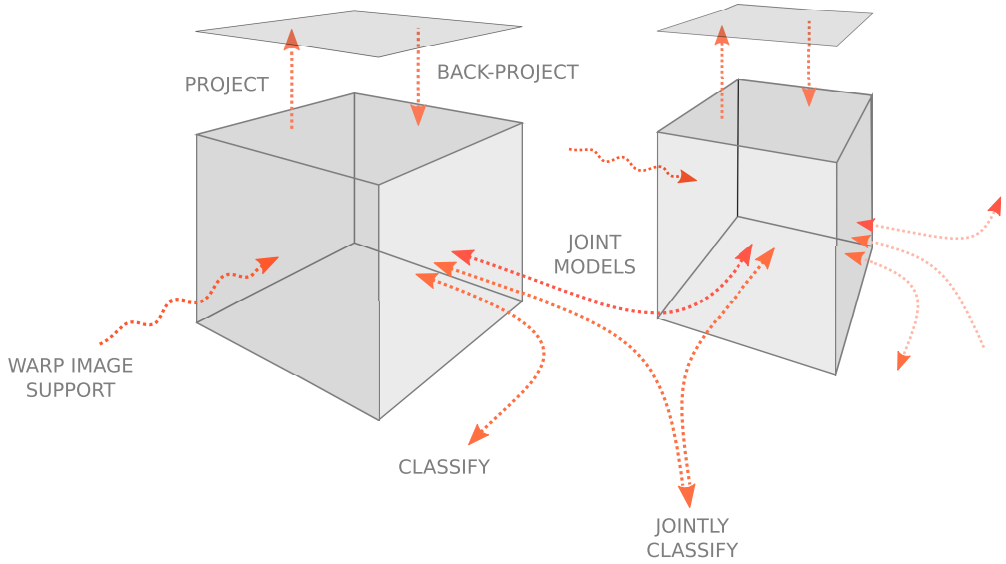


Figure 10.1: Integrated modelling.

files and produce output files. Programs are executed as standalone processes and mixing and matching is specified with command line scripts. Such means of interaction are effective in the sequential paradigm, however the integrated modelling paradigm demands tighter integration of the software components, posing the challenging problem of writing re-configurable software.

In NiftyRec, this challenge and the challenge of computational efficiency are addressed by a software design that leverages the efficiency of ANSI C, the interactivity and inherent modularity of high level programming language Matlab and Python, and computational power of GPUs.

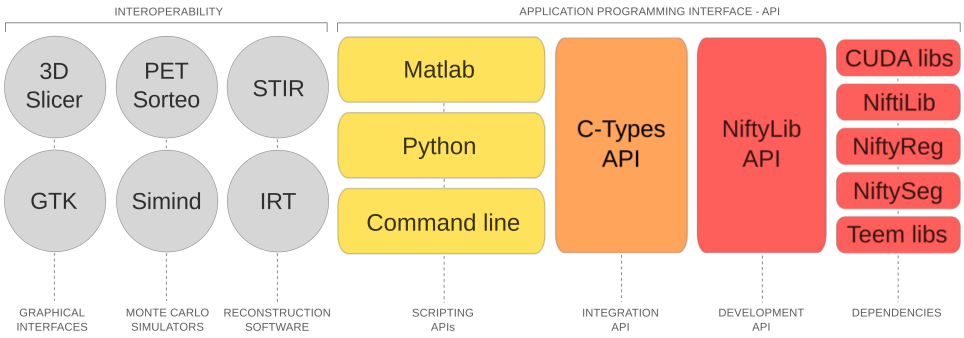


Figure 10.2: NiftyRec 1.6 software design.

Core routines are written in ANSI C with the NVidia Compute Unified Device Architecture (CUDA) language extensions for GPU acceleration. The code compiles into a set of shared libraries that provide two application programming interfaces (APIs): a low level development API based on the NiftiLib data structures and a high level API for integration into third-party applications, which exposes plain ANSI C types (Figure 10.2). The low level interface is aimed at simplifying the development of NiftyRec.

Though the programming style is that of plain C, the imaging data is represented with the data structures defined in the NiftiLib libraries. The use of such data structures minimises the number of function parameters. The high level C-types API is a thin layer built on top of the development API, aimed at simplifying the integration of NiftyRec into third-party software and the construction of bindings for high level programming and scripting languages.

Python and Matlab APIs are build on top of the C-types API, linking dynamically to the NiftyRec libraries. NiftyRec appears to the Matlab and Python programmers as a Matlab Toolbox and as a Python module, hiding entirely the complexity of the underlying algorithms and GPU acceleration. The Python package provides the same functionalities as the Matlab toolbox, though in a more object-oriented fashion. By making use of the dynamic allocation and reference counting features of the Matlab and Python interpreters, the NiftyRec shared libraries are *stateless*, making the scripting interaction free from memory management errors.

The mixing and matching of software components involved in the definition of new algorithms is specified by means of the Matlab and Python interfaces. The Matlab and Python interpreters may be considered as a mechanism for real-time, interactive linking of the functions defined in the shared libraries, replacing the hard linking that occurs when building monolithic command line applications.

Though MLEM and other reconstruction algorithms could be performed entirely on the GPU, increasing slightly the computational efficiency, the software design trades off computational efficiency and modularity. With the stateless interface, imaging data is pushed to the GPU only while a function call from the scripting interfaces is being executed. This mechanism entails the transfer of the imaging data from the RAM memory of the host computer to the GPU at each function call. While this mechanism may seem to add latency, projection and back-projection functions are designed to perform all the projections at once (i.e. for multiple detector locations in CT and for the entire gantry in PET), effectively masking the memory transfer latency.

Table 10.2 reports a list of the functions available through the C-Types, Matlab and Python APIs; Table 10.1 reports a list of the features of NiftyRec and Table 10.3 reports a list of the demo scripts, which illustrate the basic functionality of NiftyRec.

10.1.2 Inter-operability

The C-types interface and the Matlab and Python bindings simplify the integration into third party software and the development of plugins and other mechanisms that render NiftyRec interoperable with other software packages. A plugin for 3D Slicer based on the Slicer 3 Execution Model enables OSEM reconstruction for SPECT within the 3D Slicer environment. The NiftyRec Matlab Toolbox interfaces with the Image Reconstruction Toolbox (IRT) [60], with the Software for Tomographic Image Reconstruction (STIR) and with the Monte Carlo simulators Simind and PET-Sorteo. Projection and back-projection functions of IRT and of STIR are wrapped, providing drop-in replacement functions in Matlab for the NiftyRec projection and back-projection routines. This enables immediate comparison of performance

and image quality. The interfaces to the Monte Carlo simulators Simind and PET-Sorteo simplify the development of simulation studies.

10.2 Efficient projection and back-projection

Computing the registration, reconstruction and segmentation jointly is costly. Calculating the derivatives of the nodes potentials described in Chapters 3, 4 and 5 requires the computation of projections and back-projections, exponentials (Gaussian likelihoods), spatial gradients, linear interpolations and 3-D transformations. The largest computational burden, however is attributed to the projections and back-projections and the non-linear spatial transformations. Computing and storing the projection matrix $P = \{p_{bd}\}$ is problematic due to its very large size. However, due to the geometrical characteristics of the imaging systems, the systems matrix is sparse and structured. Sparsity and structure enable us to apply the matrix and its transpose (i.e. to compute the projection and back-projection) efficiently without ever storing the matrix. Projection and back-projection may be obtained by combining ray-tracing and convolution operators. Ray-tracing, convolutions and the transformation of spatial coordinates are computing tasks that adapt well to the computing architectures of Graphics Processing Units (GPUs).

For problems that present enough task parallelism, high-end GPUs can provide an acceleration of up to $10\times$ over high-end multi-core CPUs, however the speed-up can increase by another order of magnitude if the structure of the algorithms allows for efficient use of the *shared memory* of the GPU [131, 132]. While on CPUs the cache hierarchy compensates costly accesses to external RAM and cache heuristics account for a large class of computational problems, the simplified memory hierarchy of GPUs requires careful design of the algorithms for efficient memory access. The external memory is directly exposed to the programmer, who has to consider explicitly coalesced access due to the mismatch between data rate and cycle time of the DDR RAM memory of the GPU device. On the other hand, the simpler structure of the memory and vicinity of the RAM to the processor yield data throughput up to about $10\times$ higher than the throughput between CPU and RAM.

The fast RAM memory of the GPUs therefore explains the $10\times$ speedup often immediately achieved by porting applications to GPU; however the Single Instruction Multiple Data (SIMD) architecture and the *shared memory* may provide additional speedup if the computational problem can be formulated in a way that suits the architecture of the GPU. In the SIMD architecture, many simplified low footprint processor cores fit on the same chip grouped in *multiprocessors* with a single common fetch unit. Multiple cores can operate concurrently in a multiprocessor if they execute the same instruction. Therefore, if the computational problem is such that the same operation is performed on multiple segments of data, the GPU can use a great number of processors at the same time. However, if the degree of parallelisation of the computational problem is sufficiently high to employ a great number of processor cores, the data throughput of the GPU DDR RAM is often not sufficient to continuously feed data to all the processors. A key feature of GPU architectures is a small (a few $Kbytes$) on-chip multiple data-path *shared*

memory which can be read and written concurrently in a single clock-cycle by the processor-cores of a same multiprocessor. Efficient use of the shared memory is a key for the development of highly optimised algorithms for GPU. The computational power of the hundreds of cores of the GPUs can be fully exploited, essentially, if the computational problem is such that it can be decomposed in parallel task executed by independent cores and the cores in a multiprocessor can re-use the data that resides in the shared memory, minimising access to the global GPU DDR RAM memory. In order to take full advantage of the architecture, the computational problem needs not only to expose parallel tasks that run on the multiple cores, but the tasks executed by cores that are part of the same multiprocessor must be serialisable in smaller tasks that share a limited amount of data and re-use it.

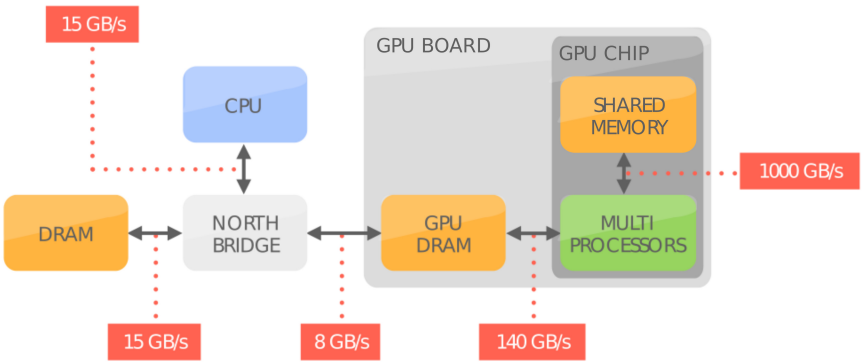


Figure 10.3: Memory hierarchy of a personal computer with General Purpose Graphics Processing Unit - GPU. The high data throughput of the RAM memory installed on-board of the GPU explains the $10\times$ speed-up often immediately achieved when porting applications to GPU. In order to exploit fully the computing resources of the GPU, the computational problem has to be formulated in a way that exposes a high number of parallel tasks characterised by memory locality. Exposing possibly identical parallel tasks enables the efficient use of the Single Instruction Multiple Data (SIMD) architecture of the multi-processors of the GPU (high *device occupancy*), while memory locality enables the efficient use of the *shared memory*. The rotation-based projector of NiftyRec achieves high device occupancy and exploits the shared memory, achieving a speed-up of up to $80\times$ when compared with the CPU implementation of NiftyRec and with third party open source reconstruction software (see Fig. 10.13 and Fig. 10.14).

Another feature offered by GPUs is hardware interpolation. A portion of memory may be declared as a 1-D, 2-D or 3-D array to the memory management unit (MMU) of the GPU. The MMU then accepts read requests based on floating-point coordinates, it decodes each request into the memory addresses of the nearest memory locations and returns a value calculated by linear (bi-linear, tri-linear) interpolation. Such hardware feature can provide dramatic speed-ups when interpolation is frequent, as is the case for ray-tracing and for rigid and non-rigid image deformation.

10.2.1 Emission Imaging

The forward projection consists in simulating the propagation and detection of light. This is a tremendously complex problem due to scattering of photons throughout the propagating medium. However in emission computed tomography, as opposed to other tomographic problems (such as diffused optical imaging), the energy of the radiation and size of the imaging volume are such that the ratio between detected un-scattered and scattered photons is relatively high. Though projectors and back-projectors that simulate scatter have been described [26], due to their tremendous computational complexity, emission computed tomography, is currently limited to rejection of the scattered photons. A comprehensive review of algorithms for the selection of un-scattered photons can be found in [133]. When the reconstruction of the emission rate tomogram is based uniquely on the un-scattered photons, the system matrix P, due to the use of planar cameras, shift invariant collimators and rotation symmetry, is sparse and structured [134]. A number of algorithms for projection and back-projection have been described. One class of algorithms exploits the sparsity of the system matrix by pre-computing the elements of the matrix and storing them with a sparse representation (see e.g. IRT [60]). Another class of algorithms exploits the sparsity and, more directly, the structure of P. The structure of P, in fact, corresponds to a natural sparse representation and often to a factorisation of the matrix. The characteristics of the detection system and the attenuation map entirely define the system matrix. The description of the imaging system, along with the attenuation map, can already be considered a convenient sparse representation of the matrix. With such class of algorithms, the factorisation of the matrix corresponds to direct computation of $P\mathbf{x}$ and $P^T\mathbf{x}$ from the description of the imaging system and attenuation map. We prefer such class of algorithms that do not rely on an intermidiate sparse representation as they do not require a costly pre-processing step and enable more flexibility: the attenuation map can change and eventually be estimated along with the activity, without re-calculating the matrix in the intermediate representation.

Single Photon Emission Computed Tomography

In SPECT, the probability that a photon emitted in v is detected in d, discarding attenuation throughout the patient, repeats over and over as v moves on a plane that is parallel to the detector plane (shift invariant) and repeats again for each camera. The response of the imaging system is captured by the depth-dependent point spread function (DD-PSF) and by the spatial position of the camera. The matrix P can be factored into two matrices, one that accounts for the DD-PSF and another matrix that accounts for the attenuation map. Without going in further detail on the factorisation of the system matrix, which underlies the rotation-based algorithm described by [134], it suffices to say that, approximating the integral attenuation from v to d with the attenuation from v to the detector element d' that lays on the camera plane in the location with highest sensitivity to v, the projection and back-projection consist in modulating with the DD-PSF the response of an ideal Gamma Camera with attenuated line integrals. Efficient computation of the line integrals for projection and back-projection by ray-tracing was proposed by Siddon [135]. However with a ray-based approach, it becomes inefficient to include the depth-dependent

PSF, as this requires the casting of a number of rays within a *cone of response* [134].

NiftyRec is inspired on the rotation-based algorithm proposed in [134], referred in the following as re-gridding. Such algorihtm is well suited for GPU acceleration as it re-organises the data into a regular grid, exposing data locality and yielding efficient use of the shared memory. For each position of the gamma camera, the image matrix volume is re-interpolated on a regular grid so that the front face of the re-sampled volume faces the detection plane - Figure 10.4. After re-interpolation, all the point sources that lay approximately on a same plane parallel to the detector are now at the same distance from the detector. The depth-dependent PSF can be incorporated efficiently by convolving each parallel plane with the PSF corresponding to the relative distance from the plane to the camera.

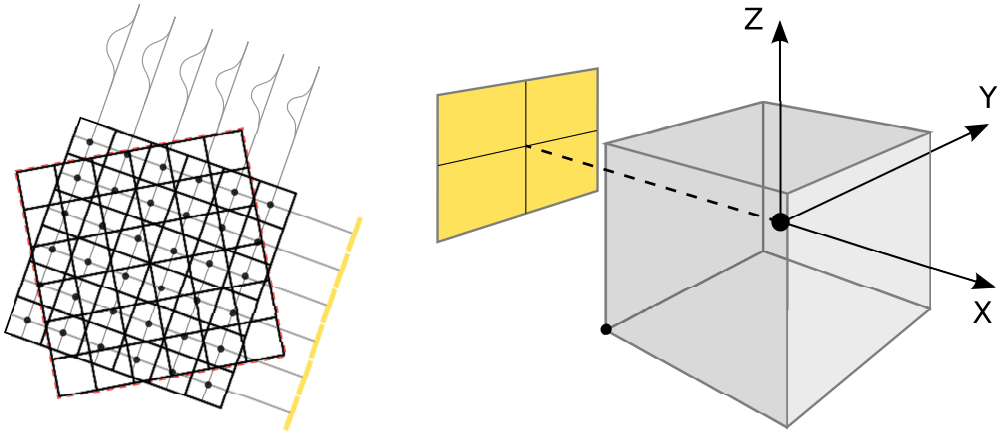


Figure 10.4: Schematic representation of the projector and backprojector for SPECT.

- **Projection:**

  Activity and the depth dependent PSF are copied to the GPU global memory and additional memory is allocated for the sinogram. The support of the image (ordered list of the x, y, z indexes of the image voxels) is extracted and stored in global memory.

  For each position of the gamma camera, the activity matrix volume is rotated so that the front face of the volume faces the detection plane. In order to optimize the usage of the GPU, rotation is performed by multiplying the support of the image by the rotation matrix, then the image is re-sampled at the locations specified by the rotated support. Rotation of the support maximizes the *device occupancy* (concurrent usage of the multiprocessors) and takes advantage of the *shared memory* by partitioning the matrix multiplication [136]. The trilinear interpolation is performed in hardware by the *texture fetch* unit of the GPU at the cost of a memory access and coalesced memory accesses are obtained by partitioning the memory transfers in blocks.

For each camera's position, the activity is rotated from its initial position, rather than from the previous camera's position, in order to minimize interpolation errors. After re-interpolation, each image plane parallel to the camera is convolved with the PSF. Convolution is performed by zero-padding the plane to double its linear size, computing its 2D FFT, multiplying by the FFT of the zero-padded PSF, back-transforming and truncating. The convolution is performed in place, in order to minimize memory occupancy. 2D FFT is performed by means of the CUDA CUFFT library that takes into account all the architectural factors and constraints of CUDA, memory coalesced access, bank conflicts and efficient shared memory usage.

Finally a kernel sums all planes and stores the result in the sinogram data structure. Shared memory cannot be used in the summation step as the number of operations (sums) is exactly equal to the number of memory accesses, however *device occupancy* and memory coalescing are optimized by partitioning the sums in blocks.

- **Back-Projection:** The sinogram and the PSF are transfered from the host machine RAM to the GPU global memory. Two volumes are allocated for the backprojection, a rotating volume and a fixed volume that is initialized to 0 and contains in the end the result of the backprojection.

  One projection at a time is extracted from the sinogram data structure and the value on each pixel is backprojected to the rotating volume along lines perpendicular to the detection plane. This step is performed by a GPU *kernel* that copies a pixel to a local *register* and then copies it back into all the voxels in the same column of the rotating volume. Each plane of the rotating volume is then convolved in place with the PSF by multiplication in the frequency domain. The rotating volume is rotated to the zero position and then accumulated into the fixed volume.

  The same series of operations is repeated for each camera's position.

Positron Emission Tomography

The projector/back-projector pair for PET is an extension of the projector/back-projector pair for SPECT, based on the re-gridding algorithm. In order to account for the many geometries of PET imaging systems, the emission imaging data is abstracted through the concept of the *viewgram*, i.e. the emission data is considered to be detected by a series of virtual planar detectors. Emission data measured in the cylindrical coordinate system of the PET scanner is re-binned a viewgram before reconstruction.

Due to re-binning, parallel lines of response (LORs) are mapped to a non-regularly spaced grid in viewgram space. In order to account for such distortion, often disregarded, NiftyRec warps the image

support prior to integration when performing the forward projection and warps accordingly the back-projection, as depicted in Fig. 10.5. The parameters $dx1, dx2, \ldots, dy1, dy2, \ldots$ are descriptive of the geometry of the system. Resolution modelling, as described in the previous section, is obtained by convolving each plane parallel to the viewgram plane before integrating along lines parallel to the viewgram plane. Dually, the partial back-projection for a given plane of the viewgram is convolved with a depth-dependent point spread function. The convolution, however, is performed in the spatial domain (not in the frequency domain), enabling the definition of a spatially variant convolution kernel (Fig. 10.5-left).

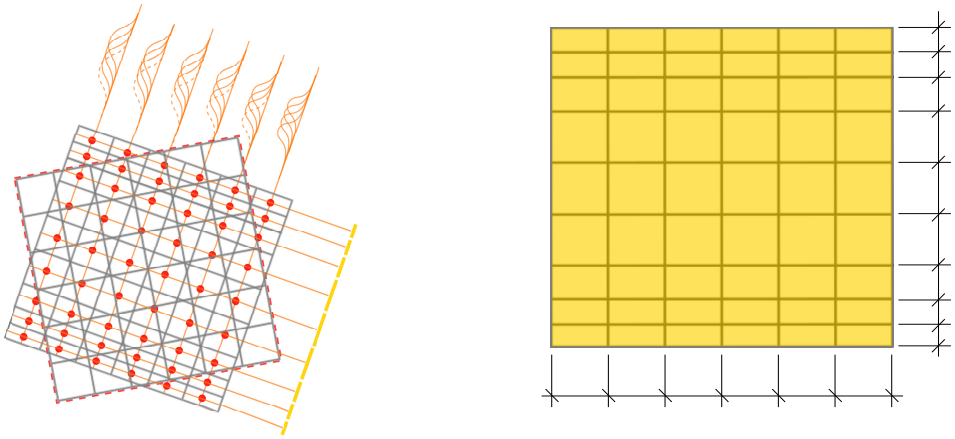


Figure 10.5: Schematic representation of the projector and backprojector for PET.

10.2.2 Transmission Imaging

NiftyRec implements projector and back-projector pairs for parallel-beam and for cone-beam geometries. These support the development of one-step and iterative reconstruction algorithms for multiple imaging modalities such as X-Ray CT, Tomosynthesis, Synchrotron X-Ray Tomography, Neutron Tomography and Optical Projection Tomography (OPT).

Cone-beam

The cone-beam projector and back-projector are based on an efficient *ray-tracing* algorithm. While a solution similar to the re-gridding algorithm for emission imaging could be implemented for cone-beam transmission imaging by replacing the rotation with an affine transformation, ray-tracing is simpler and more accurate for the cone-beam geometry. The ray-tracing algorithm implemented in NiftyRec is based on the efficient ray-box intersection algorithm described on the Siggraph Education website `http://www.siggraph.org/education/materials/HyperGraph/raytrace/rtinter3.htm`. Each GPU thread casts one ray from the X-Ray source to one of the detector pixels, computes the coordinates of the intersection with the volume of the attenuating medium and integrates the attenuation coefficient along the ray, re-sampling with tri-linear interpolation the attenuation coefficient at regular intervals. Integrals are stored on shared memory to maximise the efficiency and the

attenuation coefficients are fetched through the cached *texture memory* which caches reads that are in common between neibouring threads and provides hardware tri-linear interpolation. The back-projector is the dual of the projector, each thread reads a value from one pixel and integrates it into the back-projection volume at regular steps with tri-linear interpolation. In order to avoid race conditions, threads are launched in batches on a chess-board structure.

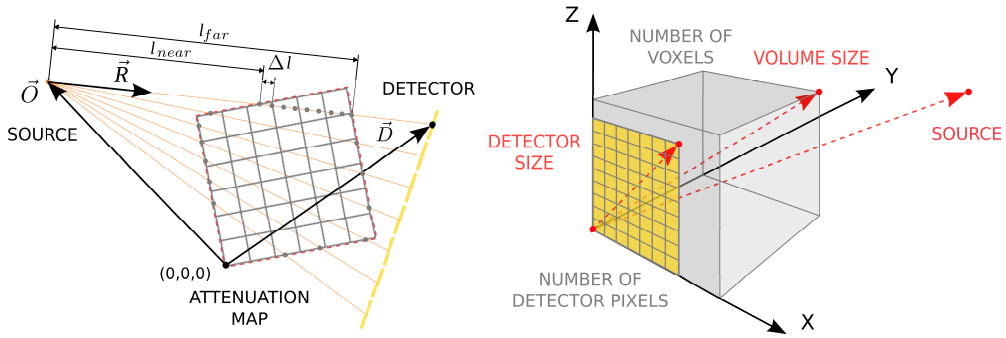


Figure 10.6: Schematic representation of the cone-beam projector for transmission imaging.

Parallel-beam

The parallel-beam projector and back-projector pair is based on the re-gridding algorithm described in 10.2.1. The advantage over ray-tracing is that the re-gridding algorithm avoids race-conditions in the back-projector. While strategies to avoid the race conditions typically limit the performance of the back-projector on parallel architectures, the re-gridding algorithm guarantees equal performance of the projector and back-projector.

10.3 List of features

Table 10.2 reports a list of the functions available through the Integration API and the Scripting API (see also Fig. 10.2). Table 10.1 reports a synthetic list of the features of NiftyRec and Table 10.3 reports a list of the demo scripts which illustrate the main features of NiftyRec. Figures 10.7-10.10 display the reconstructions obtained by executing the Matlab demos *et_demo_mlem*, *et_demo_pet_sorteo*, *tt_demo_mlem_conebeam* and *tt_demo_parallelbeam*, described briefly in the caption of the figures and in Table 10.3. The source code of *et_demo_mlem* (Fig. 10.7) is reported here:

```
Nx          = 192;                  % number of voxels x
Ny          = 128;                  % number of voxels y
Nz          = 192;                  % number of voxels z
iterations  = 200;                  % MLEM iterations
camera      = linspace(0,pi,180)';  % trajectory of the camera
fwhm        = 2.0;                  % FWHM of the non-DD PSF
GPU         = 1;                    % 1 to use the GPU
```

```
phantom       = et_spherical_phantom(Nx,Ny,Nz,Nx/2,Ny/3,Nz/2,0,100);
mumap         = et_spherical_phantom(Nx,Ny,Nz,Nx/2,Ny/2,Nz/2,0,1);

psf           = repmat(fspecial('gaussian',[5,5],fwhm),[1,1,Nx]);
sinogram      = poissrnd(et_project(phantom,camera,mumap,psf,GPU));
normalisation =  et_backproject(ones(Nx,Ny,180),camera,mumap,psf,GPU);

activity      = ones(Nx,Ny,Nz);
for i=1:iterations
  ratio       = sinogram./et_project(activity,camera,mumap,psf,GPU);
  activity    = activity./normalisation.*et_backproject(ratio,camera,mumap,GPU);
end
```

Table 10.1: Features of NiftyRec 1.6

- Projectors
  - *rotation-based* projector and back-projector for emission imaging (CPU and GPU)
  - *cone-beam* projector and back-projector for transmission imaging (CPU and GPU)
  - *parallel-beam* projector and back-projector for transmission imaging (CPU and GPU)
- Reconstruction algorithms
  - Maximum Likelihood Expectation Maximisation - MLEM .
  - Ordered Subsets Expectation Maximisation - OSEM .
  - Iterated Conditional Modes - One Step Late - ICM-OSL
  - Gradient Ascent (GA)
- Scatter correction algorithms
  - Reconstruction-based Scatter Correction - RBSC (as described in [137]).
- Regularisation
  - Markov Random Field with quadratic penalty [Chapter 3.2.2]
  - Non-parametric Entropy [Chapter 6]
- Multi-modal joint generative models [Chapter 4]
  - Non-parametric Joint Entropy - JE
  - Non-parametric Mutual Information - MI
  - Semi-parametric Joint Entropy - SPJE
  - Multivariate Gaussian Mixture Model - GM
- Deformation models
  - Affine transformation
  - Cubic-spline deformation model
- Other Features
  - Computation of the Fisher Information Matrix - FIM (CPU and GPU) [Chapter 9].
  - Massively parallel reconstructions - (CPU and GPU).

Table 10.2: NiftyRec 1.6 Integration API and Scripting API

- Common to all imaging modalities
  - Allocatio of the computing resources
    - **nr_list_gpus** Returns a list of the GPUs installed in the system and their compute capability.
    - **nr_set_gpu** Sets the GPU to be used by NiftyRec. It is used in conjunction with **nr_list_gpus**.
    - **nr_isinstalled** Checks if NiftyRec libraries are installed.
  - Image transformation
    - **nr_resize** Resizes a 3-D image by scaling.
    - **nr_rotate** Rotates a 3-D image in three-dimensions.
    - **nr_irotate** Inverse of **nr_rotate** when given the same parameters.
    - **nr_affine** Affine transformation of a 3-D image.
    - **nr_convolve** 3-D convolution.
  - File input/output
    - **nr_load_nifti** Loads a NIFTI file (.nii)
    - **nr_save_nifti** Saves a NIFTI file (.nii)
  - Synthetic data
    - **nr_spherical_phantom** Creates a spherical imaging phantom
    - **nr_helical_phantom** Creates a helical-shaped imaging phantom
    - **nr_apply_lesions** Applies synthetic lesions to a brain phantom
  - Random function generators and probability density functions
    - **nr_poissrnd** Applies Poisson noise to high dimensional data.
    - **nr_normal_density** Returns the normal probability density function for given parameters.
  - Multi-modality
    - **nr_joint_entropy_nonparametric** Non-parametric kernel density estimation
    - **nr_joint_entropy_semiparametric** Semi-parametric kernel density estimation
- Emission tomography
  - Projection and back-projection
    - **et_project** Parallel-beam projection with depth-dependent resolution modelling
    - **et_backproject** Parallel-beam back-projection with depth-dependent resolution modelling
    - **et_project_irt** Same as **et_project** but based on IRT.
    - **et_backproject_irt** Same as **et_backproject** but based on IRT.
    - **et_project_stir** Same as **et_project** but based on STIR.
    - **et_backproject_stir** Same as **et_backproject** but based on STIr.
  - Optimisation
    - **et_mapem_step** One-step-late maximum-a-posteriori expectation-maximisation (and MLEM).
    - **et_osmapem_step** Ordered-subsets one-step-late maximum-a-posteriori expectation-maximisation (and OSEM).
  - File input/output
    - **et_load_ecat** Load ECAT files.
    - **et_load_simind** Load data files produced by the Simind Monte Carlo simulator.
  - Simulation
    - **et_simind_simulate** Run the Simind Monte Carlo simulator.
  - Definition of the scanner geometry
    - **et_collimator_parallelholes** Defines the parallel-hole-collimator geometry for **et_project** and **et_backproject**
  - Advanced
    - **et_fisher_grid** Computes the Fisher Information Matrix (FIM) and the sub-sampled FIM.
    - **et_gradient_attenuation** To compute the attenuation map from the emission data.

- Transmission tomography
  - Projection and back-projection

    tt_project_ray_cone Cone-beam ray-traced projector.

    tt_backproject_ray_cone Cone-beam ray-traced back-projector.

    tt_project_parallel Parallel-beam rotation-based projector.

    tt_backproject_parallel Parallel-beam rotation-based back-projector.
  - Optimisation

    tt_mapem_step One-step-late maximum-a-posteriori expectation-maximisation (and MLEM).

    tt_osmapem_step Ordered-subsets one-step-late maximum-a-posteriori expectation-maximisation (and OSEM).
- Deformation models
  - Cubic-spline

    reg_control_points_from_affine Creates matrix of control points of the cubic-sline.

    reg_gaussian_smooth 3-D Gaussian smoothing.

    reg_resample_spline Re-sample a 3-D image according to the cubic-spline parametrisation.

    reg_autoscale_image Auto-scale image for the NMI registration of two images.

    reg_gradient_bending_energy_nodes Derivative of the bending energy of the cubic-spline model w.r.t. the displacement of the nodes of the cubic-spline.

    reg_gradient_jacobian_determinant_nodes Derivative of the Jacobian of the deformation field w.r.t. the displacement of the nodes of the cubic-spline.

    reg_gradient_NMI_voxels Partial derivative of the Normalised Mutual Information w.r.t. the image intensity.

    reg_gradient_SSD_voxels Partial derivative of the Sum of Squared Differences w.r.t. the image intensity.

    reg_gradient_voxels_to_nodes Propagates the derivatives w.r.t. the image intensity to nodes displacements.

    reg_gradient_image_voxels Spatial gradient of a 3-D image.
- Classification
  - Multi-variate Gaussian mixture model

    seg_initialise Initialise classifier.

    seg_terminate Terminate classifier.

    seg_set_input_image Set the input images.

    seg_set_mask Binary mask to exclude certain regions of the image.

    seg_set_priors Set the statistical atlas.

    seg_set_biasfield_parameters Set the parameters of the bias field model.

    seg_set_MRF_strength Set the parameters of the Potts model.

    seg_set_regularisation_covariance Regularisation parameter for the covariance matrices.

    seg_set_mean Set the initial value of the mean of each Gaussian function.

    seg_set_covariance Set the initial value of the covariance matrices.

    seg_set_segmentation Set the initial value of the partial membership.

    seg_get_input_image Get the input image data.

    seg_get_mask Get the input mask.

    seg_get_priors Get the input statistical atlas.

    seg_get_biasfield Get the estimated bias field.

    seg_get_segmentation Get the estimate of the partial membership.

    seg_get_mean Get the estimate of the mean of each Gaussian.

    seg_get_covariance Get the estimate of the covariance matrices.

    seg_get_loglikelihood Get the likelihood value.

    seg_get_biasfield_parameters Get the input parameters of the bias field model.

    seg_get_MRF_strength Get the input parameters of the Potts model.

    seg_get_regularisation_covariance Get the input regularisation parameter for the covariance matrices.

Table 10.3: NiftyRec 1.6 demos

- Emission Tomography
  - **et_demo_mlem** MLEM reconstruction SPECT with depth-dependent resolution modelling (see Chapter 3).
  - **et_demo_osem** OSEM reconstruction SPECT with depth-dependent resolution modelling (see Chapter 3).
  - **et_demo_osem_2D** 2-D OSEM reconstruction SPECT with depth-dependent resolution modelling.
  - **et_demo_mapem_mrf** Maximum-a-posteriori reconstruction with smoothing prior, one-step-late EM algorithm (see Chapter 3).
  - **et_demo_noise_instances** Reconstruction of multiple instances of noise (see Chapter 3).
  - **et_demo_mlem_scatter_rbsc** Reconstruction-based scatter correction algorithm (see [137]).
  - **et_demo_fisher_information** Estimation of the Fisher Information Matrix, 3D SPECT (see Chapter 9).
  - **et_demo_fisher_information_2D** Estimation of the Fisher Information Matrix, 2D SPECT.
  - **et_demo_petsorteo** Load data from PETSorteo simulation and reconstruct with OSEM.
  - **et_demo_simind** Load data from Simind simulation and reconstruct with OSEM.
  - **et_demo_mixture_model** Bivariate Gaussian mixture model for anatomically-driven reconstruction (see Chapter 4).
  - **et_demo_nonparametric_model** Non-parametric anatomically-driven reconstruction (see Chapter 6).
  - **et_demo_semiparametric_model** Semi-parametric anatomically-driven reconstruction (see Chapter 6)
  - **et_demo_4D_mlem_nonrigid** Four-dimensional reconstruction with non-rigid motion (see Chapter 5).
  - **et_demo_4D_motion_from_data_rigid** Estimation of the rigid motion parameters from the emission data (see Chapter 5).
  - **et_demo_estimate_attenuation** Estimation of the attenuation map from the emission data.
- Transmission Tomography
  - **tt_demo_project_conebeam** Cone-beam projection.
  - **tt_demo_project_parallelbeam** Parallel-beam projection.
  - **tt_demo_mlem_conebeam** Maximum-likelihood expectation-maximisation (MLEM), cone-beam geometry.
  - **tt_demo_mlem_parallelbeam** Maximum-likelihood expectation-maximisation (MLEM), parallel-beam geometry.
- Registration
  - **reg_demo_nonrigid_SSD** Registration with Sum of Squared Differences (SSD) cost function.
  - **reg_demo_nonrigid_NMI** Registration with Normalised Mutial Information (NMI) cost function.
- Segmentation
  - **seg_demo_manual_priors** Brain image segmentation driven by a statistical atlas.
  - **seg_demo_auto_priors** Atlas-free brain image segmentation.

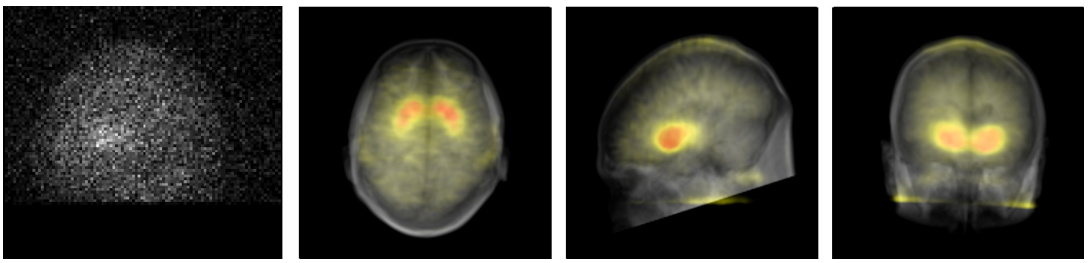


Figure 10.7: MLEM reconstruction of DaTSCAN data acquired on GE Infinia, Low Energy High Resolution (LEHR) parallel-holes collimator. The reconstruction (with depth-dependent resolution modelling) is obtained by executing the NiftyRec demo *et_demo_osem* through the Matlab interface.

10.4 Performance and accuracy

The routines for projection and back-projection are at the core of all reconstruction algorithms of NiftyRec and constitute the most computationally demanding computing task.

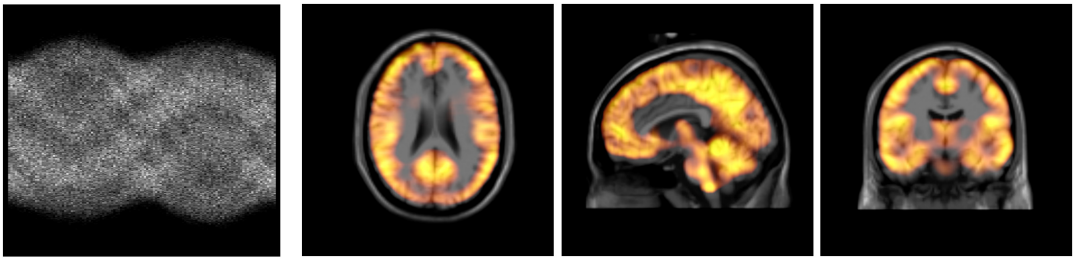


Figure 10.8: Fludeoxyglucose($-^{18}F$) (FDG) PET reconstruction. The uptake is digitally generated and the emission data is simulated with PET Sorteo. The reconstruction (with resolution modelling) is obtained by executing the NiftyRec demo *et_demo_petsorteo* through the Matlab interface.

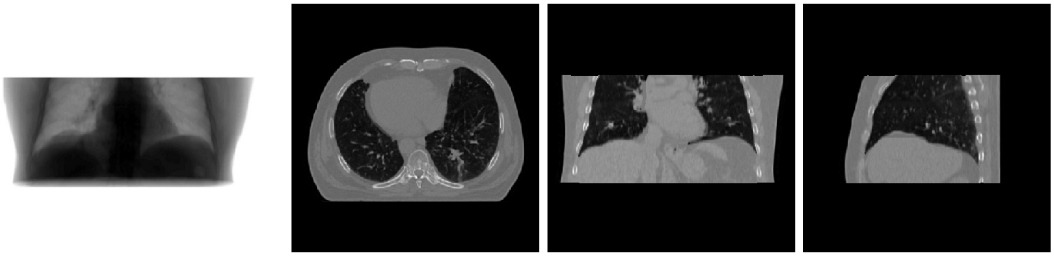


Figure 10.9: Cone-beam X-Ray CT reconstruction. This reconstruction is obtained by executing the NiftyRec demo *tt_demo_mlem_conebeam* through the Matlab interface.

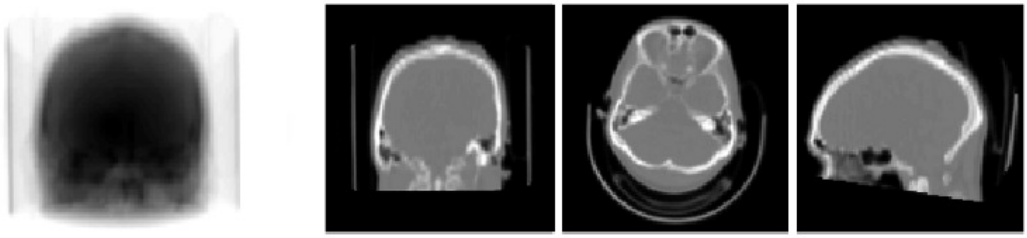


Figure 10.10: Parallel-beam X-Ray CT reconstruction. This reconstruction is obtained by executing the NiftyRec demo *tt_demo_mlem_parallelbeam* through the Matlab interface.

In order to promote interoperability and to enable the evaluation and comparison of the runtime performance and of the accuracy, NiftyRec wraps the projectors and back-projectors for emission imaging of the Image Reconstruction Toolkit (IRT) and of the Software for Tomographic Image Reconstruction (STIR). The wrappers of IRT and STIR mimic the NiftyRec functions *et_project()* and *et_backproject()*, enabling direct comparison and drop-in replacement. The algorithms for projection and back-projection of the three reconstruction softwares are different. IRT implements a voxel-driven projector back-projector pair, where one ray per voxel is cast, rather than one ray per detector bin. STIR implements multiple algorithms for projection and back-projection. The two that are most commonly used are the

detector-driven ray tracer and the detector-driven ray tracer with multiple rays. The first algorithm casts one ray for each detector bin. The second algorithm casts multiple rays from a grid of points displaced on the surface of each detector bin.

The first experiment consisted in projecting a point source placed at the centre of the 3-D imaging volume. The number of cameras was 120. Fig 10.11 displays the central line of the viewgram, providing a visualisation of how the projection of a point source is affected by the different projection algorithms. As one may have expected, the voxel-driven algorithm yields identical projection in all directions.

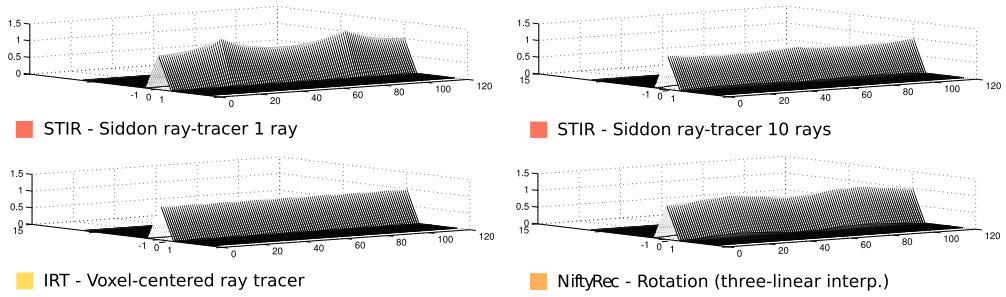


Figure 10.11: Visual comparison of the different interpolation schemes adopted in IRT, STIR and NiftyRec. The images display the central line of the viewgram obtained by projecting a point at the centre of a cubic imaging volume. The detector is displaced in 120 locations from 0 to π radians.

The second experiment consisted in performing the noiseless projection and reconstruction of the a phantom image with each of the three reconstruction softwares. Fig. 10.12 reports the phantom, the reconstructions obtained with 3000 iterations of MLEM and images representing the difference between the reconstructions obtained with the three reconstruction softwares. The plot in Fig. 10.12-bottom reports the mean squared error (MSE) between the phantom and the reconstruction as a function of the number of iterations. Low MSE is indicative of a good match between the projector and the back-projector.

For all three reconstruction softwares, the computing time for projection matches approximately the computing time for the back-projection. Therefore run-time performance was evaluated only for the projection algorithms. Fig. 10.13 reports the timing charts obtained from the projection of 3-D volumes of varying size (i.e. number of voxels). The numbers indicate the timer required for the computation of the projection in 180 directions. The size of the viewgram obtained from the projection of the volume of size $N \times N \times N$ is $N \times N \times 180$. Fig. 10.14 reports figures of the computing time for the projector for transmission imaging. Finally, Fig. 10.15 reports the time necessary for the reconstruction of a SPECT image with NiftyRec.

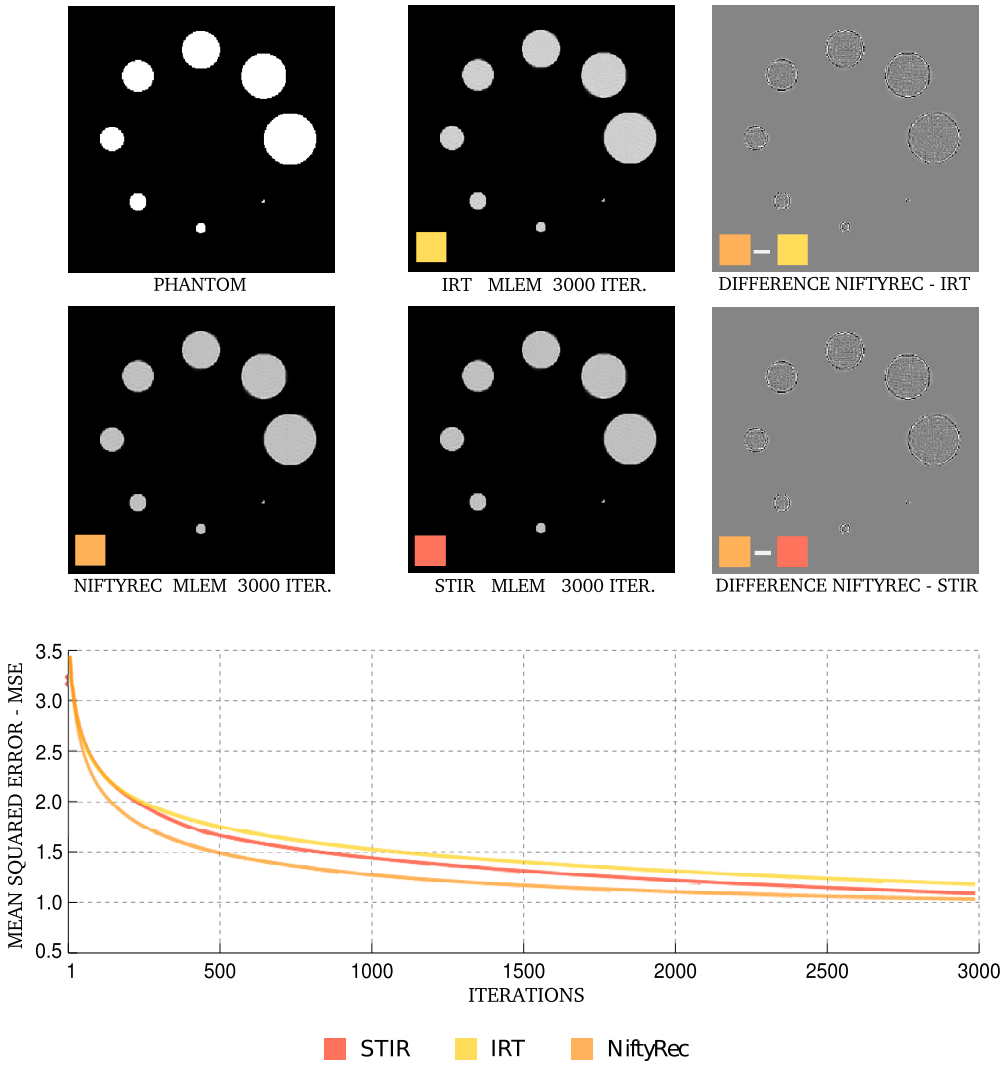


Figure 10.12: Top: Reconstruction of noiseless synthetic data. Bottom: mean squared error (MSE) between the phantom and the image reconstructed with MLEM as a function of the number of iterations. Low MSE is indicative of good match between the projector and the back-projector.

10.5 Distributed computing

NiftyRec exploits the massively-parallel computing architecture of graphics processing units, achieving unprecedented performance. However the complexity of the reconstruction algorithms and the dimensionality of the raw and reconstructed data keep increasing. Four-dimensional tomography, in particular, is extremely demanding computationally.

In order to address the increasing demand of computing resources, reconstruction may be performed on multiple computers interconnected in a distributed computing system. While the two terms *parallel computing* and *distributed computing* have a lot of overlap, and no clear distinction exists between them, let us adopt the definition given in [138]:

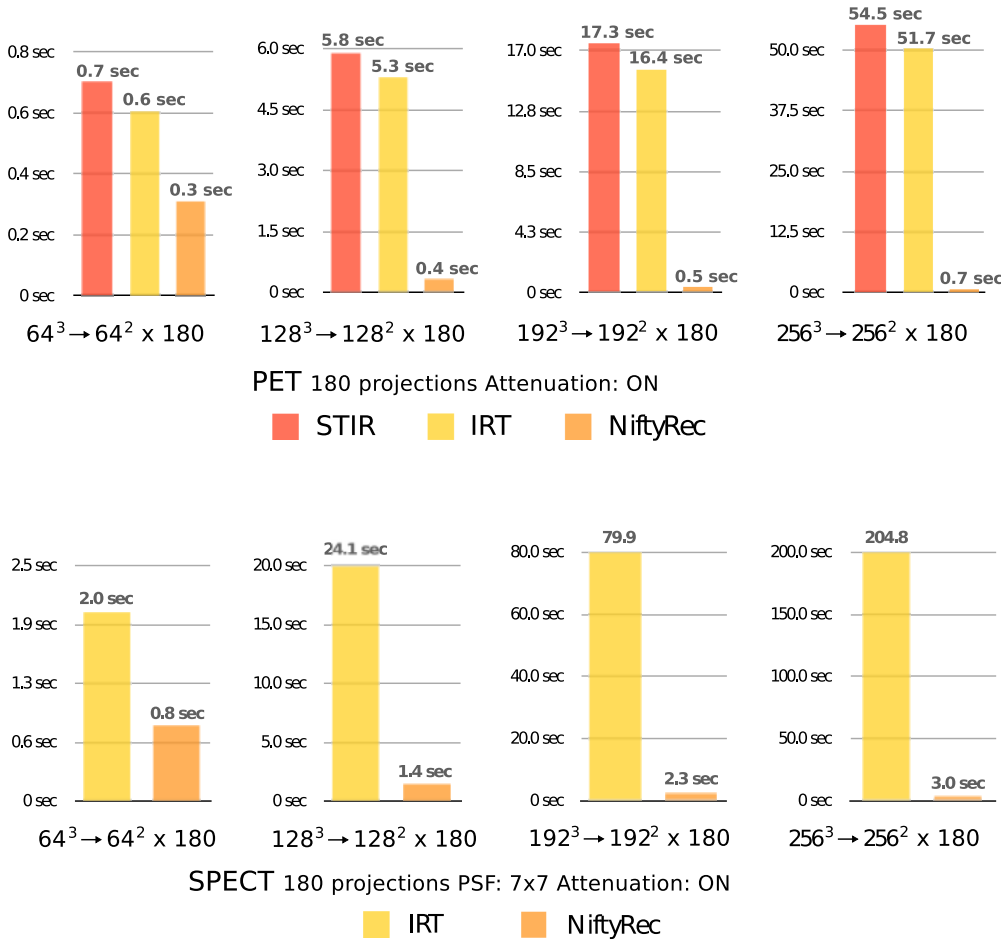


Figure 10.13: Performance chart emission imaging: The performance of the projectors for PET and SPECT is compared with the Image Reconstruction Toolbox (IRT) and with the Software for Tomographic Image Reconstruction (STIR). The charts report the computation time for 180 projections of volumes of size spanning from 64^3 to 256^3 voxels. Both the PET and SPECT projections account for the attenuation map. The SPECT projection accounts for resolution of the system by means of depth-dependent point spread function (PSF). The reported computation times were measured on a desktop computer with CPU Xeon E5430@2.66GHz quad-core, GPU NVidia GTX285 and operating system Ubuntu Linux 64 bit. STIR and IRT are compiled with the OpenMP option enabled and NiftyRec with the ENABLE_CUDA option activated.

- In parallel computing, all processors may have access to a shared memory to exchange information between processors.
- In distributed computing, each processor has its own private memory (distributed memory). Information is exchanged by passing messages between the processors.

GPUs are parallel computing arhitectures. The physical vicinity of memory and the multi-processors enables, for a certain class of computational problems, such as the projections and back-

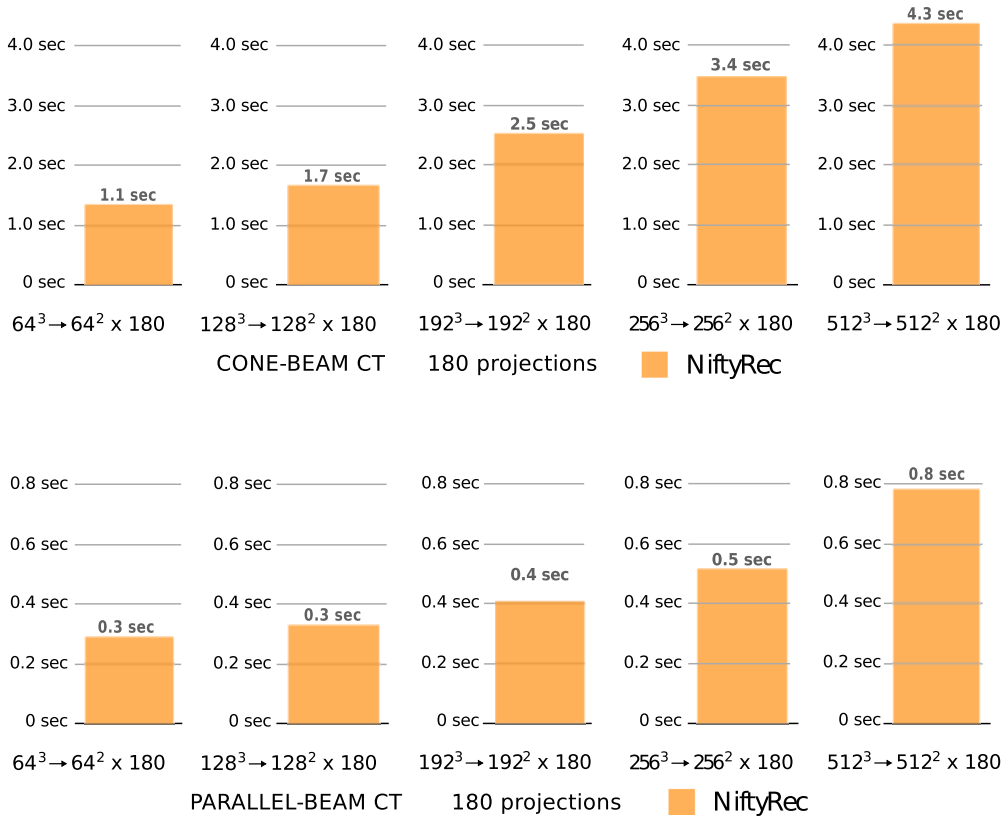


Figure 10.14: Performance chart of the projectors for transmission imaging. The charts report the computation time for 180 projections of volumes of size spanning from 64^3 to 512^3 voxels. The reported computation times were measured on a desktop computer with CPU Xeon E5430@2.66GHz quad-core, GPU NVidia GTX285 and operating system Ubuntu Linux 64 bit. NiftyRec was compiled with the ENABLE_CUDA option activated.

	32^3	64^3	128^3	256^3
Intel Xeon E5430 2.66GHz	30 sec	3 min	15 min	65 min
NVidia GeForce GTX-285	6 sec	8 sec	12 sec	45 sec
Speed-up	5	22	75	87

Figure 10.15: SPECT reconstruction with OSEM iterative algorithm (subset size is 1/8 of the number of detector bins), **180 camera positions**, **60 iterations**. Reconstruction time is reported for several sizes of the imaging volume, with and without GPU acceleration.

projections involved in tomographic reconstruction, performance otherwise not achievable with distributed computing systems. As clarified later in this chapter, in data-intensive tasks such as those involved in computed tomography, communication bandwidth is a bottleneck for distributed comput-

ing systems. The performance achieved by a single GPU, in fact, may supersede the performance of a computer cluster with thousands of nodes. However, the resources available on a single GPU are limited.

The architectures of distributed computing systems vary greatly, including multi-core single computer systems, eventually equipped with multiple GPUs; small networks of heterogeneous computers; homogeneous and heterogeneous clusters and worldwide grids for cloud computing. Multi-threading and multiprocessing technologies allow one to orchestrate the use of multiple CPUs and graphics cards on the same machine, but the number of CPU cores and GPUs is limited. Message passing technologies on the other hand allow one to scale applications to use multiple CPUs in homogeneous and heterogeneous clusters, however there is scarcity of programming models and software support for multi-GPU systems [139][140].

Though it is possible to adapt communication protocols designed for distributed computing [141], such as the Message Passing Interface (MPI), to make use of GPUs, the complexity of the multi-processor based GPU programming model and the additional complexity of the shared memory programming model for the distributed computing system complicate the design and development processes. This work aims at providing a mechanism that enables the distribution of the code that was conceived for execution on single GPU over multiple GPUs installed in a distributed computing system, with little additional effort. Additionally, the software framework described in the following aims at providing an abstraction from the architecture of the distributed computing system and from the communication system, enabling the distribution of tasks on a large range of platforms: multiple CPU cores or GPUs on a single machine, one or multiple CPUs or GPUs per machine in a cluster or in a grid.

The programming model described in the following is an instance of the Boss/Worker model (B/W). The software framework is implemented in Python.

The simple B/W model, described in the following, provides an abstraction over the communication protocol, guaranteeing automatic load balancing and a degree of latency hiding. Hiding the communication protocol, the B/W model allows the programmer to be agnostic about the underlying mechanism that allows the nodes to exchange data and to ignore communication latency hiding and load balancing considerations.
The B/W programming model, in fact, provides an abstraction over the communication protocol and guarantees automatic load balancing and latency hiding for computational problems that present a high degree of task parallelism [142][143]. Such is the case of iterative tomographic reconstruction, as clarified later in this chapter.

The first paragraph that follows describes the B/W programming model and issues specific to GPU embedding; the second paragraph that follows describes the implementation of the back-end based on MPI-2 and finally evaluation of performance and scalability on small computer cluster.

10.5.1 GPU aware Boss/Worker architecture

In order to scale execution of a program that makes use of the B/W abstraction on a number of different architectures, ranging from a single multi-core machine, to a GPU cluster, to a computing grid, the application programming interface (API) of the B/W framework provides an abstraction over the mechanism for data exchange. The back-end for communication and data exchange is an interchangeable module which can be reimplemented in order to enable the same applications to run under different computation environments.

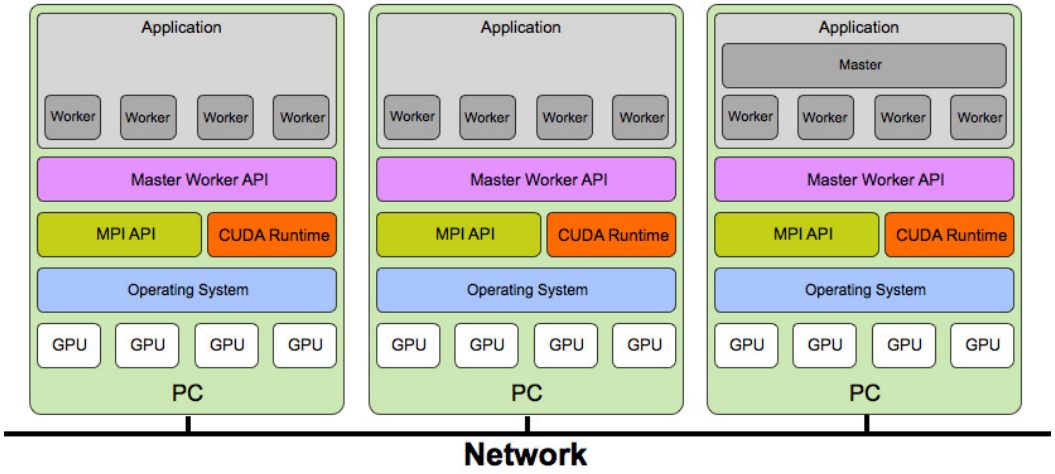


Figure 10.16: Boss/Worker abstraction

In order to maintain a common API, it is necessary to address the issue of selection and allocation of GPU devices in a way that is consistent with the abstraction and that can be implemented over a number of the back-ends. It is straightforward to allocate GPU resources when multiple GPUs run on a single machine, by means of the GPU runtime API, however batch queuing systems, message passing infrastructures and the operating system are commonly unaware of GPUs. As batch queuing systems are not GPU aware, one simple solution for the allocation of resources is to assume that there is a one-to-one matching between nodes and GPUs so that one may rely on the existence of a GPU on each node. However the following addresses heterogeneous clusters where some of the machines have one or more GPUs. The batch queuing system has no awareness of the GPUs so one may let the batch queuing system decide what nodes to allocate (this can be controlled configuring the batch queuing system), then the B/W software implements a handshaking mechanism that makes the Boss aware of the computing resources of each worker. When the application is launched, all the workers attempt to use the GPU API (*cudaGetDeviceCount* and *cudaGetDeviceProperties*) to retrieve the properties of the installed GPUs and communicate to the Boss a list of the GPUs installed on the local machine and their characteristics. The Boss then uses MPI node IDs (process IDs for the OS Pipe back-end) and GPU device number to figure out which nodes might conflict the allocation of a GPU on the same machine and assigns uniquely

one GPU to each Worker; the Worker allocates the device that it has been assigned so each Worker is associated at most to one GPU. In order for all the GPUs to be in use it is necessary to configure the queuing system to create, on each machine, at least one node per GPU.

The Boss assigns a unique *WorkerID* to each Worker and stores information about whether the Worker is GPU capable and what is the type of GPU and its compute capability ($CUDA\_1.1$, $CUDA\_2.0$, $CUDA\_3.0$,$CUDA\_4.0$,$CUDA\_5.0$,...). The programmer defines the Boss and the Worker functionalities by sub-classing two basic Boss and Worker classes. Each method of the Worker class may be executed by the Boss simultaneously on several Workers. A function *list_workers(compute_capability)* allows the Boss to obtain a list of Workers with a specific compute capability; the parameter of this function is a string that allows the Boss to query a list of all Workers that support GPU acceleration, or a list of the Workers that have a compute capability equal, greater or smaller then a specific value (for example =CUDA_2.0 lists all the Workers that have a CUDA device with compute capability exactly equal to 2.0).

The Boss has a *main* method that is executed when the Boss is first started by *M.engage()*. The programmer overrides the *main* method of the Boss to define the execution of tasks by the Workers.

In order to achieve automatic load balancing and to hide communication latency, the B/W implements and asynchronous API. The function *do(task_name, worker_ids, parameters)* determines buffering of function name, parameters and worker_ids and returns immediately a job ID to the Boss. In the background the function name and data objects are sent to the Workers. The second parameter to function *do* may also be a string that specifies compute capability; in this case the task is dispatched to any worker that has the requested compute capability. An idle Worker is chosen if available, otherwise the Worker with least number of tasks awaiting for execution is chosen. This mechanism is identical to the classical *task bag* [144][145] if the compute capability is set to *ANY*, but it allows to route tasks to nodes with GPU acceleration or specific compute capability.

If the Worker is busy, the function name, job ID and parameters are buffered on the Worker side. When the Worker completes the execution of a job, it notifies the Boss and pops a new job from the local buffer. The Boss disposes of two APIs to query the status and result of a job: the asynchronous interface returns *None* if the job (identified by its job ID) has not been completed, it returns the result otherwise; the synchronous interface, that blocks until the result is ready, is more efficient then polling the asynchronous interface when the Boss wants to synchronize with a specific job outcome. A *sync(jobIDs)* function is provided to wait until all jobs with given job IDs are completed or until all jobs are completed if the argument is 0.

The Workers are shut down explicitly by the Boss by means of a *poison pill* when the Boss shuts down. Before terminating, the Worker process deallocates the GPU device.

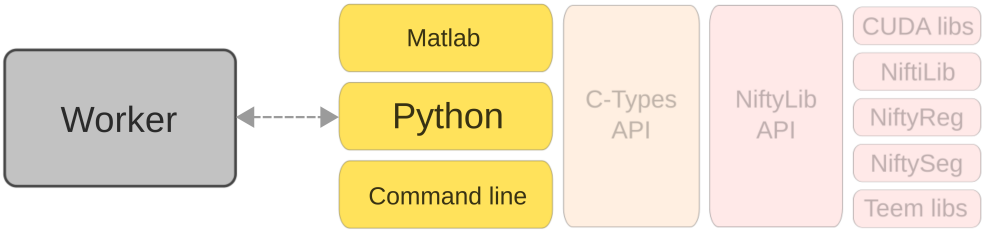


Figure 10.17: Boss/Worker abstraction

10.5.2 MPI Backend

The back-end for communication and data exchange can be implemented with a number of technologies, such as MPI, OS Pipes, Condor, BOINC, Tuple Space, addressing different cluster configurations. This allows us to run the same unmodified software on a single machine, on a cluster, or on a worldwide grid for cloud computing.

The only implementation of the back-end currently implemented is based on MPI-2. MPI and its batch queuing system provide the means to launch multiple processes on the nodes of a cluster and to send and receive messages in a synchronous and an asynchronous manner. The network layer is abstracted by assigning a node ID to each node. Though a more efficient beck-end might be designed to exploit shared memory when communicating amongst multiple processes on a single machine, MPI can create multiple nodes on the same machine and the OpenMPI implementation of MPI addresses the issue of efficient data transfer amongst processes.

In the B/W model, the Boss communicates with the Workers and the Workers send results back to the Boss, however there is no communication between Workers directly; each Worker has a *WorkerID* that is mapped, in the MPI back-end, to a node ID.

In order to implement the task bag with double queue and asynchronous behaviour, we use the asynchronous functions from the MPI API: asynchronous *send* to send messages from the Boss to the Workers and back and *iProbe* and *recv* to asynchronously receive the messages. We use the *tag* of the MPI message to define three message types: *JOB*, *RESULT*, *COMMAND*; the content of the message is always a dictionary, that has specific fields for the three types of messages. The JOB message has a *function* key, a *parameters* key, and a *jobid* key; *function* is a string that specifies the name of the function, *parameters* is a list object that lists all the parameters of the function (each with its own object type) and *jobid* is the integer ID that the Boss assigned to the job. The RESULT message, similarly, has a *result* key (list object) and a *jobid* key (integer). The COMMAND message specifies a *type* key (string object) and a *content* key (list object).

The back-end uses these three message types (tags) to send job requests, results and messages for handshaking and shutdown (*poison pill*).

10.5.3 Performance and Scalability

We present the implementation of an application for Emission Tomographic Reconstruction. Through this application we show how the B/W programming model applies to a genuine problem and we evaluate the benefit that can be obtained by distributing the execution on multiple CPUs and GPUs by means of the B/W framework.

The MLEM algorithm iteratively updates an estimate of the radio-tracer activity, with increasing likelihood of the estimation being the one that determined the observed data [17]:

$$\hat{\lambda}_b^{new} = \hat{\lambda}_b^{old} \frac{1}{\sum_{d=1}^{D} p_{bd}} \sum_{d=1}^{D} \frac{n_d \, p_{b,d}}{\sum_{b'=1}^{B} \hat{\lambda}_{b'}^{old} \, p_{b',d}}, \quad b = 1, .., B \tag{10.1}$$

A GPU accelerated projection/backprojection algorithm has been written in C/CUDA and is described in detail in [97]. The projection/backprojection function, which takes as input $\hat{\lambda}_b^{old}$, a list of cameras (their position and size), a point spread function, input projection data from each camera, and produces the backprojection (right of $\hat{\lambda}_b^{old}$ (10.1)), has been automatically wrapped with the wrapper generator based on SWIG for use with the B/W programming interface.

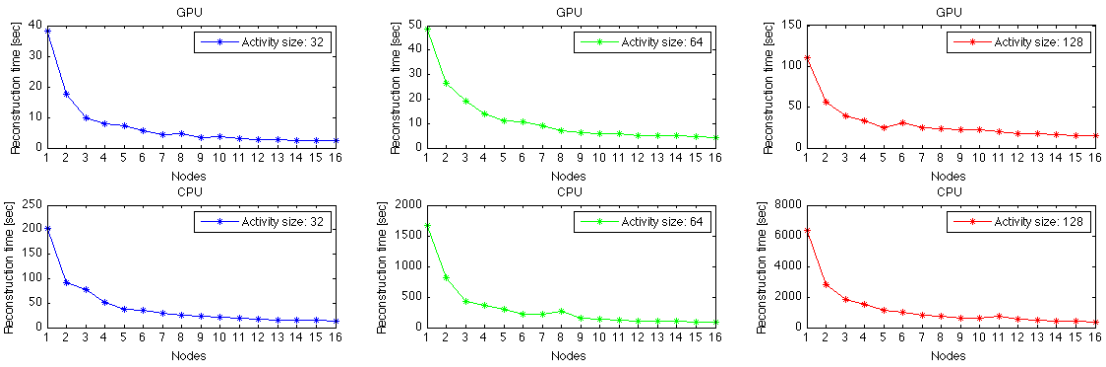


Figure 10.18: Performance of MLEM Tomographic Reconstruction with the Boss/Worker framework for activity of size 32^3,64^3 and 128^3 voxels. Top figures show execution time for reconstruction on multiple GPUs; the figures in the bottom show execution time for reconstruction on multiple CPUs. The given time is for 60 iterations of MLEM and 180 projection angles.

As can be observed from the MLEM formula in (10.1), the projection and backprojection can be calculated for any subgroup of D and the resulting backprojections need to be reduced by a sum operation in order to compute the new estimate of $\hat{\lambda}$. We exploit this task concurrency to execute the projection/backprojection on multiple GPUs buy letting each GPU compute the backprojection from a single camera (there are 180 cameras). The reduce operation is performed by the Boss for simplicity and the synchronous interface allows reduced memory footprint as one projection is loaded from the workers at the time. The Python listing for the MLEM reconstruction is reported below.

```
from PyMW import Boss, Worker, Backend_MPI
from Spect import projector, save_activity, uniform_activity,\
create_psf, create_cameras, create_test_data, subset_cameras
N = 64;
n_cameras = 180;
out_name = 'activity.nii'

class Spect_Worker(Worker)
  def project(activity, psf, cameras):
    return projector.project(activity, psf, cameras)

class Spect_Boss(Boss)
  def reconstruct(self, sinogram, psf, cameras, activity):
    #use only nodes with GPU for projection
    workersIDs = self.interface.list_workers('GPU')
    id_list = []
    for i in range(len(workerIDs)):
      cameras_w = subset_cameras(cameras,len(workerIDs),i)
      #add job to the task bag
      id = self.interface.do('project',workerIDs[i],activity,\
                                                  psf,cameras_w)
      id_list.append(id)
    #reduction:
    for id in id_list:
      activity += self.interface.get_result(id)
    return activity

  def main(self):
    sinogram = create_test_data(N,'brain','poisson')
    psf      = create_psf(N,'ddpsf')
    cameras  = create_cameras(n_cameras,180)
    activity = uniform_activity(N)
    activity = self.reconstruct(sinogram,psf,cameras,activity)
    save_activity(activity,out_name)

if __name__ == "__main__":
  B = Backend_MPI()
  W = Spect_Worker(B)
  M = Spect_Boss(B)
  M.engage()
  W.engage()
```

The code has been tested for reconstruction on a small cluster of 16 computers, each equipped with a CPU Intel Core 2 Duo E6700 and a GPU NVIDIA GTX 285 with 1Gb of DDR3; the PCs are connected by means of 1Gbps Ethernet. Each machine is equipped with Ubuntu-9.4 with NVIDIA drivers 195.36.1 and they share a network drive where we have installed Python-2.5, Numpy-1.3.0 and MPI4Py built with OpenMPI-1.4.1. We launch MPI by means of *ssh*. In figure 10.18 we report the execution time registered for reconstruction until convergence (60 MLEM iterations) with 180 projections and three different sizes of the activity: 32^3, 64^3, 128^3 voxels. The three images on the top represent the reconstruction time for the GPU implementation while the three in the bottom are for the CPU implementation. These images

show the speed up of approximately 100-fold of the GPU implementation over the implementation for CPU, when executed on a single core (no multi-threading support was implemented for the CPU version). When the size of the data is small, the improvement is less marked because of the time it takes to launch the kernels on the GPUs (tough the recently released Fermi devices claim a 10-fold improvement in kernel scheduling time).

Scaling the execution on multiple GPUs gives an immediate benefit, but the gain in performance drops when using more than 8 GPUs. This is due to the latency when transferring the activity to the Workers and back, in fact there is no other task subsequent to projection/backprojection, so after each projection/backprojection task all the workers need to synchronize to perform the reduction and produce the new estimation of the activity. When the algorithm is distributed on 8 GPUs we obtain a speed up of about 4, while the speed up for 8 CPU is of about 5 times. However it is remarkable that it is not possible to achieve the performance of a single GPU with a cluster of CPUs, due to communication latency predominating the computation.

The aim of our work is to provide a Boss/Worker abstraction that allows to distribute code on multiple machines in number of computing environments as diverse as possible. We have reviewed the Boss/Worker programming model showing its applicability and limitations. The B/W abstraction provides an effective way to distribute execution of code when the problem presents a high degree of task concurrency. In this case a B/W programming model is appealing as it provides abstraction from the communication protocol between processes and because it provides automatic load balancing and hides communication latency. We have presented the implementation of a Python based B/W framework that allows to distribute an application, unmodified, on a number of diverse computing environments; particularly we have addressed distribution of code in a multi-GPU environment. We have addressed a number of issues involved in distributing workload to multiple GPUs and shown that the B/W abstraction is well suited to multi-GPU programming.

Finally we have tested our B/W framework by implementing an algorithm for Tomographic Reconstruction in Emission Tomography. This application shows that the framework allows to distribute execution on multiple GPUs with little effort for the programmer. However for this application, the scalability is limited to a maximum of 8 to 16 GPUs because of the low computational intensity of the macroscopic tasks that can be extracted. Nonetheless, a speed up of a factor 4 is achieved by distributing the execution of the algorithm on 6/8 GPUs. Because of the intrinsic low computational intensity of the problem taken into consideration, the reconstruction algorithm does not scale well neither on CPU nor on GPU, but the GPU implementation provides a speed up of up to 100 times over the CPU implementation. The B/W framework introduces another speed up of a factor 4. The B/W programming model proves to simplify the distribution of code to multiple machines and GPUs, and its unified interface provides the means to distribute code on clusters of diverse sizes and configurations. It can be applied to scale a GPU based application to run on a small GPU cluster, or to scale an algorithm that presents sufficient

task parallelism on a bigger cluster.

10.6 A visualisation tool for multi-modal imaging data

Visualisation tools have the crucial role of delivering, for human interpretation, the information acquired by the imaging. The integrated modelling approach shifts the focus of the imaging process from the creation of images to direct extraction of clinically-relevant information and visualisations from the raw data produced by the imaging hardware. In order to realise such imaging paradigm, NiftyRec includes an interactive visualisation tool for multi-modal imaging data, designed especially for the joint visualisation of anatomical and functional information.

The visualisation of tomographic data presents the challenge of extracting 2-D visualisations for the human eye from the 3-D tomographic data. The two common approaches to the visualisation of tomographic data are *slicing* and *volume rendering*. In the first case the volume is subdivided in a series of 2-D slices and the visualisation software typically provides an interface for the selection of the slice of interest and for navigation of the slices. Conversion from the scalar value in each voxel of the tomogram to the colour of the slice is typically obtained by the use of a re-configurable look-up table denominated as colour-map. Volume rendering assigns a virtual source of light and a virtual coefficient of attenuation to every voxel of the tomogram. The image displayed on the screen in the image observed by a virtual camera that collects the light produced by the virtual light sources.

In nuclear imaging, PET and SPECT functional images are often visualised in combination with an anatomical image, typically X-Ray CT or MRI, in order to aid the localisation of regions and of anatomical structures. The joint visualisation is typically obtained by *slicing* and by *fusing* the two slices from the two modalities. There are a number of algorithms for image fusion; typically the anatomical image is rendered with a black-and-white colour-map and fused, by alpha compositing (see `http://en.wikipedia.org/wiki/Alpha_compositing`), with the functional image, rendered with colour-encoded intensity. Slicing presents the problem that the observer visualises, at a given time, only the information contained in a single slice. The process of joining the information from neighbouring slices requires some effort of memory for the observer. Volume rendering produces a global visualisation. While volume rendering is flexible, enabling even the visualisation of surfaces (see figure 10.21), the global visualisation may be overwhelming.

The visualisation tool of NiftyRec is based on ray-casting and merges the two methods of *slicing* and *volume-rendering*.

10.6.1 Volume rendering algorithm

The multi-modal visualisation tool of NiftyRec is based on a modified version of the GPU-accelerated ray-casting algorithm utilised for the cone-beam projector 10.2.2. The modified ray-casting algorithm for visualisation is depicted in Fig. 10.19. Rays are traced from the source to the screen (virtual camera). Integration is performed only within a rectangular bounding box (represented in grey in Fig. 10.19)

parallel to the camera plane. The depth and thickness of the bounding box can be modified in real time. Multiple volumetric images are defined (depicted in red and in yellow in the figure) and each volume is associated to a colour-map. If the bounding box covers entirely the images, then the visualisation is a multi-channel volume rendering, if the bounding box is 1-voxel-thick, then the image on the screen is the alpha composition of the slices of the volumes corresponding to the location of the box.

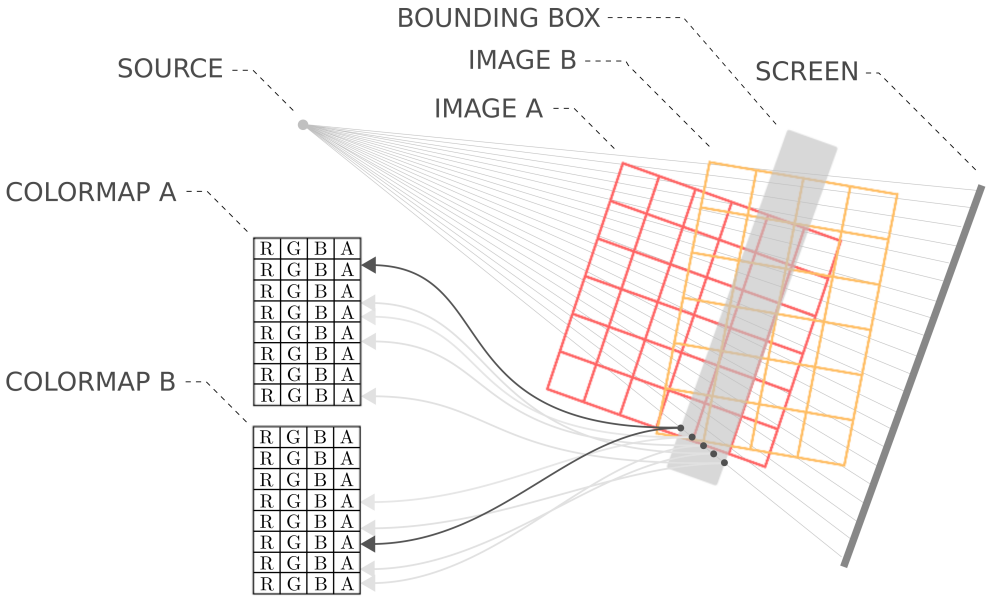


Figure 10.19: Volume renderer for multi-modal datasets.

The colour-map maps the scalar values of the image to a colour value and opacity: RGBA (for red, green, blue, alpha). The rendering algorithm computes, for each ray, the coordinates of the intersection with the bounding box, then the ray is sampled at regular intervals throughout the bounding box. The data is interpolated at each sample point, the colour-map applied to form an RGBA sample and the sample is composited onto the accumulated RGBA of the ray. The process is repeated until the ray exits the volume. The RGBA color is converted to an RGB color and deposited in the corresponding image pixel.

Integration rule

Two algorithms for the computation of the sequence of alpha compositions have been implemented in NiftyRec 1.6:

- Trapezoid rule (see `http://en.wikipedia.org/wiki/Trapezoid_rule`).
- NewtonCotes order 3 integration formula (see `http://en.wikipedia.org/wiki/Newton-Cotes_formulas`).

Fig. 10.20 displays the volume rendered image of a T1-weighted MR image. The image is rendered with black-white colour-map and with bounding box covering the entire image. Fig. 10.20-left os obtained

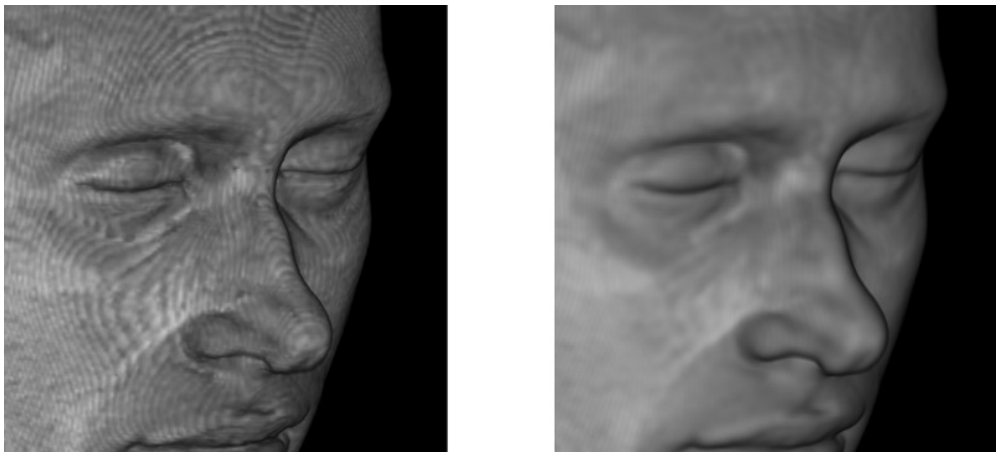


Figure 10.20: Volume renderings of T1-weighted MR image. Left: Trapezoid rule integration rule. Right: NewtonCotes order 3 integration rule.

with the trapezoid rule integration scheme, while Fig. 10.20-right with the higher order integration scheme (NewtonCotes order 3).

10.6.2 Live interaction

The position of the camera and of the source can be manipulated in real time with the mouse buttons and scroll button. The visualisation tool can also be scripted in real-time with the Python scripting language to load the imaging data, modify the bounding box, set the colour-map and create animations. The list of the methods for the interaction with the volume rendering tool (NiftyRec 1.6) is reported in 10.4. 10.21 reports images obtained with the NiftyRec visualisation tool. Animations can be found on the NiftyRec website 10.8.

Table 10.4: NiftyRec, visualisation API

set_volume(volume,channel) Set the image volume for channel **channel**
set_colormap(colormap,channel) Set the colormap table for channel **channel**
set_scale(scale,channel) The image at channel **channel** is pre-multiplied by this value.
set_offset(offset,channel) The image at channel **channel** is pre-summed to this value.
set_density(density,channel) The alpha values for channel **channel** is multiplied by this values.
set_brightness(brightness,channel) The RGB values for channel **channel** are multiplied by this values.
set_bounding_box(bbox) Coordinates of the bounding-box in voxels.
set_ray_step(voxels) Length of the sampling interval along the ray.
pan_camera(x,y,z) Pan the camera.
rotate_camera(s,t,q) Rotate the camera.
zoom_camera(zoom_value) Zoom in (positive value) and out (negative value).
dump_screenshot(filename) Save an image of the current visualisation. The extension determines the file format.

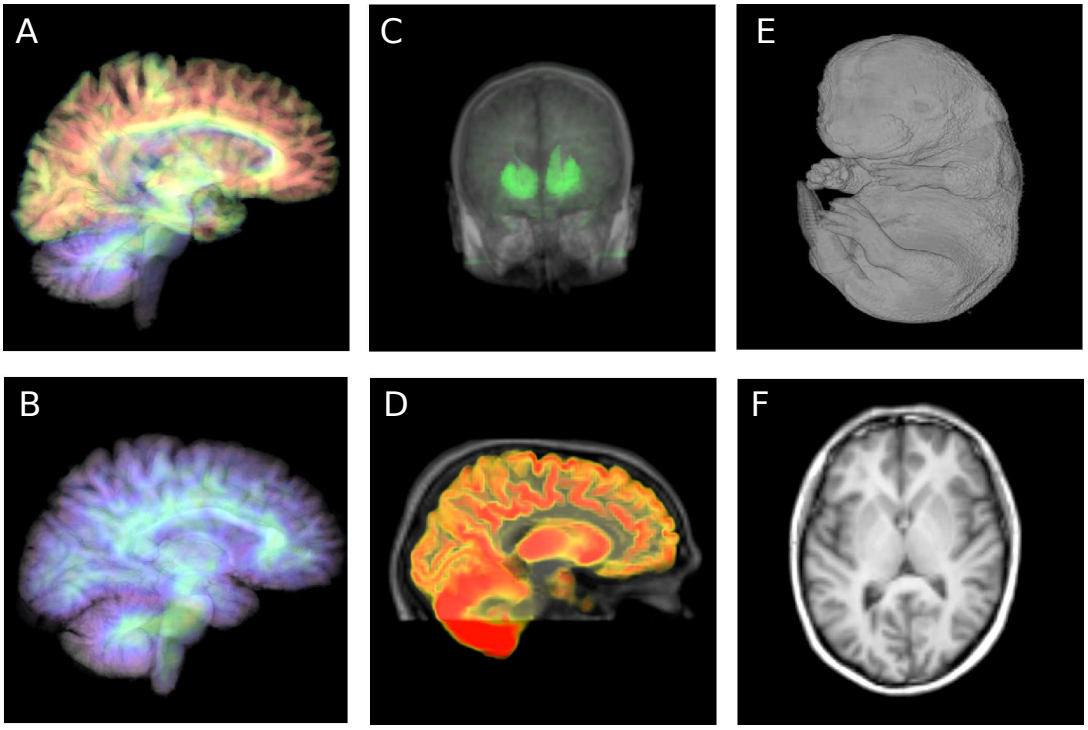


Figure 10.21: Renderings of multi-modal imaging data. (A) PIB PET, control patient; 4-voxels-thick bounding box. (B) PIB PET Alzheimer's patient; 4-voxels-thick bounding box. (C) DaTSCAN SPECT; 128-voxels-thick bounding box. (D) FDG PET; 4-voxels-thick bounding box. (E) T1-weighted MR of mouse embryo; 256-voxels-thick bounding box. (F) T1-weighted MR of human brain; 4-voxels-thick slice.

10.7 BSD license

NiftyRec is distributed under the permissive BSD license, allowing redistribution and use in source and binary forms, with or without modification, provided that the following conditions are met:

- Redistributions of source code or binaries must retain all the copyright notices, the list of conditions and a disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the organization nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

10.8 NiftyRec online

NiftyRec source, binaries and documenation are hosted at the following URL: `http://niftyrec.scienceontheweb.net/`

Chapter 11

Conclusion

11.1 Contributions

This work was inspired by the new possibilities offered by PET-MR imaging systems. The unified modelling approach, driven by the necessity to make joint use of the information from the two imaging modalities, however, has resulted in new models and approaches that find application in both the stand-alone PET and SPECT modalities and in the hybrid PET-MR systems. The main contributions of the work described in this thesis are the following:

- Formulation of the tomographic reconstruction problem in the probabilistic graphical framework (Chapter 2 and 3).
- Formulation of Bayesian finite mixture model for the integration of information from multiple imaging modalities (anatomical and functional) and fully automated parameter learning algorithm (Chapter 4).
- Formulation of a unifying probabilistic model for 4-D computed tomography, including the use of multi-modal information (Chapter 5).
- Formulation of a semi-parametric model for multi-modal imaging (Chapter 6).
- Formulation of an EM-based algorithm for the estimation of the characteristics of depth-of-interaction (DOI) detectors (Chapter 7).
- Formulation of a globally convergent learning algorithm based on manifold learning techniques for the estimation of the characteristics of depth-of-interaction (DOI) detectors (Chapter 7).
- Formulation of a manifold learning algorithm based on the density of the locally linear embedding (DLLE) (Chapter 7).
- Formulation of a method, based on a sparsified Fisher Information Matrix, for the efficient estimation of the uncertainty in emission computed tomography (Chapter 9).

- Formulation of an active learning method for real-time adaptation of emission imaging devices (Chapter 9).
- Formulation of a method for the compensation of disrupted detectors in emission computed tomography (Chapter 8).
- Formulation of a unifying computational model for scatter compensation in tomographic imaging systems (Chapter 8).
- Development of the open-source GPU-accelerated multi-modal tomographic reconstruction software NiftyRec (Chapter 10).

11.2 Conclusion

The first two chapters (chapter 2 and chapter 3) of this thesis are introductory. Chapter 2 introduced the mathematical formalism of probabilistic graphical models that was adopted to frame the (multi-modal and time-dependent) tomographic reconstruction problem and to derive various models and algorithms throughout the rest of the document. Chapter 3 derived the basic iterative algorithms commonly applied for inference in single modality static tomographic imaging, including MLEM, OSEM and OSL-MAPEM. Such derivations constitute the building blocks for the successive chapters. The second part of the paragraph described the fundamental challenge of tomographic imaging and the link between ill-posedness of the reconstruction problem, resolution, and the partial-volume-effect.

Chapter 4 described a Bayesian model that enables the use of information from multiple imaging modalities including emission computed tomography. With the growing field of PET-MR scanners, the approach and algorithms described in this chapter can have an impact on the quality of the reconstructed PET images. The chapter describes a mixture model and extensions aimed at guaranteeing full automation of the algorithm. Robust estimation is achieved by 1) adopting a population-based statistical atlas to initialise and drive the optimisation of the parameters of the pharmaceutical uptake model and of the MR acquisition system model; 2) adding contextual information in the form of a Markov Random Field over the hidden tissue labels; 3) capturing the MR image bias field. The algorithm has been evaluated in a synthetic study, finding that it improves the quantification of the pharmaceutical uptake when the simulated data reflects the assumptions of the model and, remarkably, that the uptake estimate may improve also in regions that do not obey to the model. Validation with real data remains an open problem as it would require large sets of imaging data, possibly labelled with long term clinical outcome for specific diagnostic tasks. To this extent, automation of the reconstruction algorithm is crucial. Given the wide acceptance and increasing use of of probabilistic atlas based segmentation algorithm [70] [5], the joint generative model for PET-MR might prove useful when coupled with population based spatially varying priors of the hidden states and hyper-priors for the parameters of the mixture model in order to include experience in the reconstruction process.

Chapter 5 extended the model of Chapter 4, accounting for spatial deformations. The model describes the time-dependent interaction between data from multiple imaging modalities and has several applications, unifying some of the approaches for motion correction that have been proposed in the literature and introducing new algorithms. The algorithms introduced in this Chapter have been evaluated with synthetic data, proving feasibility. It is difficult, however, to envisage at this stage implementation strategies on real imaging systems such as defining the most appropriate MRI sequences and frequency of the frames. The work described in Chapter 5 will be further investigated in a research project financed by EPSRC grant *Exploiting the unique kinetic capabilities of simultaneous whole-body PET/MRI*.

Chapter 6 extended the model described in Chapter 4, replacing the parametric model of the pharmaceutical uptake described in 4 with a non-parametric model. The resulting semi-parametric model merges the gap between the parametric and the non-parametric approaches described and utilised in the context of medical image registration.

Chapters 7 and 8 explored the new applications that emerge from the combination of a probabilistic model of the gamma imaging device and a probabilistic model of the tomographic imaging system. The first part of Chapter 7 formulated the probabilistic model of the gamma imaging device in the probabilistic graphical framework. While probabilistic modelling of the gamma imaging device is not a novelty of this paper, the second part of the chapter and chapter 8 extended the probabilistic model of the imaging device in two ways: 1) by defining the inference problem associated to the estimation of the characteristics of the gamma imaging device; 2) by coupling the model of the gamma imaging device with the probabilistic model of the tomographic system. The first extension of the probabilistic model of the imaging device resulted in two algorithms for the estimation of the characteristics of gamma imaging devices, especially for depth-of-interaction (DOI) resolving devices, but applicable to any gamma imaging device. The first algorithm emerges as the trivial solution to the inference problem associated to the characteristics of the imaging device and is based on the EM iterative algorithm. Such algorithm is locally convergent. The second algorithm is based on concepts of manifold learning (introducing a new methodology denominated *density of the locally linear embedding* - DLLE) and has global convergence properties. The learning algorithms described in this chapter require further experimentation with real imaging devices, however the results obtained with simulation studies are encouraging. The recent widespread interest in DOI detectors for high resolution PET imaging confirms the importance of the algorithms described in this chapter.

The second extension of the probabilistic model of the imaging device was explored in Chapter 8 and consisted in coupling the model of the imaging device with the model of the tomographic imaging system. Coupling the probabilistic models couples the two inverse problems that arise from the two models: the estimation of the location of interaction and energy of the gamma photons and the estimation of the activity. The formulation in the graphical framework is fundamental in order to attain the joint model described in this chapter.

The combination of the two models constitutes a novelty and highlights the important observation that the the Cramer-Rao bound does not constitute a hard limit on the accuracy (intended as degree of closeness to the true values) of the location of interaction and energy of a photon. By constructing a model of the process that emits the photons from multiple photon interactions, one may obtain estimates of the location of interaction and energy that are, in average, closer to their true values, than indicated by the Cramer-Rao lower bound associated to the single photon detection. An algorithm for the iterative refinement of a model of the underlying emission process has been described in this thesis. Furthermore, the combination of the two models may have a number of applications. Two were suggested in Chapter 8: 1) an algorithm for the compensation of disrupted photo-detectors; 2) an algorithm for scatter compensation that makes use of continuous energy measurements at low computational cost. Such algorithm may possibly find application in the next generation energy-resolving PET scanners.

All the algorithms described in this thesis work were derived within the Bayesian framework. However, inference was mainly intended as point estimation, as a decision, rather than characterisation of the full posterior probability distributions. Chapter 9 scratched the surface of the fully Bayesian approach, focusing on the problem of the choice of the design parameters of a tomographic imaging system and framing the problem in terms of information retrieval. The problem of optimum design has not been previously treated in the information theoretic framework, perhaps due to the tremendous computational complexity that arises. The second part of the paragraph described a methodology for the efficient computation of the information, based on the sparsification of the Fisher Information Matrix. Such approximation enables the exploration of design spaces otherwise forbidden. The last paragraph of Chapter 9 described the concept of a SPECT imaging system that adapts in real-time to the imaging conditions, selecting the acquisition parameters that maximise the information gain. Though more in-depth evaluation of the adaptive system and experiments with real world imaging hardware are not presented in this thesis, the active-learning SPECT system concept is an interesting application of the sparse FIM algorithm. In-depth evaluation of the effect of the approximation involved in sparsifying the FIM and the application of the method to the optimisation of the parameters of a parallel-hole collimator and of a commercial SPECT system (D-SPECT) are presented in a journal article at this time in the final stage of the revision process [21].

The work described in this thesis is nearly entirely based on the NiftyRec reconstruction software, described in Chapter 10. NiftyRec is freely available, fast, it supports multiple tomographic imaging modalities (including PET, SPECT, cone-beam and parallel-beam transmission tomography) and it is entirely scriptable in Matlab and Python. Finally, the design of NiftyRec is inspired on the graphical models described in this thesis and meant to enable the development of algorithms that integrate tightly reconstruction, registration (e.g. motion compensation) and segmentation.

References

[1] D. Townsend, Combined PET/CT: the historical perspective, Semin Ultrasound CT MR 29 (4) (2008) 232–235.

[2] Inventions of the year 2000, Time Magazine.

[3] S. Cherry, Multimodality imaging: Beyond PET/CT and SPECT/CT, Semin. Nucl. Med. 39 (5) (2009) 348–353.

[4] T. Hany, H. Steinert, G. Goerres, A. Buck, G. V. Schulthess, PET diagnostic accuracy: improvement with in-line PET-CT system: initial results, Radiology 225 (2) (2002) 575–581.

[5] J. Ashburner, K. Friston, Unified segmentation, Neuroimage 26 (3) (2005) 839–51. doi:10.1016/j.neuroimage.2005.02.018.

[6] J.Pearl, Causality: Models, Reasoning, and Inference, Cambridge University Press, 2000.

[7] P. Spirtes, C. Glymour, R. Scheines, Causation, Prediction and Search, MIT Press, Boston, 2000.

[8] R.Scheines, An Introduction to Causal Inference. Is Causality in Crisis?, V. McKim and S. Turner (eds.), University of Notre Dame Press, 1997.

[9] E. Jaynes, Information theory and statistical mechanics, Statistical Physics (1963) 181–218.

[10] H. Kiiveri, T. Speed, Structural analysis of multivariate data: A review., Sociological Methodology, Leinhardt, S. (ed). Jossey-Bass, San Francisco.

[11] D. Koller, N. Friedman, Probabilistic Graphical Models, Principles and Techniques., The MIT Press, Cambridge, Massachusetts, 2009.

[12] A. D. et al., Maximum Likelihood from Incomplete Data via the EM Algorithm, Journal of the Royal Statistical Society. Series B (Methodological) 39 (1) (1977) 1–38.

[13] C. Bishop, Pattern Recognition and Machine Learning, Springer Science and Business Media, 2009.

[14] J. Besag, On the Statistical Analysis of Dirty Pictures, J. of the Royal Stat. Soc. 48 (3) (1986) 259–302.

[15] S. Geman, D. Geman, Stochastic relaxation, gibbs distributions, and the bayesian restoration of images., IEEE Trans. on PAMI. 6 (6) (1984) 721–741.

[16] G.L.Brownell, A history of positron imaging, Physics Research Laboratory, Massachusetts General Hospital, MIT.

[17] L. Shepp, Y. Vardi, Maximum likelihood reconstruction for emission tomography, IEEE Transactions on Medical Imaging 1 (2) (1982) 113–122.

[18] M. D. et al., Truncated Hilbert transform and image reconstruction from limited tomographic data., Inv. Prob. 1 (22) (2006) 1037–1053.

[19] D. Van De Sompel, M. Brady, Robust Joint Entropy Regularization of Limited View Transmission Tomography Using Gaussian Approximations to the Joint Histogram, Information Processing in Medical Imaging 21 (2009) 638–50.

[20] Q.Xu, X.Mou, G.Wang, J.Sieren, E.A.Hoffman, Y.Hengyong, Statistical interior tomography, IEEE Transactio on Medical Imaging 30 (5) (2011) 1116–1128.

[21] N. S.Pedemonte, S.Arridge, S.Ourselin, B.F.Hutton, Efcient Determination of the Uncertainty for the Optimization of SPECT System Design : A Subsampled Fisher Information Matrix, IEEE Trans. Med. Imaging (under revision).

[22] MIT Nuclear Science and Engineering Reading Room, http://mightylib.mit.edu/Course Materials/22.01/Fall 2001/photons part 1.pdf.
URL http://mightylib.mit.edu/CourseMaterials/22.01/Fall2001/photonspart1.pdf

[23] J. H. Hubbell, Photon cross sections, attenuation coefficients, and energy absorption coefficients from 10 kev to 100 gev, Tech. rep., DTIC Document (1969).

[24] J. Hubbell, Review of photon interaction cross section data in the medical and biological context, Physics in medicine and biology 44 (1) (1999) R1.

[25] B. K. Chatterjee, L. LaJohn, S. Roy, Investigations on compton scattering: New directions, Radiation Physics and Chemistry 75 (12) (2006) 2165–2173.

[26] F. Beekman, H. D. Jong, S. V. Geloven, Efficient fully 3-D iterative SPECT reconstruction with Monte Carlo-based scatter compensation, IEEE Trans. Med. Imaging 21 (8) (2002) 867–877.

[27] M. Ljungberg, The SIMIND Monte Carlo program (1997-2013).
URL `www2.msf.lu.se/simind/simind.pdf`

[28] The Poisson distribution, The Poisson distribution.
URL `http://www.pp.rhul.ac.uk/~cowan/stat/notes/PoissonNote.pdf`

[29] H. Hudson, R. Larkin, Accelerated image reconstruction using ordered subsets of projection data, IEEE Transactions on Medical Imaging 13 (4) (1994) 601–609.

[30] P. Green, Bayesian Reconstructions From Emission Tomography Data Using a Modified EM Algorithm, IEEE Transactions on Medical Imaging 9 (1) (1990) 84–93.

[31] S.Ramani, J.A.Fessler, A splitting-based iterative algorithm for accelerated statistical x-ray ct reconstruction, IEEE Trans. Med. Imaging 31 (3) (2012) 677–688.

[32] J.Besag, Spatial interaction and the statistical analysis of lattice systems, Journal of the Royal Statistical Society. Series B (Methodological) (1974) 192–236.

[33] J. et al., Spatial Resolution Properties of Penalized-Likelihood Image Reconstruction: Space-Invariant Tomographs, IEEE Trans. Imag. Proc. 9 (5) (1996) 1346–1358.

[34] J. et al., Mean and Variance of Implicitly Defined Biased Estimators (Such as Penalized Maximum Likelihood): Applications to Tomography, IEEE Trans. Imag. Proc. 5 (3) (1996) 493–506.

[35] J.W.Stayman, J.A.Fessler, Regularization for Uniform Spatial Resolution Properties in Penalized-Likelihood Image Reconstruction, IEEE Trans. Med. Imag. 19 (6) (2000) 601–616.

[36] J.W.Stayman, J.A.Fessler, Compensation for Nonuniform Resolution Using Penalized-Likelihood Reconstruction in Space-Variant Imaging Systems, IEEE Trans. Med. Imag. 23 (3) (2004) 269–285.

[37] J.W.Stayman, J.A.Fessler, Efcient Calculation of Resolution and Covariance for Penalized-Likelihood Reconstruction in Fully 3-D SPECT, IEEE Trans. Med. Imag. 23 (12) (2004) 1543–1556.

[38] J.Nuyts, Small animal imaging with multi-pinholeSPECT, Methods 48 (2) (2009) 83–91.

[39] J.Nuyts, J.A.Fessler, A penalized-likelihood image reconstruction method for emission tomography, compared to post-smoothed maximum-likelihood with matched spatial resolution, IEEE Trans. Med. Imag. 22 (2003) 1042–52.

[40] K.Vunckx, D.Beque, M.Defrise, J.Nuyts, Single and Multipinhole Collimator Design Evaluation Method for Small Animal SPECT, IEEE Trans. Med. Imag. 27 (1) (2008) 36–46.

[41] K.Vunckx, P.Suetens, J.Nuyts, Effect of overlapping projections on reconstruction image quality in multipinhole SPECT, IEEE Trans. Med. Imag. 27 (7) (2008) 972–983.

[42] K.Vunckx, L.Zhou, S.Matej, M.Defrise, J.Nuyts, Fisher information-based evaluation of image quality for time-of-flight PET, IEEE Trans. Med. Imag. 29 (2) (2010) 311–321.

[43] L.Zhou, M.Defrise, K.Vunckx, J.Nuyts, Comparison Between Parallel Hole and Rotating Slat Collimation: Analytical Noise Propagation Models, IEEE Trans. Med. Imag. 29 (12) (2010) 2028–2052.

[44] J.Qi, R.M.Leahy, Resolution and Noise properties of MAP reconstruction for fully 3-D PET, IEEE Trans. Image Proc. 1 (5) (2000) 1346–1358.

[45] H.H.Barrett, J. Denny, R.F.Wagner, K.J.Myers, Objective Assessment of Image Quality. II. Fisher Information, Fourier Crosstalk, and Figures of merit for Task Performance., J.Opt. Soc. Am. 29 (A) (1995) 834–852.

[46] H. et al., Noise properties of the EM algorithm: I. Theory., Phys. Med. Biol. 39 (5) (1994) 833–846.

[47] D. et al., Noise properties of the EM algorithm: II. Monte Carlo simulations, Phys. Med. Biol. 39 (5) (1994) 847–872.

[48] L. et al., A modified uniform Cramer-Rao bound for multiple pinhole aperture design, IEEE Trans. Med. Imag. 23 (7) (2004) 896–902.

[49] L. et al., A vector Uniform Cramer-Rao Bound for SPECt System Design, IEEE Trans. Nucl. Sci. 56 (1).

[50] Q. et al., Accurate Estimation of the Fisher Information Matrix for the PET Image Reconstruction Problem., IEEE Trans. Med. Imag. 23 (9) (2004) 1057–1062.

[51] H. et al., Objective assessment of image quality. III. ROC metrics, ideal observers, and likelihood generating functions., J. Opt. Soc. Amer. A Opt. Image Sci. 1 (15) (1998) 1520–1535.

[52] H. et al., Objective assessment of image quality. IV. Application to adaptive optics., J. Opt. Soc. Amer. A Opt. Image Sci. 1 (23) (2006) 3080–3105.

[53] P.Khurd, G.Gindi, Decision strategies that maximize the area under the LROC curve, IEEE Trans. Med. Imag. 1 (24) (2005) 1626–1636.

[54] H. et al., A comparison of human and model observers in multislice LROC studies, IEEE Trans. Med. Imag. 1 (24) (2005) 160–169.

[55] K. et al., Comparison of channelized hotelling and human observers in determining optimum OS-EM reconstruction parameters for myocardial SPECT, IEEE Trans. Nucl. Sci. 53 (3) (2006) 1200–1204.

[56] A.Yendiki, J.A.Fessler, Analysis of observer performance in known-location tasks for tomographic image reconstruction, IEEE Trans. Med. Imag. 1 (25) (2006) 28–41.

[57] Wikipedia: Toeplitz matrix (June 2013).
URL `http://en.wikipedia.org/wiki/Toeplitz_matrix`

[58] D. Wilson, B. Tsui, Spatial Resolution Properties of FB and ML-EM Reconstruction Methods, Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC) (1993) 1189–1193.

[59] A. Reilhac, C. Lartizien, N. Costes, S. Sans, C. Comtat, R. Gunn, A. Evans, PET-SORTEO: a Monte Carlo-based Simulator with high count rate capabilities., IEEE Trans. on Nuc. Sci. 51 (2004) 46–52.

[60] J. Fessler, Image Reconstruction Toolkit.
URL `http://web.eecs.umich.edu/~fessler/code/`

[61] W. V. et al., Numerical Recipes in C - Second Edition, Cambridge University Press, 2002.

[62] Wikipedia: Multinomial Theorem (Feb 2014).
URL `http://en.wikipedia.org/wiki/Multinomial_theorem`

[63] A. Atre, K. Vunckx, K. Baete, A. Reilhac, J. Nuyts, Evaluation of different MRI-based anatomical priors for PET brain imaging, IEEE Nuclear Science Symposium Conference Record (2009) 1–7.

[64] R. Leahy, X. Yan, Incorporation of Anatomical MR Data for Improved Functional Imaging with PET, Information Processing in Medical Imaging (1991) 105–120.

[65] S. Somayajula, A. Rangarajan, R. Leahy, PET image reconstruction using anatomical information through mutual information based priors: a scale space approach, International Symposium on Biomedical Imaging (2007) 165–168doi:10.1109/ISBI.2007.356814.

[66] A.Venkataraman, Y.Rathi, M.Kubicki, C.F.Westin, P.Golland, Joint Generative Model for fMRI/DWI and Its Application to Population Studies. , MICCAI 2010 (2010) 191–199 (Part I).

[67] P.Hiltunen, S.J.D.Prince, S.Arridge, A combined reconstruction-classification method for diffuse optical tomography. , Phys. in Med. and Biol. 1 (54) (2009) 6457–6476.

[68] S. Sastry, R. Carson, Multimodality Bayesian algorithm for image reconstruction in positron emission tomography: a tissue composition model., IEEE Transactions on Medical Imaging 16 (6) (1997) 750–61. doi:10.1109/42.650872.

[69] A. Rangarajan, I. Hsiao, G. Gindi, A Bayesian joint mixture framework for the integration of anatomical information in functional image reconstruction, Journal of Mathematical Imaging and Vision 12 (3) (2000) 199–217.

[70] K.V.Leemput, F.Maes, D.Vandermeulen, P.Suetens, Automated model-based tissue classification of MR images of the brain, IEEE Transactions on Medical Imaging 18 (10) (1999) 897–908. doi:10.1109/42.811270.

[71] K.V.Leemput, F.Maes, D.Vandermeulen, P.Suetens, 4-D generative model for PET/MRI reconstruction, IEEE Trans. on Med. Imag. 18 (10) (1999) 885–896.

[72] MNI, MNI-152 statistical brain atlas. http://fsl.fmrib.ox.ac.uk/fsl/fslwiki/Atlases.
URL http://fsl.fmrib.ox.ac.uk/fsl/fslwiki/Atlases

[73] B. Menze, K.V.Leemput, D.Lashkari, M.A.Weber, N.Ayache, P.Golland, A generative model for brain tumor segmentation in multi-modal images, MICCAI II (2009) 151–159.

[74] Siemens Somatom Definition Flash.
URL http://www.siemens.com/dualsource

[75] Toshiba Aquilion One.
URL http://www.aquilionvision.com/tmd/vision/

[76] R.R.Fulton, S.R.Meikle, S.Eberl, J.Pfeiffer, C.J.Constable, Correction for head movements in positron emission tomography using an optical motion-tracking system, IEEE Trans. Nucl. Sci. 49 (1) (2002) 116–123.

[77] R. Fulton, B. Hutton, M. Braun, B. Ardekani, R. Larkin, Use of 3D reconstruction to correct for patient motion in SPECT, Phys. Med. Biol. 39 (3) (1994) 563–574.

[78] A. Kyme, B. Hutton, R. Hatton, D. Skerrett, L. Barnden, Practical Aspects of a Data-Driven Motion Correction Approach for Brain SPECT, IEEE Trans. Med. Imaging 22 (6) (2003) 722–729.

[79] A.Rahmim, K.Dinelle, J.C.Cheng, M.A.Shilov, W.P.Segars, S.C.Lidstone, S.Blinder, O.G.Rousset, H.Vajihollahi, B.M.Tsui, D.F.Wong, V.Sossi, Accurate event-driven motion compensation in high-resolution pet incorporating scattered and random events, IEEE Trans. Med. Imag. 27 (8).

[80] A.Rahmim, K.Dinelle, J.C.Cheng, M.A.Shilov, W.P.Segars, S.C.Lidstone, S.Blinder, O.G.Rousset, H.Vajihollahi, B.M.Tsui, D.F.Wong, V.Sossi, Accurate event-driven motion compensation in high-resolution pet incorporating scattered and random events, IEEE Trans. Med. Imag.

[81] F.Lamare, M. J. L. Carbayo, T. Cresson, G. Kontaxakis, C. C. L. R. A. Santos, A. J. Reader, D. Visvikis, List-mode-based reconstruction for respiratory motion correction in pet using non-rigid body transformations, Phys. Med. Biol. 52 (2007) 5187–5204.

[82] F.Buther, M.Dawood, L.Stegger, F.Wubbeling, M.Schfers, O.Schober, K.P.Schafers, List mode-driven cardiac and respiratory gating in PET, J. Nucl. Med. 50 (5) (2009) 674–681.

[83] J. McClelland, D. Hawkes, T. Schaeffter, A. King, Respiratory motion models: a review, Med Image Anal. 17 (1) (2013) 19–42.

[84] J.Nocedal, S.Wright, J.Stephen, Numerical Optimization (II ed.), Springer-Verlag, 2006.

[85] J. Bezdek, R. Hathaway, Some Notes on Alternating Optimization, Advances in Soft Computing, Lecture Notes in Computer Science 2275 (2002) 288–300.

[86] BrainWeb, Brainweb: Simulated brain database. http://www.bic.mni.mcgill.ca/ brain- web/. URL `http://www.bic.mni.mcgill.ca/brain-web/`

[87] S.Somayajula, C.Panagiotou, A.Rangarajan, Q.Li, S.R.Arridge, R.M.Leahy, Pet image reconstruction using information theoretic anatomical priors, submitting (2010).

[88] P.A.Viola, W.M.Wells, Alignment by maximization of mutual information, Computer Vision, 1995. Proceedings., Fifth International Conference on (1995) 16–23.

[89] W.M.Wells, P.Viola, H.Atsumi, S.Nakajima, R.Kikinis, Multi-modal volume registration by maximization of mutual information., Med. Image Anal. 1 (1) (1996) 35–51.

[90] F.Maes, A.Collignon, V.Dirk, G.Marchal, P.Suetens, Multimodality image registration by maximization of mutual information, IEEE transactions on Medical Imaging 16 (1997) 187–198.

[91] Shannon, A mathematical theory of communication, The Bell System Technical Journal 27 (1948) 379–423.

[92] J. Tang, A. Rahmim, Bayesian PET image reconstruction, Physics in Medicine and Biology 54 (2) (2009) 7063–7075. doi:10.1109/ISBI.2008.4541178.

[93] P. Mondal, K. Rajan, Image reconstruction by conditional entropy maximisation for PET system, IEEE Proceedings Vision, Image and Signal Processing 151 (5) (2004) 345–352. doi:10.1049/ip-vis.

[94] S. Pedemonte, A. Bousse, B. Hutton, S. Arridge, S. Ourselin, Probabilistic Graphical Model of SPECT/MRI, Lecture Notes in Computer Science, 7009 (2011) 167–174.

[95] T. Benaglia, D. Chauveau, D. Hunter, An em-like algorithm for semi- and nonparametric estimation in multivariate mixtures, Journal of Computational and Graphical Statistics 18 (2) (2009) 505–526.

[96] S. Shwartz, M. Zibulevsky, Y. Schechner, Fast kernel entropy estimation and optimization, Signal Processing 85 (5) (2005) 1045–1058.

[97] S. Pedemonte, A. Bousse, K. Erlandsson, M. Modat, S. Arridge, B. Hutton, S. Ourselin, GPU Accelerated Rotation-Based Emission Tomography Reconstruction, IEEE Nuclear Science Symposium Conference Record.

[98] A. Dale, B. Fischl, M. Sereno, Cortical surface-based analysis. I. Segmentation and surface reconstruction, Neuroimage 9 (2) (1999) 179–94.

[99] D. Higdon, J. Bowsher, V. Johnson, T. Turkington, D. Gilland, R. Jaszczak, Fully Bayesian Estimation of Gibbs Hyperparameters for Emission Computed Tomography Data, IEEE Transactions on Medical Imaging 16 (Issue 5) (1997) 516–526.

[100] H.H.Barrett, W.C.J.Hunter, B.W.Miller, S.K.Moore, Y.Chen, L.R.Furenlid, Maximum-Likelihood Methods for Processing Signals From Gamma-Ray Detectors., IEEE Trans. Nucl. Sci. 56 (2009) 725.

[101] H. Dam, S.Seifert, R.Vinke, P.Dendooven, H.Lohner, F.J.Beekman, D.R.Schaart, A practical method for depth of interaction determination in monolithic scintillator PET detectors., Phys. Med. Biol. 56 (2011) 4135–4145.

[102] C. Fiorini, A. Gola, R. Peloso, A. Geraci, A. Longoni, G. Padovini, P. Lechner, L. Struder, B. Hutton, K. Erlandsson, S. Mahmood, P. V. Mullekom, A. Pedretti, R. Moretti, G. Poli, G. Lucignani, Hicam: Development of a high-resolution anger camera for nuclear medicine., EEE Nuclear Science Symposium Conference Record (NSS/MIC) (2008) 3961–3964.

[103] H.O.Anger, Scintillation Camera., The review of Scientific Instruments 29 (1).

[104] A.Abba, A.Manenti, A.Suardi, C.Fiorini, A.Geraci, An efficient algorithm for spatial localization of multiple events from detector arrays in fpga devices., IEEE Nuclear Science Symposium Conference Record (NSS/MIC). (2008) 1661–1664.

[105] D.Thanasas, E.Georgiou, N.Giokaris, A.Karabarbounis, D.Maintas, C.N.Papanicolas, A.Polychronopoulou, E.Stiliaris, A correction method of the spatial distortion in planar images from -Camera systems., Journal of Instrumentation 4.

[106] S.Pedemonte, A.Gola, A.Abba, C.Fiorini, Optimum real-time reconstruction of Gamma events for high resolution Anger camera with the use of GPGPU., IEEE Nuclear Science Symposium Conference Record (NSS/MIC).

[107] H.Cramer, Mathematical Methods of Statistics., Princeton University Press, 1957.

[108] L.Boschini, C.Fiorini, SCIDRA: a Monte Carlo simulator for the study and optimization of gamma cameras based on a scintillator coupled to a silicon photodetector array., IEEE Nuclear Science Symposium Conference Record (NSS/MIC).

[109] G.F.Knoll, Radiation Detection and Measurement - 2nd edition., John Wiley & Sons, 1989.

[110] L.R.Furenlid, J.Y.Hesterman, H.H.Barrett, Real-time data acquisition and maximum-likelihood estimation for gamma cameras., IEEE-NPSS Real Time Conference.

[111] C.M.Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), 2nd edition., Springer, 2007.

[112] L.Saul, S.Roweis, Think globally, fit locally: Unsupervised learning of low dimensional manifolds., Journal of Machine Learning Research 4 (2003) 119–155.

[113] S.Roweis, L.Saul, Nonlinear dimensionality reduction by locally linear embedding., Science 290 (2000) 2323–2326.

[114] W. H.Vonach, Gamma-ray absorption coefficients for NaI(Tl)., Nuclear Instruments and Methods. 134 (Issue 2) (1976) 347–351.

[115] H. Barrett, T. White, L. Parra, List-mode likelihood, J. Opt. Soc. Am. A. Opt. Image Sci. Vis. 14 (11) (1997) 2914–2923.

[116] S.Kullback, R.A.Leibler, On Information and Sufficiency, Ann. Math. Statist. 22 (1) (1951) 79–86.

[117] D. Lindley, On a Measure of the Information Provided by an Experiment, Ann. Math. Statist. 27 (4).

[118] G.L.Zeng, G.T.Gullberg, Frequency Domain Implementation of the Three-Dimensional Geometric Point Response Correction in SPECT Imaging, IEEE Trans. Nucl. Sci. 39 (5) (1992) 1444–1454.

[119] W.P.Segars, Development of a new dynamic NURBS-based cardiactorso (NCAT) phantom., PhD dissertation, The University of North Carolina.

[120] Y.Zhang-O'Connor, J.A.Fessler, Fast predictions of variance images for fan-beam transmission tomography with quadratic regularization., IEEE Trans. Med. Imag. 26 (3) (2007) 335–246.

[121] G. Zheng, G. Gullberg, Exact iterative reconstruction for the interior problem., Phys Med Biol. 54 (19) (2009) 5805–5814.

[122] K. Thielemans, C. Tsoumpas, S. Mustafovic, T. Beisel, P. Aguiar, N. Dikaios, M. Jacobson, STIR: Software for Tomographic Image Reconstruction Release 2, Physics in Medicine and Biology 57 (4) (2012) 867–883.

[123] A. Rezaei, M. Defrise, G. Bal, C. Michel, M. Conti, C. Watson, J. Nuyts, Simultaneous reconstruction of activity and attenuation in time-of-flight PET, IEEE Trans. Med. Imag. 31 (12) (2012) 2224–2233.

[124] B. Fischl, D. Salat, A. V. D. Kouwe, N. Makris, F. Segonne, B. Quinn, A. Dale, Sequence-independent segmentation of magnetic resonance images, NeuroImage 23 (2004) 69–84.

[125] A. Sitek, Reconstruction of Emission Tomography Data Using Origin Ensembles, IEEE Transactions on Medical Imaging 30 (4) (2011) 946–56.

[126] A. Rosenthal, Efficient Framework for Model-Based Tomographic Image Reconstruction Using Wavelet Packets, IEEE Transactions on Medical Imaging 31 (7) (2012) 2346–57.

[127] N. Gac, S. Mancini, M. Desvignes, D. Houzet, High speed 3D tomography on CPU, GPU, and FPGA, EURASIP J. Embedded Syst. 2008 (2008) 5:1–5:12. doi:10.1155/2008/930250.

[128] J. D. Beenhouwer, D. Kruecker, S. Staelens, L. Ferrer, A. Chatziioannou, F. Rannou, Distributed Computing Platform for PET and SPECT Simulations with GATE, IEEE Nuclear Science Symposium Conference Record (NSS/MIC) M11-99 (2005) 2437–2440.

[129] J.A.Fessler, W.L.Rogers, Spatial Resolution Properties of Penalized-Likelihood Image Reconstruction: Space-Invariant Tomographs, IEEE Trans. on Imag. Proc. 5 (9) (2002) 1346–1358.

[130] Insight Toolkit, www.itk.org/.

[131] J.E.Stone, J.C.Phillips, P.L.Freddolino, D.J.Hardy, L.G.Trabuco, K.Schulten, Accelerating molecular modeling applications with graphics processors, J. Comp. Chem 28 (2007) 2618–2640.

[132] I.Ufimtsev, T.Martinez, Quantum chemistry on grapical processing units. 1. trategies for two-electron integral evaluation, Journal of Chemical Theory and Computation 4 (2) (2008) 222–231.

[133] H. Zaidi, M.-L. Montandon, Scatter compensation techniques in pet, PET Clinics 2 (2) (2007) 219–234, ¡ce:title¿PET Instrumentation and Quantification¡/ce:title¿. doi:10.1016/j.cpet.2007.10.003.

[134] G. Zeng, G. Gullberg, Frequency Domain Implementation of the Three-Dimensional Geometric Point Response Correction in SPECT Imaging, IEEE Transactions on Nuclear Science 39 (5) (1992) 1444–1454.

[135] R. Siddon, Fast calculation of the exact radiological path for a three-dimensional CT array, Medical Physics 12 (2) (1985) 252–255.

[136] NVidia CUDA Programming Guide, version 2.3 (Jan 2008).

[137] D. Kadrmas, E. F. B. Tsui, Application of Reconstruction-Based Scatter Compensation to Thallium-201 SPECT:Implementations for Reduced Reconstructed Image Noise, IEEE Trans. Med. Imaging 17 (3) (1998) 325–333.

[138] S. Ghosh, Distributed Systems - An Algorithmic Approach, Chapman and Hall/CRC, 2007.

[139] M.Schellmann, S.Gorlatch, D.Meilander, T.Kosters, K.Schafers, F.Wubbeling, M.Burger, Parallel medical image reconstruction: From graphics processors to grids, PaCT 2009 LNCS (5698) (2009) 457–473.

[140] H.Scherl, S.Hoppe, M.Kowarschik, Design and implementation of the software architecture for a 3-d reconstruction system in medical imaging, ICSE.

[141] J.C.Phillips, J.E.Stone, K.Schulten, Adapting a message-driven parallel application to gpu-accelerated clusters, SC2008.

[142] J.Dongarra, J.F.Pineau, Y.Robert, Z.Shi, F.Vivien, Revisiting matrix product on master-worker platforms, International Journal of Foundations of Computer Science 19 (6) (2008) 1317–1336.

[143] J.Dongarra, J.F.Pineau, Y.Robert, Z.Shi, F.Vivien, Matrix product on heterogeneous master-worker platforms, Proceedings of the 13th ACM SIGPLAN Symposium on Principles and practice of parallel programming (2008) 53–62.

[144] T.G.Mattson, B.A.Sanders, B.L.Massingill, Patterns for Parallel Programming, Addison Wesley, 2005.

[145] M.Herlihy, N.Shavit, The Art of Multiprocessor Programming, Morgan Kaufmann, 2008.

PUBLICATION LIST

Peer Reviewed Journal Papers

- **Niccolo Fuin and S. Pedemonte**, S. Arridge, S. Ourselin, B.F. Hutton: Efficient Determination of the Uncertainty for the Optimization of SPECT System Design : A Subsampled Fisher Information Matrix. IEEE TMI, July 2013 (under revision).
- D. Kazantsev, S. Arridge, **S. Pedemonte**, A. Bousse, K. Erlandsson, B. Hutton, S. Ourselin: An anatomically driven anisotropic diffusion filtering method for 3D SPECT reconstruction. Phys. Med. Biol. 57 (2012) 3793-3810.
- A. Bousse, **S. Pedemonte**, D. Kazantsev, S. Arridge, S. Pedemonte, K. Erlandsson, Ourselin, B. Hutton: Markov Random Field and Gaussian Mixture for segmented MRI-based partial volume correction in PET. Phys. Med. Biol. 2012, in press.

Peer Reviewed Conference Papers

- **S. Pedemonte**, B.F. Hutton, S. Arridge, S. Ourselin: Steady-state model of the radio-pharmaceutical uptake for MR-PET. MICCAI, Oct 2012.
- **S. Pedemonte**, A. Bousse, B.F. Hutton, S. Arridge, S. Ourselin: 4-D Generative Model for PET/MRI Reconstruction. MICCAI, Oct 2011.
- **S. Pedemonte**, B. F. Hutton, S. Arridge, S. Ourselin: Joint Estimation of the Activity and the Events of Interaction in SPECT Systems. IEEE NSS/MIC, Anaheim, CA, USA, Nov. 2012.
- **S. Pedemonte**, M. J. Cardoso, A. Bousse, C. Panagiotou, D. Kazantsev, S. Arridge, B. F. Hutton, S. Ourselin: Class Conditional Entropic Prior for MRI Enhanced SPECT Reconstruction. IEEE NSS/MIC, Knoxville, Oct 2010.
- **S. Pedemonte**, A. Bousse, K. Erlandsson, M. Modat, S. Arridge, B.F. Hutton, S. Ourselin: GPU Accelerated Rotation-Based Emission Tomography Reconstruction. IEEE NSS/MIC, Knoxville, 2010.
- **S. Pedemonte**, A. Gola, A. Abba, C. Fiorini: Optimum real-time reconstruction of Gamma events for high resolution Anger camera with the use of GPGPU. IEEE NSS/MIC, Orlando, Oct 2009.
- N. Fuin and **S. Pedemonte**, S. Arridge, S. Ourselin, B.F. Hutton: Subsampled Fisher Information Matrix for Efficient Estimation of the Uncertainty in Emission Tomography. IEEE NSS/MIC, Anaheim, CA, USA, Nov. 2012.
- N. Fuin, **S. Pedemonte**, S. Arridge, S. Ourselin, B.F. Hutton: Use of the Fisher Information Matrix to Optimize the Acquisition Protocol for a D-SPECT System. IEEE NSS/MIC, Anaheim, CA, USA, Nov. 2012.

- A. Bousse, **S. Pedemonte**, K. Erlandsson,B.A. Thomas, S. Ourselin, S. Arridge, B. Hutton, Markov Random Field and Gaussian Mixture for MRI-driven Partial Volume Correction in PET. IEEE NSS/MIC, Anaheim, CA, USA, Nov. 2012.
- K. Vunckx, S.R. Arridge, A. Bousse, D. Kazantsev, **S. Pedemonte**, B. F. Hutton. Unifying Global and Local Statistical Measures for Anatomy-Guided Emission Tomography Reconstruction. IEEE NSS/MIC, Anaheim, CA, USA, Nov. 2012.
- D. Kazantsev, A. Bousse, **S. Pedemonte**, B.F. Hutton, S. Ourselin, S.R. Arridge: An Anatomically Weighted Higher Order Total Variation Regularization with Bregman Iteration for 3D Emission Tomography. IEEE NSS/MIC, Anaheim, CA, USA, Nov. 2012.
- B.A. Thomas, K. Erlandsson, A.Reilhac, A. Bousse, D. Kazantsev, **S. Pedemonte**, K. Vunckx, S. Arridge, S. Ourselin, B. Hutton: A comparison of the options for brain partial volume correction using PET/MRI. IEEE NSS/MIC, Anaheim, CA, USA, Nov. 2012.
- D. Kazantsev, S.R. Arridge, **S. Pedemonte**, A. Bousse, B.F. Hutton, S. Ourselin: Robust Anisotropic Diffusion prior with Anatomical Regularization for 3D SPECT Reconstruction, Fully3D Reconstruction, Potsdam, Germany, July 2011.
- D. Kazantsev, A. Bousse, **S. Pedemonte**, S.R. Arridge, B.F. Hutton, S. Ourselin: Edge Preserving Bowsher Prior with non-local weighting for 3D SPECT Reconstruction. ISBI, Chicago, April 2011.
- N. Fuin, A. Bousse, **S. Pedemonte**, D. Kazantsev, S. Ourselin, S. Arridge, B.F. Hutton: Collimator design in SPECT, an optimisation tool. IEEE NSS/MIC, Knoxville, 2010.
- D. Kazantsev, **S. Pedemonte**, A. Bousse, C. Panagiotou, S.R. Arridge, B.F. Hutton, S. Ourselin: ET Bayesian Reconstruction using Automatic Bandwidth Selection for Joint Entropy Optimization. IEEE NSS/MIC, Knoxville, 2010.
- A. Bousse, N. Fuin, K. Erlandsson, **S. Pedemonte**, D. Kazantsev, S. Ourselin, S. Arridge, B. Hutton: Point Spread Function Optimization for parallel hole SPECT. IEEE NSS/MIC, Knoxville, 2010.
- A. Bousse, **S. Pedemonte**, D. Kazantsev, S. Ourselin, S. Arridge, B. Hutton: Weighted MRI-based Bowsher priors for anatomical-based SPECT brain image reconstruction. IEEE NSS/MIC, Knoxville, 2010.

Workshops

- **S. Pedemonte**, A. Bousse, B.F. Hutton, S. Arridge, S. Ourselin: Probabilistic Graphical Model of SPECT/MRI. MICCAI workshop on Machine Learning in Medical Imaging, Oct 2011.
- **S. Pedemonte** and S. Ourselin: Boss/Worker Model for Multi-GPU Programming, MICCAI workshop on High Performance Computing, Oct. 2011.

- **S. Pedemonte**, S. Arridge, B. F. Hutton, S. Ourselin: NiftyRec Tomographic Reconstruction Toolbox, IEEE ISBI workshop on Open Source Software for Medical Image Analysis, Barcellona, May 2012.
- B.A. Thomas, K. Erlandsson, I. Drobnjak, **S. Pedemonte**, K. Vunckx, A. Bousse, A. Reilhac-Laborde, S. Ourselin, B.F. Hutton: Framework for the construction of a Monte Carlo simulated brain PET-MR image database. PSMR 2013.

Patents

- **S. Pedemonte**, S. Ourselin, B. Hutton: Method and Apparatus for Processing Medical Imaging Data Including Emission Tomography Data. 2011, UK 10206274001.