ECOLE CENTRALE PARIS
ECOLE DOCTORALE SCIENCES POUR L'INGENIEUR (ED 287)

T H E S I S

submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy of Ecole Centrale Paris
Specialty : APPLIED MATHEMATICS

Defended on October 4^{th} 2013
by

Fabrice MICHEL

Multi-Modal Similarity Learning for 3D Deformable Registration of Medical Images

prepared at Ecole Centrale de Paris,
Center for Visual Computing

Jury :

Chairman :	Nassir NAVAB	-	Technische Universität München
Reviewers :	Xavier PENNEC	-	INRIA-Sophia Antipolis
	Nicolas VAYATIS	-	École Normale Supérieure de Cachan
Advisor :	Nikos PARAGIOS	-	École Centrale Paris
Examiners :	Julia SCHNABEL	-	University of Oxford
	Michael M. BRONSTEIN	-	Università della Svizzera Italiana
	Ali KAMEN	-	Siemens Corporate Research

2013ECAP0055

to my father, he always has been proud
of me, and with him gone I still strive
to make him proud

à mon père, il a toujours été fier de
moi, et malgré sa disparition je
m'efforce toujours de le rendre fier

Acknowledgements

Even though this thesis may seem as the work of one, it is indeed the work of many. And putting my name as only author of this work should not cast a shadow on the people without whom it would not have been possible. First and foremost, I would like to acknowledge the people I worked with on this thesis: Prof Nikos Paragios who provided deep scientific insight and stimulation while at the same time keeping a friendly atmosphere that was paramount throughout the years. Prof Michael Bronstein and Prof Alex Bronstein who taught me to think outside of the box, their energy helped me surpass myself. Prof Nassir Navab who welcomed me in his team where I had the most vision changing discussions about my work. I am extremely grateful to them to also have accepted being part of my thesis committee and I would also like to thank the reviewers Xavier Pennec and Nicolas Vayatis, and examiners Julia Schnabel and Ali Kamen for their careful reading of this manuscript and fruitful remarks in the report and during the defense.

The financial support for this thesis was granted by INRIA Saclay - Ile de France and Ecole Centrale Paris / Medicen Paris Region. This support helped me focus my day to day work on this thesis.

Second, I would like to state that I had the privilege to work among some of the most talented people I know and I dared to call them my friends, in France at Ecole Centrale Paris or in Germany at TU Muenchen, and I would like to thank all of them. I would especially like to thank Martin de La Gorce, PhD, for his generous help when I first started this work, Xiang Bo and Regis Behmo, PhD, for their patient help during the rehearsal of my defense, Loren Schwarz, PhD for being more than a friend and getting out of his way to welcome me in Munich.

Some people did not help me directly on my thesis, but their support helped me go through the good, the bad and the ugly with ease. I would like to especially thank my parents for their moral as well as financial support during those years. Christelle lovingly supported me and even sometimes put up with me when my results did not achieve my expectations and through the frustrations along the years.

The years that lead to this work have been full of life and helped me figure out my passion for research and its never ending enlightenment. The amazing feeling of discovering the work of others, setting the limits of the unknown for finally stepping into the unknown

is only truly met during this kind of deep focus on a subject that is the pursuit of a PhD thesis. I would recommend it in a heartbeat to anyone who is thrilled by knowledge.

Abstract

Even though the prospect of fusing images issued by different medical imagery systems is highly contemplated, the practical instantiation of it is subject to a theoretical hurdle: the definition of a similarity between images. Efforts in this field have proved successful for select pairs of images; however defining a suitable similarity between images regardless of their origin is one of the biggest challenges in deformable registration.

In this thesis, we chose to develop generic approaches that allow the comparison of any two given modality. The recent advances in Machine Learning permitted us to provide innovative solutions to this very challenging problem. To tackle the problem of comparing incommensurable data we chose to view it as a data embedding problem where one embeds all the data in a common space in which comparison is possible. To this end, we explored the projection of one image space onto the image space of the other as well as the projection of both image spaces onto a common image space in which the comparison calculations are conducted. This was done by the study of the correspondences between image features in a pre-aligned dataset.

In the pursuit of these goals, new methods for image regression as well as multi-modal metric learning methods were developed. The resulting learned similarities are then incorporated into a discrete optimization framework that mitigates the need for a differentiable criterion. Lastly we investigate on a new method that discards the constraint of a database of images that are pre-aligned, only requiring data annotated (segmented) by a physician. Experiments are conducted on two challenging medical images data-sets (Pre-Aligned MRI images and PET/CT images) to justify the benefits of our approach.

keywords: *Machine-Learning, deformable registration, multi-modal, metric-learning, 3D Medical Image*

Résumé

Alors que la perspective de la fusion d'images médicales capturées par des systèmes d'imageries de type différent est largement contemplée, la mise en pratique est toujours victime d'un obstacle théorique : la définition d'une mesure de similarité entre les images. Des efforts dans le domaine ont rencontrés un certain succès pour certains types d'images, cependant la définition d'un critère de similarité entre les images quelle que soit leur origine et un des plus gros défis en recalage d'images déformables.

Dans cette thèse, nous avons décidé de développer une approche générique pour la comparaison de deux types de modalités donnés. Les récentes avancées en apprentissage statistique (Machine Learning) nous ont permis de développer des solutions innovantes pour la résolution de ce problème complexe. Pour appréhender le problème de la comparaison de données incommensurables, nous avons choisi de le regarder comme un problème de plongement de données : chacun des jeux de données est plongé dans un espace commun dans lequel les comparaisons sont possibles. A ces fins, nous avons exploré la projection d'un espace de données image sur l'espace de données lié à la seconde image et aussi la projection des deux espaces de données dans un troisième espace commun dans lequel les calculs sont conduits. Ceci a été entrepris grâce à l'étude des correspondances entre les images dans une base de données images pré-alignées.

Dans la poursuite de ces buts, de nouvelles méthodes ont été développées que ce soit pour la régression d'images ou pour l'apprentissage de métrique multimodale. Les similarités apprises résultantes sont alors incorporées dans une méthode plus globale de recalage basée sur l'optimisation discrète qui diminue le besoin d'un critère différentiable pour la recherche de solution. Enfin nous explorons une méthode qui permet d'éviter le besoin d'une base de données pré-alignées en demandant seulement des données annotées (segmentations) par un spécialiste. De nombreuses expériences sont conduites sur deux bases de données complexes (Images d'IRM pré-alignées et Images TEP/Scanner) dans le but de justifier les directions prises par nos approches.

mots clés: *Apprentissage Statistique, Recalage Déformable, Multi-Modal, Apprentissage de Métrique, Image Médicale 3D*

Contents

List of Figures

List of Algorithms

Chapter 1

Introduction

The intent of this work is to highlight a domain which is at crossroads between *Computer Vision*, *Medical Image Analysis* and *Machine Learning*: the comparison of data issued by different modalities. We will focus here on images but to some extent, a wider range of data can be considered. In *Computer Vision* and *Medical Image Analysis*, the problem of the alignment of two (or more) images has attracted a lot of attention, and is referred to as the *Registration problem* in *Medical Image Analysis*. In order to put the images into alignment, one has to define a criterion that will give some insight on the goodness of fit. For this application, the criterion has to be based on the images and their spatial position and in the very frequent case where images are issued by different modalities, the registration model will have to deal with the comparison of data issued by different modalities (in this work we will refer to a modality as an imaging device, such modalities range to the very simple household camera to the very expensive MRI scanner).

In the recent years, a new interest in *Machine Learning* has grown for *Metric Learning*, the aim of which is to learn in a data set the proximity of elements based on a user defined or automatically defined criteria. The definition of a metric is in essence the definition of a new space in which the data is mapped. Learning the metric allows to have a finely tuned control on the proximity of objects considered similar by the user or by an automated system in the new space.

The aim of this work is to benefit from the recent advances in *Metric Learning* towards the creation of novel finely tuned alignment criteria, for multi-modal and deformable image registration.

1.1 Background and motivations

Image alignment is a topic that has been studied a lot, in *Computer Vision*, the deformable image alignment problem sees applications in stitching and mosaicing as well as optical

flow to name a few, in *Medical Image Analysis* deformable image alignment that is referred as the registration of medical images is an active field of research and has been for over 20 years.

1.1.1 Applications in *Computer Vision*

Stiching and Mosaicing: this is a very old problem of *Computer Vision* yet it is still largely unsolved. The idea is to compose a larger photograph with smaller pictures by assembling them seamlessly. The problem is twofold, first correspondences between two sub-pictures need to be found in order to fuse them correctly, but second and foremost the pictures there has been at least a rotation of the camera between the two pictures, which yields a change in perspective between them, but there might also be a change in the camera (changing focal is technically already a change in camera), and larger distortions can appear. Now if some objects are moving between pictures, we are faced with a deformable alignment problem, where we try to map two realities that are similar but of different shapes.

Optical flow: Optical flow, refers to the problem of detecting the movement from one frame to the next in a video. This problem is of huge interest as can be the stepping stone to many other algorithms. Among the most famous are the video compression algorithms. Detecting motion in a video allows to better compress a video by applying different compression rates to different parts of the image according to the motion they are under. Intuitively we don't need to retain all non moving parts of the video for each frame, but only keep them stored for the first frame it appears in. Optical flow is also paramount in the complex tracking problems such as following people in crowds.

Finding motion from one frame to the other however can be a tough problem. It is usually based on the comparison pixels by pixels: for each pixels in one image we look for the most comparable pixels in its vicinity in the other image, the displacement of this pixel is denoted as its flow. The problem is that the object in the image might undergo serious deformations rendering a naive comparison approach useless. Deformable image alignment plays an essential role in this case.

1.1.2 Registration of Medical Images

In the context of the clinical practice, *Computer Vision* can help the physician in various ways. Among other things, it can be used to automate tedious tasks and allow the medical crew to focus on the patient, it can help the diagnostic decision and give the physicians

reliable measures, last it can assist the physician on some medical procedures. Medical image registration is a tool used in all these tasks. Here I am going to give some use cases.

Tumor tracking: When tumors are spotted in a patient, especially when the tumor is benign or when it is in a critical area like the brain, resection is not always the course of action taken, and treatments like chemotherapy or radiation therapy can be considered. In those cases the physicians need to have a close control on the progression of the tumor. To this end the volume of the tumor is a measure that is often taken into consideration. If the volume of the tumor decreases, the treatment is effective.‘

In order to track the progression of the tumor over time, several volumetric scans will be taken (either CT-scan, MRI, PET-scan or other technologies depending on the kind of tumor and the biological tissues involved). The radiologist or the surgeon will then delineate the tumor in 3D in each image, and be able to control the volume of the tumor directly. However , in each of the scans, the patient might not be in the exact same position, meaning that the tumor will not be imaged at the same angle, which could lead to severe biases in the volume estimation. Image alignment will be used here to recover in one image the position of the patient in the other. Rotations and translations will be applied to one image to closely match the other, this way both tumors will be visualized at the same angle.

Worse still, tumors are often located in deformable tissues and organs, that are affected by the breathing of the patient and their position on the table. The most common example is the deformation incurred by the internal organs during the breathing, but gravity also takes a real part and a slight variation in the position of the patient can lead to differently looking images even if the patient hold his breath. Evidently when the organs are deformed, the tumors inside them are also deformed. In this case, one cannot measure reliably the volume of the tumor since it is directly linked to the deformation that the tumor is subject to. Deformable registration aims at recovering the deformation in one image such that it matches that of the other. Now we are not trying to find the rotation and the translation aligning the images anymore but we are trying to displace every bit of tissue so that the images are comparable, this is arguably a much harder problem. In figure 1.1, the deformation due to the breathing in the organs, and its impact on a lung tumor is shown in CT images.

Intra-operative tissue localization: In most cases before a surgical procedure a pre-operative radiologic image is taken in order to help the surgical planing and to identify sensitive tissues and organs that should not be tempered with during the dissection. Unfortunately, the localization of body tissues is intricate and the tissue movements and shifts caused by the dissection make things even harder. It is then often hard for the surgeon to

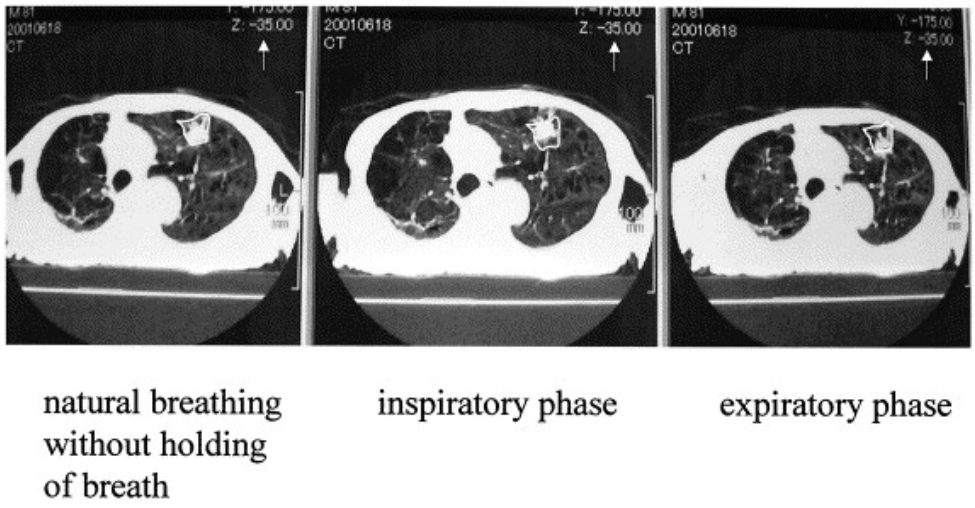


Figure 1.1: Image Courtesy of [Onimaru 2003]. Organs deformation due to breathing and its impact on a lung tumor delineated in the image.

locate precisely the tissues that he planed for resection and also the tissues that should not be tempered with. It is common practice in some procedures to have a radiologist perform a sonography during the intervention to clarify the position of some tissues. The sonography images are often not readily interpretable by the surgeon and the presence of the radiologist is crucial. This technique is of course subject to the bias of the interpretation of the radiologist opinion but also doesn't give an accurate estimation of the current position to the surgeon. In most cases it is of course not thinkable to use a CT scanner during the surgery to have a clearer insight. However if one such image could be produced, the surgeon could gain more control on the operation.

This problem attracts a lot of attention in the medical image analysis community, and implies the deformable (to account for tissue shift and dissection) registration of the intra-operative sonography with the pre-operative radiologic image. This is an even harder problem than the one of the tumor traking, because both images are not issued by the same radiologic modality. We will see in details why it is theoretically a much harder problem, but a quick look at two typical images, one of sonography and one of a MRI-scanner for instance figure (1.2), can convince the reader that this problem is not trivial.

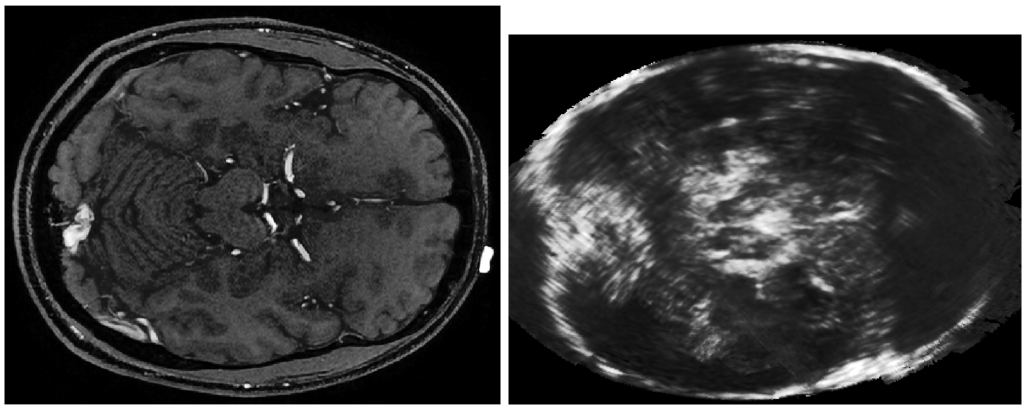


Figure 1.2: Images courtesy of the Technical University of Munich. Side by side comparison of an MRI image of the brain and an extra-cranial ultrasound image of the brain. The problem of alignment is of the utmost complexity

Chapter 2

Metric Learning for Multi-Modal Image Registration

In this work, I will refer to Image Registration as the process which goal is to put two images in alignment. The criterion for the alignment may vary from an alignment achieved only through a series of global translation to an alignment achieved through local translation of image parts. In any image registration process, three theoretical bricks are involved. The first brick is the definition and modelization of the deformation field applied to one image to match the other. The second brick is the criterion that is used to tell if two images match. Finally, one needs an optimizer which purpose will be to find the transformation that satisfies the matching criterion best.

In this work, I will mainly focus on the second brick which is the definition of a matching criterion, in the case where the two images considered are issued by different modalities. The recent advances in the field of metric learning and discrete optimization motivated us in the definition of matching criteria based on the prior knowledge of images that are already in alignment.

2.1 Image Registration

Let us consider two images I and J that are discrete maps mapping from $\Omega \subset \mathbb{R}^d$ to $\mathbb{R}$, d is the dimension of the image space and most frequently $d = \{2, 3\}$. We will refer to I as the *static*, *fixed* or *target* image and to J as the *moving* or *source* image in the registration, as one is matched to the other. In this context image registration is the process used to find a transformation $\mathbf{T}^\star$ belonging to the space of transformations $\mathcal{T} \subset \{\mathbf{T} \in \mathbb{R}^d \to \mathbb{R}^d\}$ such that $J \circ \mathbf{T}$ is as close as possible to I in the sense of the criterion $\mathcal{C} \in \{\mathbb{R} \times \mathbb{R} \to \mathbb{R}\}$. This

is often achieved through the minimization of an energy functional:

$$\mathbf{T}^\star = \underset{\mathbf{T}\in\mathcal{T}}{\operatorname{argmin}}\, \mathcal{C}\left(I, \gamma\left(J \circ \mathbf{T}\right)\right) \tag{2.1}$$

Here we added γ which is the interpolation map that puts $J \circ \mathbf{T}$ in the same pixel grid as I. For the clarity of this work we will voluntarily drop it from the equations and assume it is taken care of in the deformation process. If no restriction is done on the transformation space $\mathcal{T}$, this problem is ill-posed and the solution is not unique. This issue is accounted for in two ways: first the transformation $\mathbf{T}$ is parametrized with the parameter vector $\boldsymbol{\theta}$ and the search is not carried out in the space of transformations but rather in the space of the parameters of $\mathbf{T}$, the space Θ of which $\boldsymbol{\theta}$ is a member. Second, a smoothness or regularization term $\mathcal{R}$ in the energy is used as a way to select between potentially many transformation candidates.
The energy minimization is then rewritten:

$$\left\{ \begin{array}{rcl} \mathbf{T}^\star & = & \mathbf{T}_{\boldsymbol{\theta}^\star} \\ \boldsymbol{\theta}^\star & = & \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}}\, \mathbf{E}(\boldsymbol{\theta}) = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}}\, \mathcal{C}\left(I, J \circ \mathbf{T}_{\boldsymbol{\theta}}\right) + \lambda\mathcal{R}(\mathbf{T}_{\boldsymbol{\theta}}) \end{array} \right. \tag{2.2}$$

In this formulation, the three parts of registration become apparent. First we have the transformation parametrization and the definition of the regularization term $\mathcal{R}$. Second we have the definition of the comparison criterion $\mathcal{C}$. The balance between $\mathcal{C}$ and $\mathcal{R}$ is ruled by λ. Finally we need to devise a way to reach the minimum of the energy functional and thus find a suitable optimizer for the pair $\mathcal{R}$ and $\mathcal{C}$.

Let us first introduce briefly some of the most common types of transformations.

2.2 Transformation Model

When we address the problem of registration from the point of view of the transformation model, two separate classes arise. We can distinguish between the non deformable class of registration problems on one hand and the deformable class of registration problems on the other hand. If we speak in terms of control points and degrees of freedom, a non-deformable transformation amounts to have a control point in each corner of the image and depending on the type of transformation we apply some displacement to the points. The more independent the movement of the control points is, the more degrees of freedom we have and the wider is the range of deformation attained. In all cases the movement of each pixel is globally dependant on the movement of the others.

A deformable transformation on the other hand does not restrain itself to the corner of the image and places control points over all the image, this way all pixels can be moved

independently. In practice though, having a control point per pixel is not computationally efficient, and a control point controls a group of pixels. In the case of deformable registration the movement of each pixel is only locally dependant on the movement of the others.
Let us review some of the most common transformation models.

2.2.1 Non-deformable transformation

In all non-deformable cases, the number of degrees of liberty (the dimension of Θ) is very low compared to the number of pixels. The occurrences of multiple solutions to the same problem are thus next to none. The regularization term $\mathcal{R}$ is of no use in this case.

Rigid body transformation: this is the simplest non-deformable transformation. Here only rotations and translations are considered. In 3D this amounts to 3 rotation angles and 3 translation parameters. The dimension of Θ is then 6. Examples of it being used can be found in [Maes 1997, Roche 2001]. If the transformation is parametrized with $\boldsymbol{\theta} = (r_\alpha, r_\beta, r_\varphi, t_x, t_y, t_z)$, then one can represent the rigid body transformation $\mathbf{T}_{\boldsymbol{\theta}}$ using homogeneous coordinates (in this setting quaternions are often used to parametrize the 3D rotation):

$$\begin{cases} \mathbf{R}_{\boldsymbol{\theta}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos r_\alpha & -\sin r_\alpha & 0 \\ 0 & \sin r_\alpha & \cos r_\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos r_\beta & 0 & -\sin r_\beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin r_\beta & 0 & \cos r_\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos r_\varphi & -\sin r_\varphi & 0 & 0 \\ \sin r_\varphi & \cos r_\varphi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ \mathbf{T}_{\boldsymbol{\theta}} = \mathbf{R}_{\boldsymbol{\theta}} \cdot \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{cases} \tag{2.3}$$

Affine transformation[Jenkinson 2001]: is the most distorted type of global transformation, allowing 12 degrees of freedom in 3D. Rigid body deformations are a subset of the affine transformations where scaling and shearing are not taken into consideration. Affine are the most commonly used transformations and many deformable methods apply affine registration to the images prior to the deformable registration (e.g. [Rueckert 1999]). If $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{12})$ then,

$$\mathbf{T}_{\boldsymbol{\theta}} = \begin{bmatrix} \boldsymbol{\theta}_1 & \boldsymbol{\theta}_2 & \boldsymbol{\theta}_3 & \boldsymbol{\theta}_{10} \\ \boldsymbol{\theta}_4 & \boldsymbol{\theta}_5 & \boldsymbol{\theta}_6 & \boldsymbol{\theta}_{11} \\ \boldsymbol{\theta}_7 & \boldsymbol{\theta}_8 & \boldsymbol{\theta}_9 & \boldsymbol{\theta}_{12} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

2.2.2 Deformable transformation

A wide range of deformable transformations can be found in the literature, it is out of the scope of this work to make an exhaustive description of all the different methods, instead we redirect the curious reader to the work of Sotiras [Sotiras 2011]. Here we will describe two transformation models that have attracted a lot of attention in the last decade. Most commonly, when dealing with deformable transformations, the transformation is described in each pixel position as an additive displacement imposed to each pixel: $\mathbf{T}_{\boldsymbol{\theta}} = \mathbf{Id} + \mathbf{u}_{\boldsymbol{\theta}}$.

Diffusion Models and Demons Diffusion models are part of the physics based models for which the transformation $\mathbf{T}$ is not parametrized but instead is governed by a state equation. In the case of diffusion models, the state equation is the diffusion equation:

$$\triangle \mathbf{u} + \mathbf{F} = 0 \tag{2.5}$$

This diffusion model was introduced in [Thirion 1998], where the analogy to Maxwell's demons for a mixed gas is used. The demons are mathematical entities that apply forces on a membrane to help the image diffuse through it. The membrane is one way (as with a gas) and only let forces through in one direction. In the case of 3D medical images, demons are placed in every location where the image is not constant ($\nabla I \neq 0$) and push the field of deformation. The algorithm is an iterative process of small displacements for which the optical flow is conserved. The demon force is given by optical flow with velocity $\mathbf{v}$:

$$\mathbf{v} = \frac{(I - J \circ \mathbf{T}) \nabla I}{(\nabla I)^2 + (I - J \circ \mathbf{T})^2} \tag{2.6}$$

There is then a regularization step on the displacement field achieved through Gaussian smoothing. Later, in [Pennec 1999] it is shown that the *Demon's Algorithm* is mathematically equivalent to the gradient descent on the Sum of Square Distance criterion:

$$\mathcal{C}(I, J, \mathbf{T}) = \int_{\Omega} \left(I \circ \mathbf{T}^{(-1)} - J\right)^2 \tag{2.7}$$

The Gaussian smoothing can also be interpreted in the same framework as:

$$\mathcal{R}(\mathbf{T}) = \int_{\Omega} \|K \star \mathbf{T}\|^2 \tag{2.8}$$

Where K is the smoothness kernel. This algorithm has grown very famous for its computational efficiency and many variants have since been done. Extensions to all the commonly used similarity criteria have been proposed [Cachier 2003] and most notably the diffeomorphic version of the *Demon's Algorithm* was proposed in [Vercauteren 2007b].

Interpolated deformation models and Free-form deformations Interpolation models are the physics based model counterparts, originating from *Computer Vision*, there is no consideration made for forces pushing the field for deformation. The interpolation model is based on the assumption that the displacement is known for a select number of pixel positions in the image and the global deformation is interpolated from their movement. Depending on the interpolation method, the moving pixels location can be placed randomly in the deformation field [Shen 2002] or be placed on a regular grid as is the case in the Free-Form Deformations (FFD) transformation model [Sederberg 1986]. Computational efficiency and interpolation accuracy made of cubic *B-splines* based FFDs one of the dominant transformation models in medical image registration [Rueckert 1999].

Let us here briefly describe this model, according to the works of [Rueckert 1999]. We consider a grid with uniform spacing δ, this grid is superimposed on the image domain. Each vertex of the grid constitutes a control point. Since this is an interpolation model, the transformation is parametrized by the displacement in 3D of the control points:

$$\boldsymbol{\theta} = \{\boldsymbol{\theta}_{i,j,k} | i = \{1, \ldots, n_x\}, j = \{1, \ldots, n_y\}, k = \{1, \ldots, n_z\}\}$$

The cubic *B-splines* interpolation model then writes:

$$
\begin{cases}
\mathbf{u}_{\boldsymbol{\theta}}(x,y,z) &= \sum_{l=0}^{3}\sum_{m=0}^{3}\sum_{n=0}^{3} B_l(u)B_m(v)B_n(w)\boldsymbol{\theta}_{i+l,j+m,k+n} \\
& u = \dfrac{x}{n_x} - \left\lfloor \dfrac{x}{n_x} \right\rfloor, \quad v = \dfrac{y}{n_y} - \left\lfloor \dfrac{y}{n_y} \right\rfloor, \quad w = \dfrac{z}{n_z} - \left\lfloor \dfrac{z}{n_z} \right\rfloor \\
& i = \left\lfloor \dfrac{x}{n_x} \right\rfloor - 1, \quad j = \left\lfloor \dfrac{y}{n_x} \right\rfloor - 1, \quad k = \left\lfloor \dfrac{z}{n_x} \right\rfloor - 1 \\
B_0(u) &= \dfrac{1}{6}(1-u)^3 \\
B_1(u) &= \dfrac{1}{6}(3u^3 - 6u^2 + 4) \\
B_2(u) &= \dfrac{1}{6}(-3u^3 + 3u^2 + 3u + 1)^3 \\
B_3(u) &= \dfrac{u^3}{6}
\end{cases}
\tag{2.9}
$$

Cubic *B-splines* allow for a smooth transformation, and it has been proven [Choi 2000, Rueckert 2006], that if the displacement of the control points does not exceeds 0.4δ, the resulting transformation is diffeomorphic. In order to achieve larger displacements, a composition of diffeomorphic transformations is realized [Rueckert 2006].

Several regularization terms have been used for FFD based transformations. Here we present the regularization term used in [Rueckert 1999]:

$$
\mathcal{R}(\mathbf{T}_{\boldsymbol{\theta}}) = \iiint_{\Omega} \left(\frac{\partial^2 \mathbf{u}_{\boldsymbol{\theta}}}{\partial x^2}\right)^2 + \left(\frac{\partial^2 \mathbf{u}_{\boldsymbol{\theta}}}{\partial y^2}\right)^2 + \left(\frac{\partial^2 \mathbf{u}_{\boldsymbol{\theta}}}{\partial z^2}\right)^2 + 2\left[\left(\frac{\partial^2 \mathbf{u}_{\boldsymbol{\theta}}}{\partial xy}\right)^2 + \left(\frac{\partial^2 \mathbf{u}_{\boldsymbol{\theta}}}{\partial xz}\right)^2 + \left(\frac{\partial^2 \mathbf{u}_{\boldsymbol{\theta}}}{\partial yz}\right)^2\right]
\tag{2.10}
$$

This term imposes a penalty on transformations that are not smooth enough. In this case, the search space $\mathcal{T}$ is the space of the piecewise cubic *B-splines* with smoothness constraint parametrized by λ.

2.3 Optimization strategy

The objective of this dissertation is clearly not to give an exhaustive view of the optimization strategies. Optimization is in itself a complete field of research and many if not all

applicable optimization strategies have been tried. Moreover, many registration methods have tuned optimization algorithms to their special needs. Describing optimization strategies exhaustively would be like trying to examine each and every registration method. Some insight on the many optimization strategies in registration is given in [Sotiras 2011]. However, one can separate the optimization algorithms with respect to their input variables. In our case, the optimization strategy is used to find the minimum of the energy $\mathbf{E}$ with respect to the parameter set $\boldsymbol{\theta}$ in the search space Θ.

On one hand the space in which the solution is searched is a continuous space, and the problems are solved in dense subspaces of $\mathbb{R}$ or $\mathbb{C}$ or even more exotic dense valued spaces. Optimization in this continuous space is usually referred to as *continuous optimization*. This is opposed to *discrete optimization*, the search space of which is a subspace of $\mathbb{Z}$ or $\mathbb{N}$

2.3.1 Continuous Optimization

A good overview of some of the most used optimization algorithms for registration can be found in [Klein 2007]. The optimization strategy used to reach the optimal set of parameter is iterative, starting with an initial guess $\boldsymbol{\theta}_0$:

$$\left\{ \begin{array}{lcl} \boldsymbol{\theta}_{t|t=0} & = & \boldsymbol{\theta}_0 \\ \boldsymbol{\theta}_{t+1} & = & \boldsymbol{\theta}_t + \alpha_t \mathbf{d}_t \end{array} \right. \tag{2.11}$$

most of the time, the *search direction* $\mathbf{d}_t$ is dependant on the value of $\boldsymbol{\theta}_t$ at previous iterations, it might also be the case for the *step size* α_t.

The simplest of all continuous optimization strategies is the gradient descent approach, in which the step size α_t is constant, and the direction $\mathbf{d}_t$ is given by the gradient of the energy in the direction of the parameter $\boldsymbol{\theta}$: $\mathbf{d}_t = \nabla_{\boldsymbol{\theta}_t} \mathbf{E}(\boldsymbol{\theta}_t)$. While simple this method still yields good results and was used for instance in [Rueckert 1999]. The step size of the gradient descent can also be adjusted for by line search at each iteration to yield the maximal decay in the energy along this direction [Press 1986]. This technique however suffers from poor convergence rate (we will not formally define convergence here but one can view it as a stabilization in the decrease of the energy) and is appropriate only when the estimation of the gradient is not very time consuming.

More powerful optimization strategies such as *Conjugate gradient methods* do not follow directly the direction of the gradient but also take into account the direction of the gradient at previous iterations to have a more thorough search of the space.

To have an even better convergence rate, more information on the energy functional is needed. Second order information like the Hessian of the functional is used in the *Newton-type* methods. Computation of the Hessian is usually not very efficient, and most of the

time, only approximations of the Hessian matrix are considered which are themselves based on approximations of the gradients of the functional as in the *Broyden-Fletcher-Goldfarb-Shanno* (BFGS) method [Luenberger 2008]. Newton-like procedures are most notably used in *Demon's* registration algorithms where Gauss-Newton and other second order method are employed [Vercauteren 2007a].

Other continuous optimization strategies include *stochastic gradient* strategies, in which the gradient is approximated with the use of random perturbation vectors, this allow for a greater computational efficiency as well as can make the method less prone to falling into local minima. Last but not least, methods that do not make use of first or second order information on the energy; such as the *Powell's* methods can be of great interest as we shall see in the next sections, as not all objective functions are differentiable, as well as the estimation of the derivatives might be computationally prohibitive. Starting from a set of initial parameter vectors (usually the normals to each axis), Powell's method operates by performing linear searches along each search vector, the new direction is updated as a linear combination of the search vectors. Use of this method has been done in [Maes 1997, Roche 1998, Pluim 2000, Chung 2002].

2.3.2 Discrete Optimization

We have seen in the case of continuous optimization that for most algorithms, one of the driving evolutions is the search for more efficient methods that will allow for less and less functions and gradient evaluations. Sometimes also, the gradient is not formally expressible as well as the Hessian. In those cases, local approximations to these quantities are computed, thus requiring even more computation time. Methods that do not make use of derivatives like *Powell's* do exist but are usually used when there is no access to higher order quantities, since those method have very low convergence rate. Last but not least, each of these methods needs an energy specific treatment and adaptation to a new similarity criterion is not straightforward, since it implies finding derivation strategies.

This is opposed to discrete methods that perform a global search such that they are not sensitive to the initial position $\boldsymbol{\theta}_0$. They do not require the derivatives of the functional and are very modular with respect to the change of the energy, as long as some conditions on the energy are still respected. Lastly, the discretization of the search space allows for computationally more efficient methods. The main drawback of these methods also stems from the discretization, since the solution will only be as precise as the discretization is, and augmenting the precision will go as a trade off with the computational efficiency.

Discrete optimization is closely linked to discrete graph optimization and discrete Markov Random Fields (MRF). In the setting of graph optimization, each parameter $\boldsymbol{\theta}_p$ is modeled by a graph node, and the interaction between the graph parameters given by

the functional energy $\mathbf{E}$ is modeled by the edges of the graph. The graph G is represented by the set of its nodes $\mathcal{V}$ and edges $\mathcal{E}$: $G = \{\mathcal{V}, \mathcal{E}\}$. The value taken by each parameter $\boldsymbol{\theta}_p$ is discretized and given by a label $\boldsymbol{\ell} \in \{\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_N\}$. In this dissertation, we will only consider graph with unary and pairwise interactions, and no clique of higher order will be considered. This consideration constrains the energy to be the sum of unary potentials $u_p, p \in \mathcal{V}$ and of pairwise potentials $v_{p,q}, \{p, q\} \in \mathcal{E}$.

$$\mathbf{E}(\boldsymbol{\ell}) = \sum_{p \in \mathcal{V}} u_p(\boldsymbol{\ell}_p) + \sum_{\{p,q\} \in \mathcal{E}} v_{p,q}(\boldsymbol{\ell}_p, \boldsymbol{\ell}_q) \tag{2.12}$$

The optimization is here done on $\boldsymbol{\ell}$ in the space of allowable label sets $\mathcal{L}$ and the optimal parameter $\boldsymbol{\theta}^\star$ is given as a function f of the optimal labeling $\boldsymbol{\ell}^\star$:

$$\left\{ \begin{array}{rcl} \boldsymbol{\theta}^\star & = & f\left(\boldsymbol{\ell}^\star\right) \\ \boldsymbol{\ell}^\star & = & \underset{\boldsymbol{\ell} \in \mathcal{L}}{\operatorname{argmin}}\, \mathbf{E}(\boldsymbol{\ell}) \end{array} \right. \tag{2.13}$$

In the case of registration, the similarity criterion will be cast onto the unary potentials while the regularization term will be attributed to the pairwise interactions. Here we will discuss the works of *Glocker et al.* [Glocker 2008], where deformable registration using FFDs is cast into the framework of MRFs.

First of all, the similarity criterion $\mathcal{C}$ takes a less general form and is made into a point wise criterion:

$$\mathcal{C}\left(I, J \circ \mathbf{T}\right) = \int_\Omega c\left(I, J \circ \mathbf{T}\right) \tag{2.14}$$

In the case of FFD, a grid of control points is superimposed on the image, and the grid spacing is not necessarily the same as the pixel size. In the graphical model, we want each node to represent a control point, so we want to evaluate the similarity in each control point. Thus we need to evaluate the contribution of each control point $\mathbf{p} \in G$ to the similarity cost:

$$\mathcal{C}\left(I, J \circ \mathbf{T}\right) = \frac{1}{|G|} \sum_{p \in G} \int_{\mathbf{x} \in \Omega} \hat{\eta}\left(|\mathbf{x} - \mathbf{p}|\right) c\left(I(\mathbf{x}), J \circ \mathbf{T}(\mathbf{x})\right) \mathrm{d}\mathbf{x} \tag{2.15}$$

where $\hat{\eta}$ is the term that projects the information from the image level to the grid level:

$$\hat{\eta}\left(|\mathbf{x} - \mathbf{p}|\right) = \frac{\eta\left(|\mathbf{x} - \mathbf{p}|\right)}{\int_{\mathbf{x} \in \Omega} \eta\left(|\mathbf{x} - \mathbf{p}|\right) \mathrm{d}\mathbf{x}} \tag{2.16}$$

and η is the function that weights the contribution of each control point to the deformation following the deformation model. In the case of FFD, η is a sum of *B-splines* interpolation functions as presented in equation (2.9).

Now using the similarity cost in equation (2.15), we can construct the unary potential by defining that the labels will have influence on the control point's displacement $\boldsymbol{\theta}_{\mathbf{p}}^{\boldsymbol{\ell}_{\mathbf{P}}}$:

$$u_p(\boldsymbol{\ell}_p) = \int_{\mathbf{x}\in\Omega} \hat{\eta}\left(|\mathbf{x}-\mathbf{p}|\right) c\left(I(\mathbf{x}), J(\mathbf{x}+\boldsymbol{\theta}_{\mathbf{p}}^{\boldsymbol{\ell}_{\mathbf{P}}})\right) \mathrm{d}\mathbf{x} \tag{2.17}$$

the pairwise term, representing the regularization of the deformation field, is defined as a simple Ising model:

$$v_{p,q}(\boldsymbol{\ell}_p, \boldsymbol{\ell}_q) = \omega_{pq} \left|\boldsymbol{\theta}_{\mathbf{p}}^{\boldsymbol{\ell}_{\mathbf{P}}} - \boldsymbol{\theta}_{\mathbf{q}}^{\boldsymbol{\ell}_{\mathbf{Q}}}\right| \tag{2.18}$$

this regularization makes a registration process similar to fluid-like registration presented in [Christensen 1994]. Finally, the edge system $\mathcal{E}$ is defined as $\mathcal{E} = \{(p,q)|\mathbf{p}\in\mathcal{V}, \mathbf{q}\in\mathcal{N}(\mathbf{p})\}$, where $\mathcal{N}(\mathbf{p})$ is an appropriate neighbourhood of $\mathbf{p}$, and allows for a regularization of the deformation field in the neighbourhood of the control point.

A whole lot of algorithms have been around to solve discrete MRFs with pairwise interactions, but two of them stand out as they have actually been used for medical image registration, TRW-S [Kolmogorov 2006] and Fast-PD[Komodakis 2007, Komodakis 2008]. Both are based on an Linear Programming (LP) relaxations of the problem, as the integral LP program is NP-hard. In TRW-S (sequential tree-reweighted message passing) the graph is decomposed into a set of trees, then sequentially, every node of the graph is visited and belief propagation is performed on each tree containing it. The solution is obtained by averaging on the result of all trees involved. TRW-S was successfully used for registration in [Shekhovtsov 2008]. Fast-PD on the other end treats the problem as a primal-dual gap minimization problem. It is very flexible for image registration since it only requires the pairwise potentials to be non-negative, the approximate optimality is guaranteed and computationally very efficient. Successful applications of FastPD to registration problems can be found in [Glocker 2008, Glocker 2009, Ou 2009, Zikic 2010].

2.4 Image Matching Criteria

The image matching criterion is the core subject of this thesis. So far we talked about the transformation model and the optimization strategy, but in the end the quality of the registration will greatly depend on how well we can tell if two images are into alignment. A lot of alignment strategies have been devised over the years, some are completely independent from the transformation model and optimization strategy and some are not. We will consider the latter case in a spirit of conciseness and also considering that the added modularity of considering each brick separately leads to more flexible registration algorithms.

Matching criteria can be easily classified in two categories by identifying the type of images they regard. When faced with two images that are issued by the same modality,

or the same type of modality, we are in the case of uni-modal image matching and we have naturally a uni-modal matching criterion. In the case where we have two different or different looking modalities, we are in the multi-modal case and we have a multi-modal criterion.

Let us put this intuition in a mathematical setting. The assumption here is that the two images are related by two quantities, the first that we are trying to recover is the transformation $\mathbf{T}$, the second, χ is an intensity mapping that relates the intensities of one image to that of the other:

$$I(\mathbf{x}) = \chi_{\mathbf{x}} \circ J \circ \mathbf{T}(\mathbf{x}) \tag{2.19}$$

χ is unfortunately typically unknown. To cover the most general of cases, we describe the intensity mapping as position dependent: $\chi_{\mathbf{x}}$.

Using this definition, we can give a more precise definition of the uni and multi-modal problems. Uni-modal problems are cases in which $\chi_{\mathbf{x}}$ is usually the combination of the identity function or a linear function of the intensities, independent of the position, with a position independent (assumption of a Gaussian distribution) noise. In multi-modal cases, $\chi_{\mathbf{x}}$ is assumed to be a combination of a non-linear sometimes position dependent (i.e. the same intensity input will give different intensities outputs depending on the position) intensity mapping with an unknown distribution noise.

The multi-modal case is thus arguably a much harder case than the uni-modal one. The uni-modal case was historically the first to be studied, and some of the techniques used for uni-modal cases can be modified to fit multi-modal problems.

2.4.1 Uni-modal Registration

In the case where χ is an identity map or a low level Gaussian distributed noise, using the natural distance in the space of images (namely the euclidean distance) is the most straightforward criterion. This criterion is referred to as the *Sum of squared differences* (SSD):

$$\mathcal{C}\left(I, J \circ \mathbf{T}\right) = SSD\left(I, J \circ \mathbf{T}\right) = \int_{\Omega} \left(I - J \circ \mathbf{T}\right)^2 \tag{2.20}$$

Considered as an alternative to SSD when the noise distribution is deviating from a Gaussian distribution, the L_1 distance in the space of images is the *Sum of Absolute Differences* (SAD):

$$\mathcal{C}\left(I, J \circ \mathbf{T}\right) = SAD\left(I, J \circ \mathbf{T}\right) = \int_{\Omega} \left|I - J \circ \mathbf{T}\right| \tag{2.21}$$

In the more involved case where χ is a linear map with position independent noise, statistical measures of correlation can be considered. One can notably cite the *Normalized Cross Correlation* (NCC)or the *Correlation Coefficient* (CCoef)[Kim 2004] considered when dealing with uni-modal images related by a linear intensity transformation:

$$\mathcal{C}\left(I, J \circ \mathbf{T}\right) = NCC\left(I, J \circ \mathbf{T}\right) = \frac{\int_\Omega \left(I - \overline{I}\right)\left(J \circ \mathbf{T} - \overline{J \circ \mathbf{T}}\right)}{\sqrt{\int_\Omega \left(I - \overline{I}\right)^2 \int_\Omega \left(J \circ \mathbf{T} - \overline{J \circ \mathbf{T}}\right)^2}} \tag{2.22}$$

All the measures presented so far are focusing on pixel intensities and low level statistics for image comparison. While fast, these methods do not take into account higher order information that arises in the neighbourhood of a pixel, and such measures are very sensitive to variabilities between images. In MRI images for instance, two images of the same patient acquired with the same modality can appear very different due to the fact that there is no control on the intensity range in the image and that there is a bias field in the image, which is a position dependant transformation of the intensities.

In the recent years a lot of interest has grow for metrics that are not unimodal per-se but applied to unimodal problems with large variations. The local variations of the bias field are precisely the most difficult to account for since they often imply some nonfunctionalities when looking at the intensity mapping from the intensity space of one image to the intensity space of the other image. To account for this phenomenon, local metrics have been developed [Hermosillo 2002, Cachier 2000, Lorenzi 2013].

In [Shen 2002], the comparison of *attributes vectors* (wich we will also name *feature vectors*) is considered, using image segmentations, each voxel in the image is given attributes depending on the edges next to it. While pixels with similar values arise a lot in images, it is much more seldom to have similar attribute vectors in different locations of the image, which reduces greatly the number of local minima. Following the same intuition, [Xue 2004] extracts multiscale, translation and rotation invariant attribute vectors that are based on Daubechies Wavelets.

2.4.2 Multi-modal Registration

Even though uni-modal criteria can be used with some success in the multi-modal case, when the intensity relationship between images becomes non-linear or position dependant, the uni-modal metrics deliver only deliver sub-par results.

As noted in [Sotiras 2011], two types of approaches can be distinguished for multi-modal registration:

1. The use of statistics based and information theoretic measures that recover non-linear or even non-functional interactions. We will name this kind of measure *Broad multi-modal metrics* as they apply to any two kinds of modality without regards for the specifics of the considered modality.

2. Casting the multi-modal problem into a new uni-modal problem, either by simulating one modality from the other (as seen in Chapter 4), or by embedding both modalities in a common image space (as seen in Chapter 5 and 6). We will name this kind of measure *Modality Specific metrics* as they have been specifically tailored for a pair of modalities.

Broad Multi-Modal Metrics Probably the most commonly used multi-modal registration metric is the Mutual information (MI) [Wells III 1996, Viola 1997, Collignon 1995, Maes 1997]. Mutual information is an information theoretic measure where the objective is to maximize the quantity of mutual information between two images. The quantity of information is described by the Shannon entropy of the image:

$$H(I) = -\sum_{i \in \{0,\dots,m\}} p\left(I\left(\mathbf{x}\right) = i\right) \log p\left(I\left(\mathbf{x}\right) = i\right) \tag{2.23}$$

and

$$H\left(I, J \circ \mathbf{T}\right) = -\sum_{i,j \in \{0,\dots,m\}} p\left(I\left(\mathbf{x}\right) = i, J \circ \mathbf{T}\left(\mathbf{x}\right) = j\right) \log p\left(I\left(\mathbf{x}\right) = i, J \circ \mathbf{T}\left(\mathbf{x}\right) = j\right) \tag{2.24}$$

the Mutual Information is then expressed as:

$$\mathcal{C}\left(I, J \circ \mathbf{T}\right) = MI\left(I, J \circ \mathbf{T}\right) = H(I) + H(J \circ \mathbf{T}) - H\left(I, J \circ \mathbf{T}\right) \tag{2.25}$$

Interestingly enough, the mutual information can be expressed as a divergence between densities using the Kullback-Leibler divergence $D_{KL}(p\|q) = \sum_i p(i) \log(p(i)/q(i))$:

$$MI\left(I, J \circ \mathbf{T}\right) = D_{KL}(p(I)p(J \circ \mathbf{T})\|p(I, J \circ \mathbf{T})) \tag{2.26}$$

In this form, the maximization of the mutual information can be interpreted as the maximization of the dependence between both images since if $p(I)p(J \circ \mathbf{T}) = p(I, J \circ \mathbf{T})$ both images are independently distributed. A survey on mutual information based methods is available in [Pluim 2003].

Normalized Mutual information (NMI) [Studholme 1999] is an improvement over Mutual information, since it is less sensitive to the evaluation of the joint entropy $H\left(I, J \circ \mathbf{T}\right)$ that might be impaired when there is no sufficient overlap between images.

$$\mathcal{C}\left(I, J \circ \mathbf{T}\right) = NMI\left(I, J \circ \mathbf{T}\right) = \frac{H(I) + H(J \circ \mathbf{T})}{H\left(I, J \circ \mathbf{T}\right)} \tag{2.27}$$

NMI was successfully used in numerous studies [Studholme 2001, Blackall 2000, Castellano-Smith 2001, Pluim 2000, Studholme 2000].

Another successful approach to the broad multi-modal metric problem is the correlation ratio (CR) [Roche 1998]. As opposed to mutual information, this measure assumes a functional relationship between images and accounts for non-linearities in the intensity mapping χ. However in addition to mutual information, correlation ratio takes into account the proximity in the intensity space and thus conveys spatial information in the metric. Correlation ratio is expressed as a variant of the correlation coefficient (CCoef):

$$\mathcal{C}\left(I, J \circ \mathbf{T}\right) = CR\left(I, J \circ \mathbf{T}\right) = \frac{Var\left(E[I|J \circ \mathbf{T}]\right)}{Var\left(I\right)} \tag{2.28}$$

Just as Mutual-information is the Kullback-Leibler divergence of two intensity distributions, other types of divergences and entropies have been proposed.

In [Pluim 2004], Pluim *et al.* look at the whole class of information theoretic measures from which mutual information is drawn. This class is named the class of the f–*Divergence of probability distributions*. The f–divergence (fD) is a generalization of the Kullback-Leibler divergence and is defined as:

$$\begin{cases} f\left(P \| Q\right) & = & \sum_i q_i f\left(\frac{p_i}{q_i}\right) \\ q_i f\left(\frac{p_i}{q_i}\right) & = & \begin{cases} 0 & \text{if } p_i = q_i = 0 \\ p_i \lim_{x \to \infty} \frac{f(x)}{x} & \text{if } p_i > 0, q_i = 0 \end{cases} \end{cases}$$

Again, the f–divergence can be used for comparing two images by maximizing the statistical dependence of the distributions of the two images:

$$fD\left(I, J \circ \mathbf{T}\right) = f(p(I)p(J \circ \mathbf{T}) \| p(I, J \circ \mathbf{T})) \tag{2.29}$$

the use of some well studied f-divergences was shown to yield more accurate results for CT-MR registration and MR-PET registration.

Addressing the shortcomings of mutual information, namely the lack of convexity and the lack of symmetry of the measure, the *Jensen-Rényi Divergence* (JRD) was proposed

[He 2003, Hamza 2003]. The Jensen-Rényi Divergence is based on a generalization of the Shannon entropy that is the *Rényi entropy*:

$$\begin{aligned} H_\alpha(I) &= H_\alpha(p\left(I\left(\mathbf{x}\right)=1\right),\ldots,p\left(I\left(\mathbf{x}\right)=m\right)) \\ &= \frac{1}{1-\alpha}\log\sum_{i\in\{0,\ldots,m\}}\left(p\left(I\left(\mathbf{x}\right)=i\right)\right)^\alpha \end{aligned} \tag{2.30}$$

Where $\alpha \in (0,1)$ and when α tends to 1, it can be shown that H_α converges to the Shannon entropy H. Rényi entropy is concave. Using this definition and defining $p_{ij} = p\left(J\circ\mathbf{T}\left(\mathbf{x}\right)=j|I\left(\mathbf{x}\right)=i\right)$, the Jensen-Rényi Divergence between two images is defined as:

$$\begin{aligned} \mathcal{C}\left(I,J\circ\mathbf{T}\right) &= JRD\left(I,J\circ\mathbf{T}\right) \\ &= H_\alpha\left(\sum_i w_i p_{i1},\ldots,\sum_i w_i p_{im}\right)-\sum_i w_i H_\alpha\left(p_{i1},\ldots,p_{im}\right) \end{aligned} \tag{2.31}$$

in this formulation, if $w_i = p\left(I\left(\mathbf{x}\right)=i\right)$ and $\alpha = 1$ then JRD becomes the Mutual information. In [Hamza 2003], $w_i = 1/m$ is shown to yield better results.

In [Neemuchwala 2002] a more simple version of the JRD is used, the α*–Jensen Difference* (αJD):

$$\begin{aligned} \mathcal{C}\left(I,J\circ\mathbf{T}\right) &= \alpha JD\left(I,J\circ\mathbf{T}\right) \\ &= H_\alpha(\beta J\circ\mathbf{T})+(1-\beta)I)-\beta H_\alpha(J\circ\mathbf{T})-(1-\beta)H_\alpha(I) \end{aligned} \tag{2.32}$$

Setting $\alpha = 1$ and $\beta = 1/2$ gives again Mutual-information. In this work, a minimum spanning tree is used to estimate the Rényi entropy, which yields significantly lower memory usage and time complexity.

The major problem of the information theoretic measures presented here is that they only take into account single pixel probabilities. This presents the problem of the validity of the measure since you can randomly permute the pixels in one image and still have the same entropy for instance (note that the joint entropy does not suffer from this problem) [Rueckert 2000]. To circumvent the problem, in [Rueckert 2000], higher order mutual informations are considered with 4D-histograms that take into account the neighbourhood information. Even though this method proves to yield better results than traditional mutual information, this kind of technique suffers from the curse of dimensionality. When the number of dimensions increases, the number of samples for each dimension has to increase

accordingly to keep the statistical relevance of the criterion. Measures have been taken to counteract this problem, like reducing the number of histograms bins or taking random lines [Bardera 2006, Rueckert 2000], but the problem is inherent to the method and even higher orders cannot straightforwardly be considered.

The introduction of local context in information theoretic measures has also been done by means of computations of local mutual informations [Karaçali 2007, Loeckx 2010, Zhuang 2011, Russakoff 2004]. This kind of methods, since they measure locally the images can handle much more efficiently position dependent intensity changes such as the bias field in MRI images and position dependent noise.

Modality Specific Metrics There is a natural transition between the information theoretic metrics we have seen earlier and Modality specific metrics. We have seen that most information theoretic metrics can be expressed as a divergence or a distance between two intensity distributions. Most of the time for broad multi-modal metrics, those two distributions are the joint distribution of intensities in both images and the product of intensity distributions in both images.

Now let us assume that we have access to a pair of perfectly registered images, before the registration (I_l, J_l). The idea behind modality specific metrics is to use the prior knowledge of the already aligned images and drive a better registration using this knowledge. The first attempts to use this knowledge have been using the aforementioned distribution divergences. Indeed, since we have a training pair of registered images, we can for instance evaluate their joint distribution of intensities: $p_{\text{learned}}(I_l, J_l)$. Now a reasonable objective can be that of minimizing the divergence between the learned distribution and the actual distribution, for instance with the Kullback-Leibler Divergence(KLD) [Chung 2002]:

$$\mathcal{C}\left(I, J \circ \mathbf{T}\right) = KLD\left(I, J \circ \mathbf{T}\right) = D_{KL}\left(p_{\text{learned}}(I_l, J_l) \| p(I, J \circ \mathbf{T})\right) \tag{2.33}$$

Following the same idea, in [Guetter 2005], the KLD is evaluated in conjunction with mutual information between the source and the deformed target, in an attempt to benefit from both worlds and to not rely entirely on the learned distribution, in cases where the new pair is too different from the training pair.

Similarly to [Guetter 2005], the same rationale of mixing both worlds has been investigated in [Liao 2006], where a divergence different form KLD is beeing used, the *Jensen-Shannon Divergence* (JSD):

$$\begin{aligned}\mathcal{C}\left(I, J\circ\mathbf{T}\right) &= JSD\left(I, J\circ\mathbf{T}\right) \\ &= \frac{1}{2}D_{KL}\left(p_{\text{learned}}(I_l, J_l)\left\|\frac{1}{2}\left(p(I, J\circ\mathbf{T}) + p_{\text{learned}}(I_l, J_l)\right)\right.\right) \\ &+ \frac{1}{2}D_{KL}\left(p(I, J\circ\mathbf{T})\left\|\frac{1}{2}\left(p(I, J\circ\mathbf{T}) + p_{\text{learned}}(I_l, J_l)\right)\right.\right)\end{aligned} \tag{2.34}$$

The JSD has the advantage over KLD of being symmetric and has all the mathematical properties of a distance metric. Also JSD is bounded by $\log 2$, which makes it easier to normalize and balance against the mutual information term.

A more recent set of techniques consider this problem in a two steps problem. First both images are projected into the same image space, in which the projected representations of the images are considered comparable using simple measures uni-modal measures (e.g. SSD). Then the registration is conducted in this uni-modal space using the simple measure.

A first instance of this framework is when one image is projected in the image space of the other. In essence this is a simulation of one modality from the other. Given the image in one modality, one tries to guess how it would look had it been acquired by the other modality. This approach will be discussed in details in chapter 4.

A second instance of the modality specific metric problem, is when we project in a first step both modalities in a common space, and then perform the comparison of the images in this new space. For this type of methods we will refer to *common space embedding*. To some extent, all the information theoretic measures used with learned joint probabilities can be seen as common space embedding. Indeed in these methods, we first compute the joint probability of images which in turns is a projection in the space of the joint histogram. Then comparison is performed against another joint probability.

As we have already seen , when making a joint histogram, only pixel-wise information is taken into account. Other approaches have been taken where images from both modalities are first transformed into an image that does not focus on pixel intensity information but rather on intensity variation information, or even local statistical information. The rationale behind these methods is that since we are taking images of the same type of tissue, the intensity information may be different but describe the same reality. This intuition is revealed by projecting the images into a space in which intensity information is not taken into account anymore.

This is the case in [Maintz 2001] where images are first transformed with mathematical morphology operators such as erosions and dilatations, openings and closings. Then

the transformed images are rigidly registered using cross-correlation. An example of the transformed images is shown in figure 2.1. The resulting transformation still loses a lot of information from the original image and some artefacts are added due to misinterpretations of the noise. While this method is reported to perform relatively well for rigid registration, it is hard to see it performing equally well in the case of deformable registration.

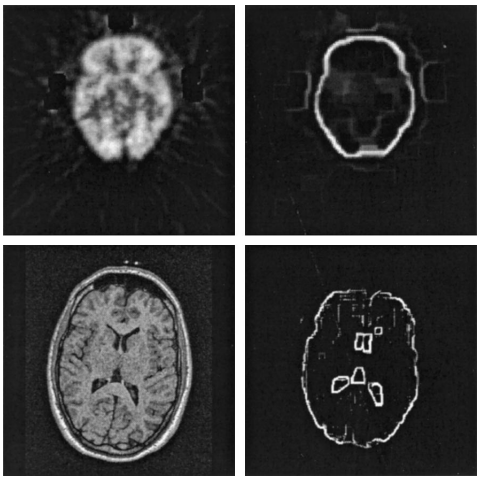


Figure 2.1: Figure extracted from [Maintz 2001]: Top row: left, original PET image, right, transformed image, Bottom row: left, original MR image, right transformed MR

More recently in [Wachinger 2011], the same kind of approach is taken but the pixel intensity information is more exploited. Two kinds of transformation are discussed. In both cases patches of the images are extracted densely, so for each pixel position in the image there is a corresponding patch. Then each patch undergoes a transformation that maps it to a scalar value that is representative of the patch. The transformations that are considered respect two rules, the first one is the locality preservation, meaning that if two patches are similar in terms of intensity (under the ℓ_2 norm for instance), then their transformations must be similar. The second condition, named structural equivalence ensures that similar patches from both modalities are mapped to similar transformation values. The first transformation studied that satisfies both these rules is the local Shannon entropy of the patch. Entropy images of MRI images can be seen in figure 2.2.

The second transformation involves manifold learning, Laplacian eigen maps are used to define the transformation. For each set of patches for both images, an adjacency graph is first created, the proximity of the patches in this graph is proportional to their euclidean distance. Then using Laplacian eigen maps a unidimensional embedding is constituted in which the proximity relationships of the graph are conserved. This unidimensional mapping is in effect a scalar map from the patches and represents the image. Figure 2.3

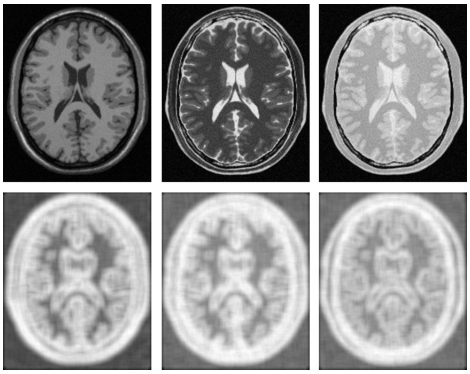


Figure 2.2: Figure extracted from [Wachinger 2011]: Top row: original images, from left to right T1, T2 and PD MRI images, bottom row entropy images

depicts the difference between entropy images and Laplacian images. On can see that the intensity mapping is fairly consistent across MRI modalities. One can still question the efficiency of the method when considering harder pairs of modalities in which the embeddings are much less comparable.

Lastly, [Lee 2009] paves the way for multi-modal metric learning for image registration. Using a training data set of perfectly aligned images images, similar patches are extracted in similar locations. The similarity is learned on these patches $(\mathbf{x}_i, \mathbf{y}_i)$. Using the properties of kernels, the patches are projected in feature spaces of possibly infinite dimensions. The projections are compared in the feature spaces by means of a learned similarity function:

$$s(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} \sum_{\overline{\mathbf{y}}} \alpha_i \psi(\mathbf{x}_i, \mathbf{x}) \cdot (\psi'(\mathbf{y}_i, \mathbf{y}) - \psi'(\overline{\mathbf{y}_i}, \mathbf{y})) \tag{2.35}$$

where ψ and ψ' are kernels, here, Gaussian kernels are considered; This measure is learned by means of a quadratic program that is in essence a modified Support Vector Machine:

$$\begin{aligned} &\min_{w,\xi} \frac{1}{1}\|w\|^2 + C\sum_{i=1}^{n} \xi_i \\ &s.t. \quad s(\mathbf{x}_i, \mathbf{y}_i) - \max_{\mathbf{y}, \mathbf{y} \neq y_i} s(\mathbf{x}_i, \mathbf{y}) \geq 1 - \xi_i \quad \forall i \\ &\xi \geq 0 \quad \forall i \end{aligned} \tag{2.36}$$

S_i then represents the set of the most violated active constrains for each instance i. This measure, although performing well for rigid registration has not been yet demonstrated

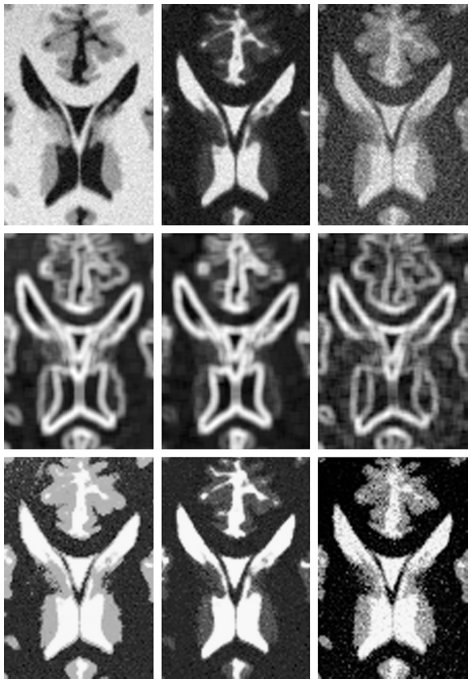


Figure 2.3: Figure extracted from [Wachinger 2011]: Top row: Original images, Middle row: entropy images, Bottom row: Laplacian images

for deformable registration. Although timing is not discussed in the paper, the estimation of the similarity criterion might take time, since the solving of a quadratic problem is considered. With the added consideration that cross validation will be needed to find the parameters of the kernels, the learning time of the similarity might be problematic.

Chapter 3

Preliminary: Feature extraction and Gabor Features

In the next chapters, the focus will be on finding an appropriate similarity criterion for comparing two images issued by different modalities. The leading idea is in this work is to have a *dense* criterion, meaning that we want a criterion that is defined for each and every pixel or voxel (we will always refer to pixel, even in 3D for the sake of clarity) location in the image. The methods that we will use to infer the criterion are based on machine learning methods, which from a set of training data, extract a pattern from the available information and allow us to reproduce this pattern on unforeseen examples. The quality of the reproduced pattern greatly depends on the quality of the input information.

Often times, working with the pixel information, *i.e.* intensity, is not a viable option. Medical images are corrupted by noise and image intensities are by definition sensitive to noise, ideally we would like to extract information at the pixel position that is true to the underlying image information not corrupted by noise. More generally, we would like information that is the less sensitive to image deteriorations as possible and the more close to the true image information as possible. Another great disadvantage of the pixel intensities is that they don't convey much information about the image, and based on this sole information the learned criterion might be performing poorly. Ideally we would like to have for each and every pixel location a sense of the neighbouring pixels, this additional information will help us distinguish between pixels that might have the same intensities but different surroundings, and will also improve the learning for the same reason.

Instead of extracting only the intensity information at the pixel location, now we are going to extract *features* or *feature vectors*, while the intensities lie in a one dimensional space (usually $\mathbb{R}$), feature vectors lie in a d dimensional space $\mathcal{X} \subset \mathbb{R}^d$. This obviously leads to computational overhead that is only acceptable thanks to the steady advances in computing power in the recent years. Nonetheless, the methods that we will use will have

to be able to scale well with large d.

3.1 Feature extraction framework

Let us now consider an image $I : \Omega \subset \mathbb{R}^{\{2,3\}} \longrightarrow \mathcal{I} \subset \mathbb{R}$, here $\mathcal{I}$ denotes the intensity space and is a feature space of dimension 1. We can define a feature extraction function $\boldsymbol{\pi}$ that acts on an image I and a pixel position $\mathbf{x}$, and extracts a feature vector of I in position $\mathbf{x}$:

$$\boldsymbol{\pi} : \begin{cases} \mathcal{I} \times \Omega & \longrightarrow & \mathcal{X} \subset \mathbb{R}^d \\ (I, \mathbf{x}) & \longmapsto & \boldsymbol{\pi}\left(I, \mathbf{x}\right) \end{cases} \tag{3.1}$$

Usually, $\boldsymbol{\pi}$ is parametrized with values that reflect the extent of the neighbourhood considered or the level of invariance to various image artefacts.

As an example let us now see the most commonly used feature extraction function, the patch extraction function. A patch is a simply a vector of the intensities that are encountered in a square (or cubic) vicinity of the position of interest as demonstrated in figure (3.1), for a 3×3 patch.

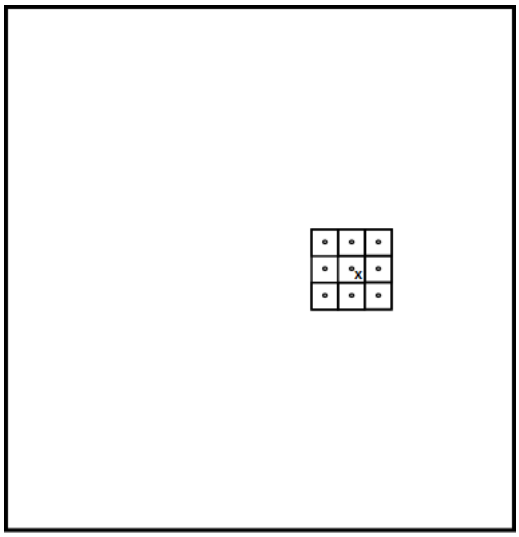


Figure 3.1: Extraction of a 3×3 feature patch $\boldsymbol{\pi}_{3,3}\left(I, \mathbf{x}\right)$ at position $\mathbf{x}$

The patch extraction function can also be viewed as a image filtering process. In a general case, let us consider the $m \times n$ patch extraction function $\boldsymbol{\pi}_{m,n}$. If we consider the filter bank:

$$\Pi^k_{m,n} \in \mathcal{M}_{m,n} : \begin{cases} \left[\Pi^k_{m,n}\right]_{i,j} = 1 & \text{if } i + mj = k \\ \left[\Pi^k_{m,n}\right]_{i,j} = 0 & \text{otherwise} \end{cases} \tag{3.2}$$

where $\mathcal{M}_{m,n}$ is the space of matrices $m \times n$. Then we have:

$$\left[\boldsymbol{\pi}_{m,n}\left(I, \mathbf{x}\right)\right]_k = \Pi^k_{m,n} \star I\left(\mathbf{x}\right), \quad k \in \{1, \ldots, mn\} \tag{3.3}$$

Feature extraction has been widely used in the Computer Vision community and notably for feature detection in images [Mikolajczyk 2005, Schmid 2000], yet the literature is much scarcer in the case of medical images. This is mainly due to the problem of the computation times that are prohibitive when considering volumes instead of planar images. However some very popular computer vision methods have made their way through medical imaging. Built in invariances in feature descriptors are paramount for medical image analysis, where images issued by the same modalities show a lot of variability due to the lack of consistensy in image intensities, heavy image noise and image artifacts.

Looking for locally invariant descriptors is closely related to texture analysis, a good introduction to texture analysis can be found in [Tuceryan 1993]. Texture analysis among others can be used for [Tuceryan 1993] **t**exture classification, **t**exture synthesis, **s**hape from texture and **t**exture segmentation applications. Here our interest will be on texture segmentation, since it often involves finding texture descriptors for each pixels and then clustering the descriptor space.

Texture analysis for texture segmentation methods, can roughly be divided into two categories:

- Statistical methods
- Signal Processing methods

3.1.1 Statistical Feature descriptors

Statistical methods compute statistical measures on windows densely sampled in the image, the main disadvantage of these methods is the high computational complexity and the computation window size, which is in most cases too big (such as 19×19 pixels), thus yielding very slow computation times.

Haralick Texture Features (GLCM) Statistical methods were among the first methods considered for texture analysis. They were introduced in [Haralick 1973]. They introduced the now broadly used notion of gray level co-occurrence matrix (GLCM). These features where presented in 1973, tested on a *PDP 15/20* prehistoric computer, and are still widely used.

The GLCM measure the occurrence frequency of two intensity levels i and j at a given distance d and in a given orientation θ in a window W of size $L_x \times L_y$; intensity levels come from a sub-sampling of the intensity range. To account for rotation, it is proposed [Haralick 1973] to use the average and the range over the orientations. 14 features in total are presented, among those four of them are invariant to monotonic gray scale transformations, those are the most used features in the literature. Haralick Texture features have been successfully used in a medical context in [Pescia 2008].

Local Binary Patterns (LBP) They have been introduced in [Ojala 2002], where is defined a local operator that can be evaluated at each pixel of the image and returns a scalar value. This value is designed to be gray-scale, scale and rotation invariant.

If we assume that the pixel where we compute the LBP is at $(0, 0)$ and its value is g_c, then let us define the values g_p given by the intensities at $(-R\sin(2\pi p/P), R\cos(2\pi p/P))$ then:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p \tag{3.4}$$

where

$$s(x) = \begin{cases} 1, x \geq 0 \\ 1, x < 0 \end{cases} \tag{3.5}$$

Then the rotation invariance is achieved through the minimum in all the circular rotations:

$$LBP^{ri}_{P,R} = \min\{ROR(LBP_{P,R}, i) \mid i = 0, 1, \ldots, P-1\} \tag{3.6}$$

Some of the shortcommings of LBP shortcommings are pointed out in [Zhou 2008]

- It is sensitive to noise
- Only uniform parts of the texture are accurately described, leaving much information behind.

In [Zhou 2008], the proximity of the non-uniform patterns to the uniform ones is learned in order to take into account the non uniformity. Examples of use in medical applications include [Setia 2006].

Ranklets Introduced in [Smeraldi 2002], *Ranklets* provide an interesting way of merging the filtering schemes and local statistics for local descriptor design. A thorough description of the process is provided in [Smeraldi 2003]. Ranklets are a complete family of multiscale, orientation selective features, based on the *Haar wavelet*. They use the

wilcoxon statistic on ranks, so the completeness here, refers to the ability, given the complete family of features, to recover the full pixel rank of the image, and not the pixel values since they are discarded from the beginning.

Let $r(I^W(x))$ denote the rank in the ordering of pixel intensities, of the sample $I(x)$ in the window W. The window size is ruled by the Haar wavelet support. The Haar wavelet window is divided into two partitions later called *treatment* (T) and *control* (C), according to the sign of the Haar wavelet (see figure 3.2).

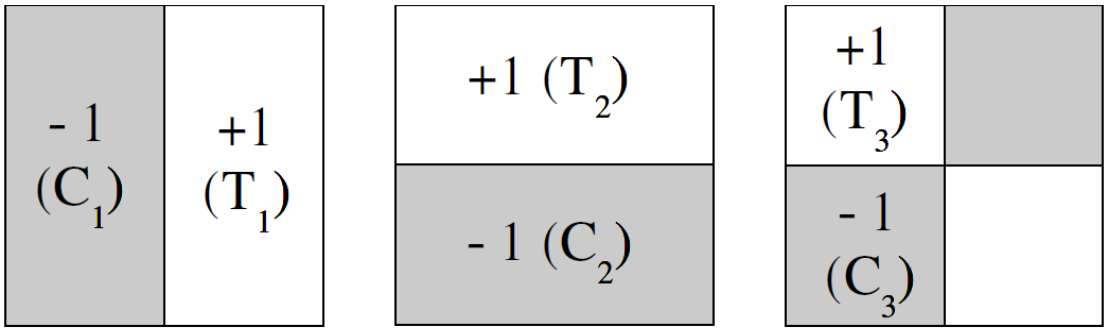


Figure 3.2: Figure extracted from [Smeraldi 2002]: Haar ranklets partitionning

Defining $\mathcal{W}_s^j$ as:

$$\mathcal{W}_s^j = \sum_{x \in T_j} r(I^W(x)) \tag{3.7}$$

The value value of the ranklet $\mathcal{R}^j$ is given by:

$$\mathcal{R}^j = 2\frac{\mathcal{W}_s^j - (N/2+1)N/4}{N^2/4} - 1 \tag{3.8}$$

Following the multiscale design of Haar wavelets, ranklets are multiscale.

A review of Ranklets performance is given in [Masotti 2008], they show that ranklets are very robust to gray scale variations and show that the average value over the vertical, horizontal and diagonal ranklet images, yield an almost completely invariant descriptor for $90°$-rotations. Ranklets have successfully been used in medical applications in [Masotti 2006].

3.1.2 Signal Processing methods

A good introduction to texture classification and segmentation can be found in [Randen 1999]. A comparative study is held between the various filtering methods for texture classifica-

tion. The basic assumption for most filtering approaches is that the energy distribution in the frequency domain identifies a texture. The frequency domain is thus analyzed using different filters and their processed outputs forms feature vectors for each and every pixel in the image. The filtering output is usually not processed as is, but is going through a local energy function first to lessen the impact of noise and to accentuate the discrimination.

Steerable Filters Intoduced in [Freeman 1991], Steerable filters allow to get the response of the filter for virtually any orientations with a composition of responses to a base of filters. Theoretical arguments are given to find the conditions under which any function $f(x, y)$ steers, *i.e.* , when it can be written as a linear sum of rotated versions of itself.

$$f^{\theta}(x, y) = \sum_{j=1}^{M} k_j(\theta) f^{\theta_j}(x, y) \tag{3.9}$$

The response of filters to different orientations leads to very efficient detections [Jacob 2004] and was recently used in a medical setting for guide wire detection [Honnorat 2010].

Monogenic Signal The monogenic signal was introduced by [Felsberg 2001] as an extension of the *Analytic signal* to a N-dimensional space. The analytic signal is used to decouple the local magnitude and the local phase of a signal in 1D, as shown in figure 3.3.

In 2D, not only the local phase and the local energy are accessible but also the local orientation (see figure 3.4). They are accessible through a filtering process using the filters:

$$\begin{aligned} H_1(\omega_1, \omega_2) &= \frac{i\omega_1}{\sqrt{\omega_1^2 + \omega_2^2}} \\ H_2(\omega_1, \omega_2) &= \frac{i\omega_2}{\sqrt{\omega_1^2 + \omega_2^2}} \end{aligned} \tag{3.10}$$

Then the *local amplitude*, *local phase* and *local orientation* respectively A, φ, θ:

$$\begin{aligned} A(x_1, x_2) &= \sqrt{f^2 + (h_1 \star f)^2 + (h_2 \star f)^2} \\ \varphi(x_1, x_2) &= \text{acos}\left(\frac{f(x_1, x_2)}{A(x_1, x_2)}\right) \\ \theta(x_1, x_2) &= \text{atan2}\left(h_2 \star f, h_1 \star f\right) \end{aligned} \tag{3.11}$$

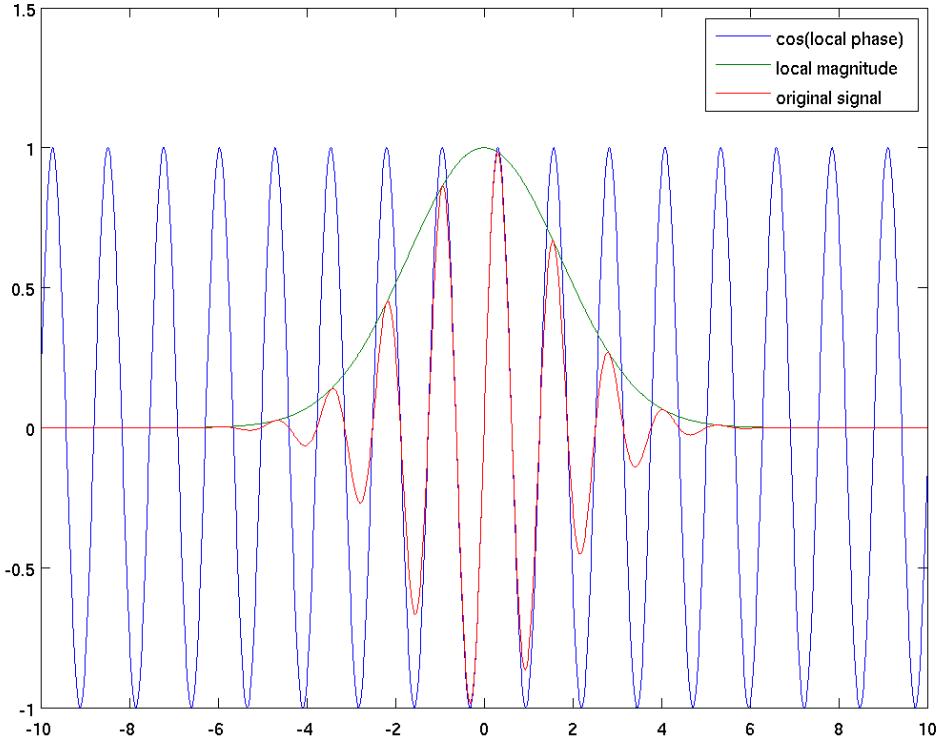


Figure 3.3: Decoupling of a sine wave moduled by a decaying exponential with the *analytic signal*

Figure 3.4: figure extracted from [Kokkinos 2008]: Monogenic Signal example

The local phase and the local orientation are particularly interesting in our case, since they present *local gray scale invariance*. Examples of use in medical imaging include:

[Pan 2006, Grau 2007, Mellor 2004].

Wavelets Some of the computer vision favorite image texture descriptors have crossed through medical image processing. It is the case of the very popular wavelet bases that provide a way to decompose an image in frequency and space on a complete base [Mallat 1999] that have been used in a medical setting for instance in [Xue 2004].

Scale Invariant Feature Transform (SIFT) features [Lowe 2004] are probably the most used features in the computer vision community currently. The feature consists in a normalized histogram of gradient orientation and magnitude that is computed for each point and weighted according to their distance to the point of interest, this makes the feature detector scale invariant and robust to changes in the image orientation. Variants of the SIFT descriptors have been proposed, where the focus is set on making a very fast feature extractor [Juan 2009]. The application of SIFT features to medical images can for instance be found in [Han 2010].

3.2 Gabor features

Gabor filters are very popular in image processing and have been widely used in medical applications. Named after *Denis Gabor*, the filter is in essence a gaussian filter modulated by a sine wave. In 2D, many parameters can be set, such as the width of the gaussian (2 parameters), the frequency and the phase of the sinusoid and the orientation of the wave (3 parameters).

The 2D gabor function $g(x, y)$ is defined as:

$$g(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right) + i\,2\pi W x\right] \tag{3.12}$$

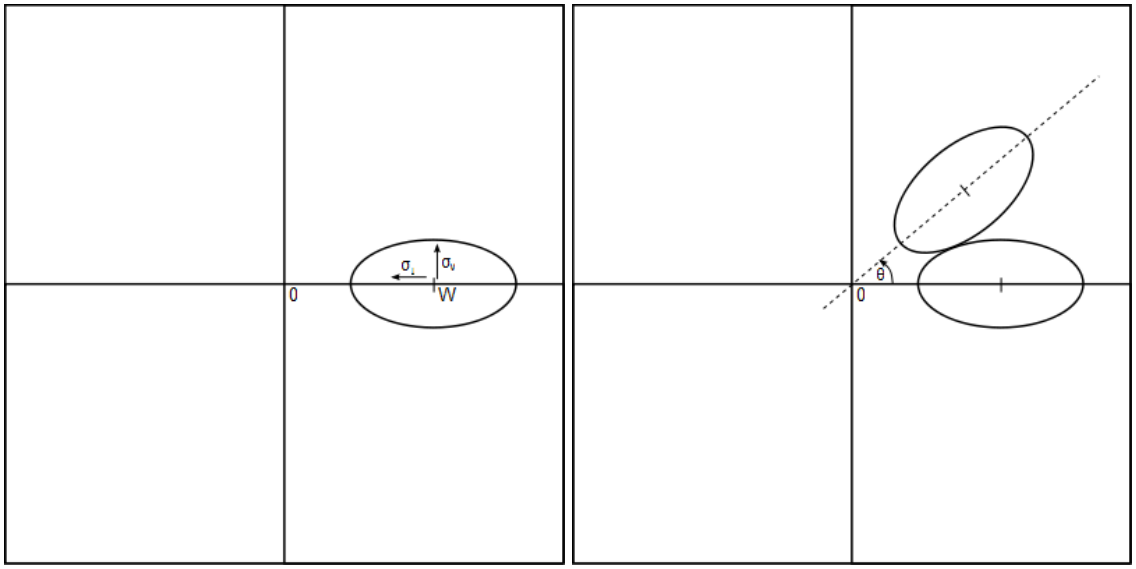
Parameters σ_x and σ_y set the width of the gaussian filter, and paremeter W sets the frequency of the sine wave. It is interesting to take a look at the Fourier transform of the gabor filter $G(u, v)$:

$$\begin{cases} G(u,v) &= \exp\left[-\frac{1}{2}\left(\frac{(u-W)^2}{\sigma_u^2} + \frac{v^2}{\sigma_v^2}\right)\right] \\ \sigma_u &= \frac{1}{2\pi\sigma_x} \\ \sigma_v &= \frac{1}{2\pi\sigma_y} \end{cases} \tag{3.13}$$

if we only look at the half peak value level set (an ellipse in which more than 75% of the energy of the gaussian is contained), then we get an ellipse with parameters proportionate to σ_u and σ_v, centred in W on the axis of u (see figure 3.5(a)).

In actuality the gabor filter covers all the Fourier space, but its energy is concentrated around W and 75% is in the half peak level set. Filtering with this gabor filter will mainly give a response in the frequency area covered by the ellipse in figure 3.5(a). Ideally we would like all the Fourier space to be covered, so we are going to apply rotations and scaling to g (equivalently to G) to be able to cover all the space:

$$g_{\lambda,\theta}(x,y) = \lambda g\left(\lambda\left(x\cos\theta + y\sin\theta\right), \lambda\left(-x\sin\theta + y\cos\theta\right)\right) \tag{3.14}$$



(a) (b)

Figure 3.5: Visualization of the half peak ellipse of a gabor filter in frequency space. (a) viszualization of the ellipse parameters, (b) rotation by parameter θ.

In essence, all these parameters are linked, and the condition that the set of gabor filters (a *filter bank*) should pave the frequency space as much as possible reduces the parameter space. In [Manjunath 1996], a discrete parametrization to the filter bank is given that allows to only keep two parameters, the number of orientations K and the number of scales S and set 2 frequencies that are the upper and lower frequencies of interest, U_h an U_l respectively. Then solving for the original parameters such that each half peak ellipse in the filter bank touch as shown in figure 3.6, yields:

$$W = U_h \tag{3.15}$$

$$\lambda = a^{-m} = \left(\frac{U_h}{U_l}\right)^{-\frac{m}{S-1}} \quad m \in \{0, \ldots, S-1\} \tag{3.16}$$

$$\theta = \frac{n\pi}{K} \quad n \in \{0, \ldots, K-1\} \tag{3.17}$$

$$\sigma_u = \frac{(a-1)\,U_h}{(a+1)\,\sqrt{2\ln 2}} \tag{3.18}$$

$$\sigma_v = \tan\left(\frac{\pi}{2K}\right)\sqrt{\frac{U_h^2}{2\ln 2} - \sigma_u^2} \tag{3.19}$$

Working with Multi-Modal MRI images and PET-CT images, we found that $U_h = 0.2$ and $U_l = 0.05$ yields the best metric learning generalization results.

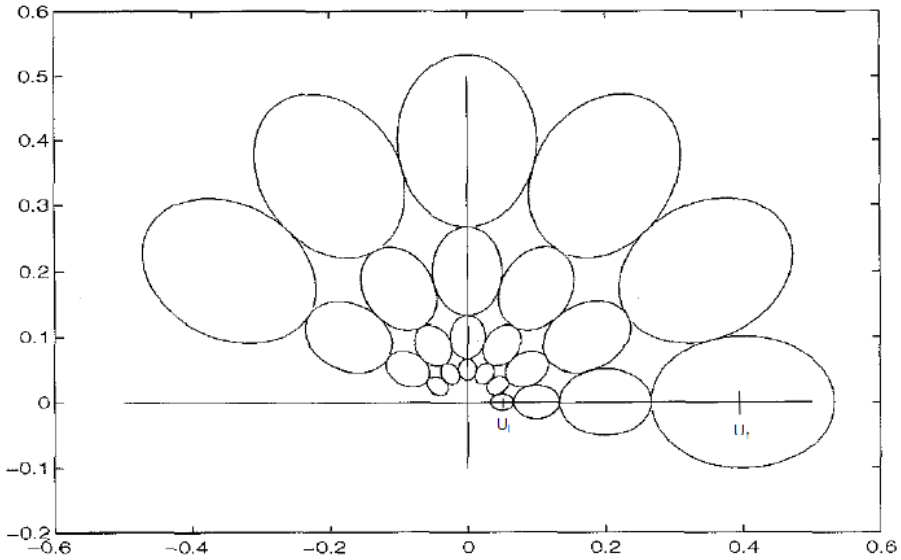


Figure 3.6: Figure extracted from [Manjunath 1996], a paving of the Fourier space where the half peak magnitudes touch to minimize the gaps as well as the redundancy. Here $K = 6$ and $S = 4$

Computation of the Gabor features is usually very slow, especially in the low frequencies were the extent of the filter is very large. This becomes an even greater problem when

considering 3D images. In this thesis we used the approach found in [Zhan 2003] where instead of computing a set of 3D gabor features, two orthogonal sets of 2D Gabor features, as shown in figure 3.7. Using a second gabor 2D function $h(y, z)$ defined as:

$$h(y,z) = \frac{1}{2\pi\sigma_y\sigma_z} \exp\left[-\frac{1}{2}\left(\frac{y^2}{\sigma_y^2} + \frac{z^2}{\sigma_z^2}\right) + i\,2\pi W y\right] \tag{3.20}$$

And $h_{\lambda,\theta}(y, z)$ in the same way as in equation (3.14) we compute the 3D Gabor Features by first filtering by g then by h.

Such Gabor features were successfully used in [Sotiras 2010, Ou 2009, Ou 2011, Wang 2010, Xiang 2011, Xiang 2011, Xiang 2012, Parisot 2012a].

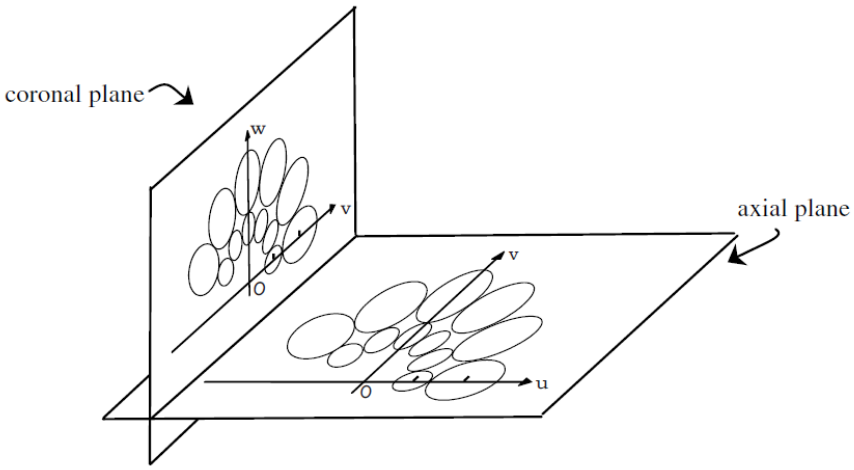


Figure 3.7: Figure extracted from [Zhan 2003], instead of computing 3D gabor features, two orthogonal sets of Gabor features are considered.

3.3 Fast Infinite Impulse Response Anisotropic Gabor filtering

When considering the convolution with Gabor filters, the filter support while infinite can be large if we only consider the components above some threshold. This makes the computations of the gabor filter usually computationally very demanding, especially when we consider a filter bank of 128 filters (2×16 orientations $\times 4$ scales), on 3D images. Here

we used the recent advances in the field of Infinite impulse response filter to build a fast anisotropic Gabor filter bank.

Let us first consider and decompose the expression of the Gabor function:

$$\begin{aligned} g_{\lambda,\theta}(x,y) &= \frac{\lambda}{2\pi\sigma_x\sigma_y}\exp\left[-\frac{\lambda^2}{2}\left(\frac{(x\cos\theta+y\sin\theta)^2}{\sigma_x^2}+\frac{(-x\sin\theta+y\cos\theta)^2}{\sigma_y^2}\right)\right] \\ &\cdot \exp\left[i2\pi\lambda W\left(x\cos\theta+y\sin\theta\right)\right] \\ &= w_{\lambda,\theta}(x,y)\, c_{\lambda,\theta}(x,y) \end{aligned} \tag{3.21}$$

where $w_{\lambda,\theta}$ is the gaussian filter and $c_{\lambda,\theta}$ is the sine wave modulation. Following [Bernardino 2006] if we notice that for all $k,l \in \mathbb{N}^2$:

$$\begin{aligned} c_{\lambda,\theta}(x-k,y-l) &= \exp\left[i2\pi\lambda W\left((x-k)\cos\theta+(y-l)\sin\theta\right)\right] \\ &= \exp\left[i2\pi\lambda W\left((x\cos\theta+y\sin\theta)-(k\cos\theta+l\sin\theta)\right)\right] \\ &= c_{\lambda,\theta}(x,y)\, c^{\star}_{\lambda,\theta}(k,l) \end{aligned} \tag{3.22}$$

where $c^{\star}$ is the complex conjugate of c. Then we can rewrite the convolution:

$$\begin{aligned} w_{\lambda,\theta}\, c_{\lambda,\theta} \star I(x,y) &= \sum_{k,l} I(k,l)\, w_{\lambda,\theta}(x-k,y-l)\, c_{\lambda,\theta}(x,y)(x-k,y-l) \\ &= c_{\lambda,\theta}(x,y)\sum_{k,l} I(k,l) c^{\star}_{\lambda,\theta}(k,l)\, w_{\lambda,\theta}(x-k,y-l) \\ &= c_{\lambda,\theta}(x,y)\cdot\left[\left(I\cdot c^{\star}_{\lambda,\theta}\right)\star w_{\lambda,\theta}(x,y)\right] \end{aligned} \tag{3.23}$$

This last equation is very interesting since it replaces the convolution with a gabor function with a modulation by a sine wave, followed by a convolution with a gaussian and demodulation by a sine wave. Convolution with a gaussian filter is a very active and researched field and, current algorithms provide very fast implementations even for anisotropic gaussian functions. Modulations and demodulations are simple multiplications and do not give significant computational overhead. More over, when we treat a set of images with the same dimensions (as it is the case for the slices of a 3D volume), we only need to compute $c_{\lambda,\theta}$ once, and only the fast gaussian filtering is required for each image.

In [Bernardino 2006], only the isotropic case is considered, indeed the computation of fast infinite impulse response (IIR) isotropic gaussian filters has been used for a long time

(see [Deriche 1993] for instance). Infinite impulse response filters, are not based on the convolution with a discrete window but give a response densely for each and every pixel in the image, without regards for the size of the window since the filter is considered to have an infinite support.

Recently, fast computations of IIR anisotropic gaussian filters has been proposed in [Geusebroek 2003], the implementation of this method[1] has been used to make fast IIR anisotropic Gabor filtering in this thesis.

Lastly we would like to point out that since we treat each slice independently this process is highly parallelizable.

3.4 Building invariances for Gabor filter banks

Gabor filters are natively robust to noise and perfectly adapted to medical images. But other invariances can be brought to the Gabor filtering framework with minimal computational overhead. The first very common invariance that can be brought to the filter is the invariance to intensity shifts, if we denote by $\tilde{g}$ the new filter, then intensity shift invariance is:

$$\forall x, y \quad \tilde{g} \star (I(x, y) + c) = \tilde{g} \star I(x, y) \tag{3.24}$$

This is simply done by setting to 0 the 'DC term' (zero frequency term) of the Fourier transform of the filter. Only setting to zero the DC term, leads to artefacts such as ringing since the cut off in frequency space is too harsh, instead setting to zero this term (which in turn is only removing the mean of the filter), we remove a gaussian function with peak value equals to the zero value in frequency to have a smooth cut-off.

Scale and rotation invariances can also be crucial to a good performance in registration, since most registration algorithms perform local scalings and rotations. Fortunately, the gabort filter bank provides a sampling of the rotation and scale space, reorganizing each extracted feature vectors into a 2D array with respect to scale and orientation, it is easy to see that a rotation of the image around the pixel position of the feature vector amounts to a translation in the array, and the same with the scale. Using this fact and following [Kokkinos 2008], we compute the Fourier Transform Modulus of the 2D array which is itself invariant to translations, thus removing the dependance of the feature vector to translations and scalings.

[1] Source code available at http://www.science.uva.nl/ mark/downloads/anigaussm.zip at the time of writing

3.5 Experiment

Let us have a look here at the differences between patch based features and Gabor based features. We took a brain image and only consider half of the image, the other half of the image is a symmetrized version of the first half. The second half is corrupted by noise as can be seen on the top row of figure 3.8. Now we extract a feature vector (either patch or Gabor) at one position in this image (either of the red squares in the top row of figure 3.8), and compute the mahalanobis distance of this feature vector with all the feature vectors extracted in the image. This results in a *distance map*, a data set that is the size of the image and represents a map of all the features vectors to the one considered.

In this distance map, we would like to see a narrow low spot at the extracted pixel position, that would assess the similarity of this pixel with its neighboring pixels, and also a low spot in the symmetrized position corrupted by noise, that would assess the robustness of the feature to noise corruption.

Figure 3.8) is arranged in 3 rows, the first row is the original image, the red squares are the positions where the compared feature vectors were extracted. Since we are dealing with 3D images, we show an axial view and a sagittal view of the brain. Next there is two sets of two rows, each set corresponds to a pixel position (a red square in the top row). In the top row of each of these sets is the experiment with the Gabor features, where 4 scales were taken and only 5 orientations, yielding a total of 40 features. Computation time for this brain image, $256 \times 256 \times 48$ in size, was 19 seconds. And no parallelization was done to make the computations faster. The bottom row of each set displays the experiment with $5 \times 5 \times 3$ patches. The last figure in each row depicts a line extracted in the distance map along the arrow, of the left image. The arrow points to the point of extraction of the feature and the red circle shows the expected position of the low spot in the distance map.

The first remark we can make about this experiment, is when we look at the left part of the images. We can see that it is much easier to distinguish an extracted Gabor feature vector from all the other features vectors than it is for a patch. Gabor features are much more discriminative in this case. On the right of the images, especially on the last row, we can see that the addition of noise renders the patches completely useless since no low point is found in the part corrupted by noise. On the other hand, Gabor features still yield significant low spots in the areas corrupted by noise. When the feature vector is extracted in an area with a strong edge, the corruption by noise renders the problem harder, since it makes the edge less detectable, thus making the distance maps less accurate.

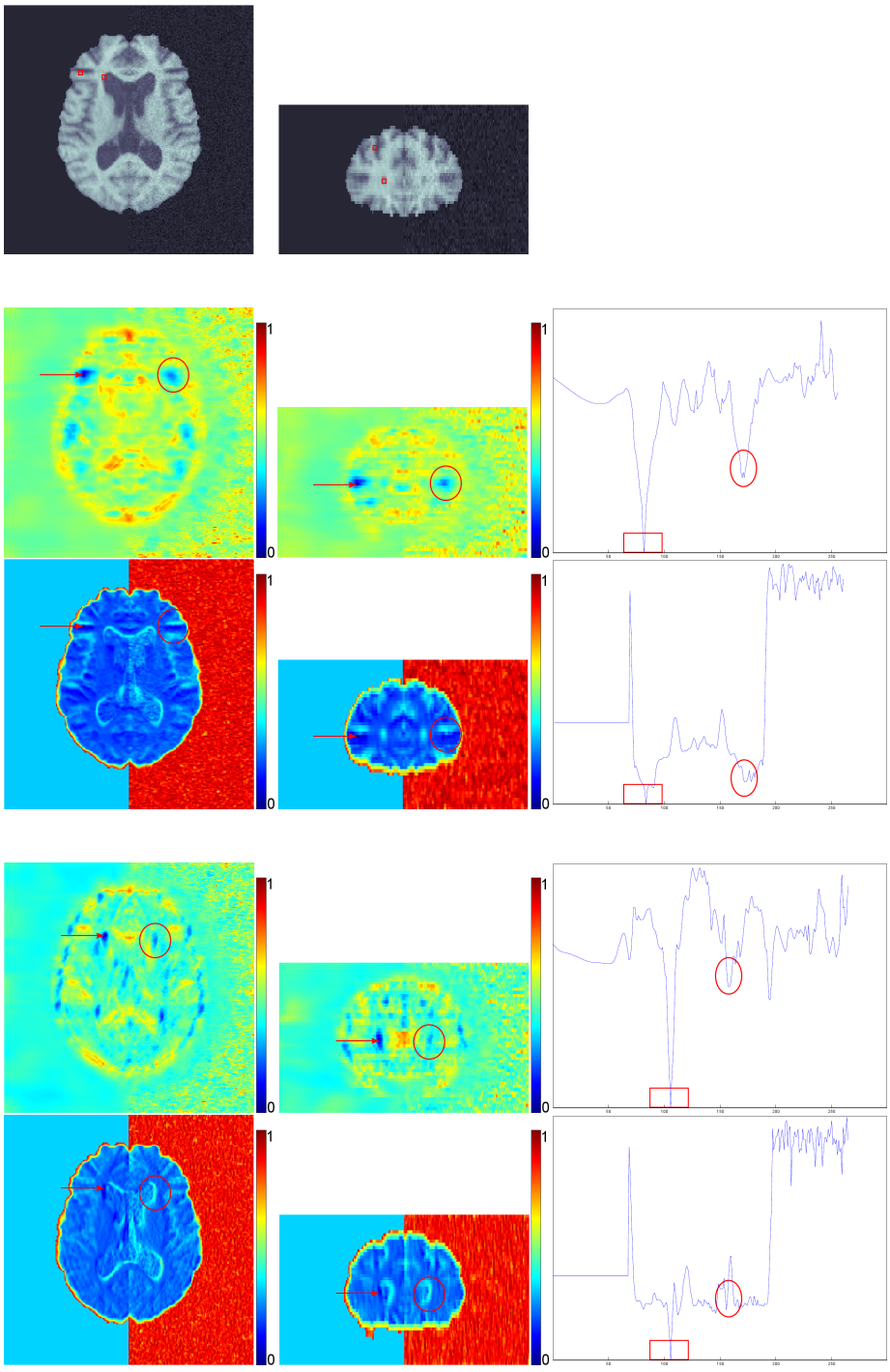


Figure 3.8: Gabor feature and patches distance maps. Top row: original image where the left of the brain is the original brain and the right is a symmetrized version of the left corrupted by noise. The red squares represent the features for which the distance maps are created. For each set of 2 rows, we depict the distance maps for Gabor features (top row) and patches (bottom row). The arrow points to the position of the feature vector of comparison and the red circle locates the expected position of the low value. On the right are diagrams showing one line of the distance map extracted on the axial image (left most) along the arrow.

Chapter 4

3D image regression for multimodal registration

In this chapter let us consider that we have two image modalities A and B, and we want to align images that are issued by such modalities. The source image will be denoted by J_B to show that it has been issued by B, in the same manner, the target image will be denoted by I_A. The intensity spaces of each image will be denoted as $\mathcal{I}_A$ and $\mathcal{I}_B$ respectively. The goal of this chapter is to map the target image from intensity space $\mathcal{I}_A$ to intensity space $\mathcal{I}_B$ and then carry out the comparison of images in $\mathcal{I}_B$. This will be done through a mapping $f : \mathcal{I}_A \longrightarrow \mathcal{I}_B$, in practice, f may not be applied directly on the intensities of I_A but on feature vectors extracted on it. In essence this is a simulation of one modality from the other. Given the image in one modality, one tries to guess how it would look had it been acquired by the other modality. There are two ways of doing this, one can use the physical properties of the imaging modalities and try to reproduce the imaging process given one image. The other way would be by learning the relationship between intensities on a set of perfectly aligned images, and use this knowledge to try and guess one the appearance of one modality in the other modality's space.

In [Roche 2001], the simulation of an ultrasound (US) image is done from the intensity and gradient information of an MR image. Even though basic US physics are not respected, building an appropriate metrics for the resulting images yield good results. Inspired by the previous results, [Wein 2008] simulated an ultrasound (US) image is from a CT image. The CT image is first projected onto the plane of incidence of the US, then a 2D US image is simulated from this CT. Here, the simulation is done using the physical properties of US images. The transmission and reflections of the US beams can be computed using the acoustic impedances of the tissues, which are assumed proportional to the tissue density. Since the X-ray attenuation in the CT images is also proportional to the tissue density, a backscatter US image can be recreated from the CT (figure 4.1). Using

the simulated image, the registration is performed in the image space of the ultrasound image, and the images are compared using a modified version of the correlation ratio.

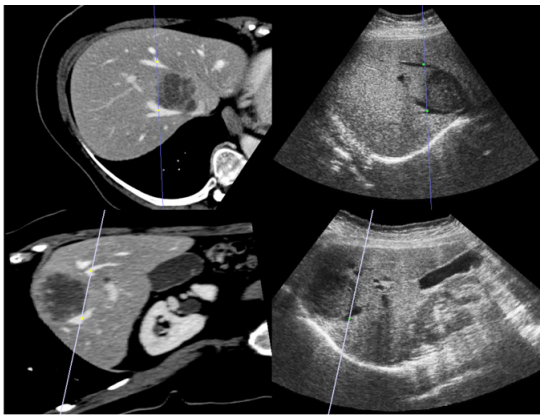


Figure 4.1: Figure extracted from [Wein 2008]: On the left, CT images projected on the plane of incidence of the US. Top right is the real US frame and Bottom right is the simulated US image

An important drawback of the physics based methods is that the physics based model has to be computed in a timely manner which cannot be done in all cases, and mostly that it is not modular in the sense that a new model has to be devised for each new pair of modalities.

A more modular method would be to learn the correspondence between images and be able to predict one image from the other by using the knowledge of previously aligned pairs. Let us assume that we have access to a data base of perfectly aligned pairs of images of modalities A and B, this data set will be denoted as $\{(I_A, J_B)_i^{\text{train}} : i \in 1, \ldots, N\}$, then given a new pair of images $(I_A^{\text{new}}, J_B^{\text{new}})$ that are not aligned, we want to first simulate the appearance of I_A in modality B: $f\left(I_A^{\text{new}}\right)$ then compare both images in the intensity space $\mathcal{I}_B$ with the distance function d_f:

$$d_f\left(I_A^{\text{new}}, J_B^{\text{new}}\right) = \left\| f\left(I_A^{\text{new}}\right) - J_B^{\text{new}} \right\|^2 \tag{4.1}$$

This problem, stated as above is a problem of statistic regression, let us first review the usual statistic regression algorithms.

4.1 Regression

In this chapter we will use the language of statistic regression and use the term target variable to express the varaible the regression is trying to infer, and use the term regressor

to express the varaibles that are used to explain the target variable.

4.1.1 Linear regression

This is the most basic type of regression. An hypothesis is made on the linear relationship between the variable to explain and the regressors. The model can be written :

$$f(\mathbf{x}) = \alpha_0 + \sum_{i=1}^{n} \alpha_i \mathbf{x}^i \tag{4.2}$$

where $\mathbf{x}^i$ is the i^{th} component of $\mathbf{x}$. Let us write

$$\mathcal{F} = \left\{ f \in \mathcal{C}^1(\mathcal{X}) \quad | \quad \forall \mathbf{x} \in \mathcal{X},\ f(\mathbf{x}) = \alpha_0 + \sum_{i=1}^{n} \alpha_i \mathbf{x}^i \right\}$$

We look then in $\mathcal{F}$ for the regression function f, minimizing the least square problem :

$$\underset{f \in \mathcal{H}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} (y_i - f(\mathbf{x}_i))^2 \right\}$$

Then if we write the matrix M:

$$M = \begin{bmatrix} 1 & x_1^1 & \cdots & x_1^p \\ 1 & \vdots & \cdots & \vdots \\ 1 & x_n^1 & \cdots & x_n^p \end{bmatrix} \tag{4.3}$$

Then the minimization problem solution writes :

$$\alpha = (M^T M)^{-1} M^T Y \tag{4.4}$$

where Y is the vector $[y_1, \ldots, y_n]$.

Linear regression is the most basic type of regression but its understanding leads to very powerful algorithms. Most data sets do not behave linearly, so applications of linear regression are very rare. Also and this is one of the major set backs, the matrix $(M^T M)$ is not guaranteed to be well conditioned or even invertible , and this happens if two variables are co-linear which might arise. Reducing the matrix M to only its non-zero (or above some threshold) singular values solves this ill condition problem but the co-lineartiy of variables is not taken into account.

4.1.2 Ridge regression

Ridge regression solves the problem of the ill-posedness of linear regression by adding a regularization on the parameters of the regression function. The problem to be minimized then becomes:

$$\underset{f\in\mathcal{H}}{\operatorname{argmin}} \left\{ \frac{1}{n}\sum_{i=1}^{n} \left(y_i - f(x_i)\right)^2 + \lambda \left\|f\right\|^2 \right\} \tag{4.5}$$

Which can also be rewritten:

$$\underset{\alpha\in\mathbb{R}^n}{\operatorname{argmin}} \left\{ \frac{1}{n} \left(M\alpha - y\right)^T \left(M\alpha - y\right) + \lambda \alpha^T M \alpha \right\}$$

the solution of which is

$$\alpha = \left(M + n\lambda I\right)^{-1} Y \tag{4.6}$$

other norms than the L^2 norm can be considered which lead to other types of regularization. In this case the problem is no longer ill posed, since $(M + n\lambda I)$ is of full rank.

4.1.3 Kernel Ridge Regression

One way to cope with the linearity of linear regression and ridge regression is to use kernels. As it is sometimes difficult to readily work on the input space, we usually use a mapping ϕ from $\mathcal{X}$ to a Hilbert space H (the feature space) were the computational scheme is well known. It is often very hard to compute the feature $\phi(\mathbf{x})$ on an element in $\mathcal{X}$. Instead of computing ϕ directly, it is often better to compute the inner-product in the feature space, which is given by the mapping of a kernel on the input space :

$$k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle$$

The kernel k has to be symmetric, and positive definite, that is :

$$\forall \left(a_1, \ldots, a_n\right) \in \mathbb{R}^n,\ \forall \left(\mathbf{x}_1, \ldots, \mathbf{x}_n\right) \in \mathcal{X},\ \sum_{i,j=1}^{n} a_i a_j k(\mathbf{x}_i, \mathbf{x}_j) \geq 0$$

Given that we look for a solution in the feature space, the representer theorem yields :

$$\exists K \text{ p.d. kernel and } \alpha \in \mathbb{R}^n \text{ s.t. } \hat{f}(\mathbf{x}) = \sum_{i=1}^{n} \alpha_i K(\mathbf{x}_i, \mathbf{x})$$

Writing K the matrix of coefficients $K(\mathbf{x}_i, \mathbf{x}_j)$, The problem then simplifies as:

$$\underset{\alpha \in \mathbb{R}^n}{\operatorname{argmin}} \left\{ \frac{1}{n} (K\alpha - y)^T (K\alpha - y) + \lambda \alpha^T K \alpha \right\} \tag{4.7}$$

It is now an ordinary quadratic problem, the solution of which is :

$$\alpha = (K + n\lambda I)^{-1} y \tag{4.8}$$

Kernel ridge regression can be very powerful and was used for image regression in [Hofmann 2008], where kernel ridge regression is used to predict the intensities of CT images given MR images. In the case of image prediction, the feature vector $\mathbf{x}$ is assumed to be a pair made of a local patch and normalized coordinates: $\mathbf{x} = (\mathbf{p}_i, \mathbf{c}_i)$, and the kernel is defined as:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(\frac{-\|\mathbf{p}_i - \mathbf{p}_j\|^2}{2{\sigma_{\text{patch}}}^2} \right) \exp\left(\frac{-\|\mathbf{c}_i - \mathbf{c}_j\|^2}{2{\sigma_{\text{pos}}}^2} \right) \tag{4.9}$$

σ_{patch} and σ_{pos} are determined by *cross-validation*. This method, however, makes use of the pixels coordinates in the image to deal with position dependent image variations. In the case of static image prediction (when there is no need for registration), this method is very adapted, but as soon as the new image is deformed, and so different from the training set in position, the prediction fails to be accurate. The authors tried to learn each kernel function f in sub regions to localize the problem but report possible registrations with only minor misalignments. In figure 4.2, one result extracted from [Hofmann 2008] is shown in the case of the static CT prediction from an MR image.

The biggest drawback of kernel methods is that the kernel matrix is the size of the training sample which doesn't allow for large training sample base since we need to invert the kernel matrix in the kernel ridge regression.

4.1.4 Bayesian interpretation of linear regression

We have seen that the regression function is a linear combination of regressors in the case of linear regression. In order to model the variability in the training data set, we need to add noise to the prediction in order to recover each and every sample, if we assume there is N samples in the training data set then:

$$\forall i \in \{1, \ldots, N\}, \quad y_i = \alpha_0 + \sum_{j=1}^{n} \alpha_j \mathbf{x}_i^j + \epsilon_i$$

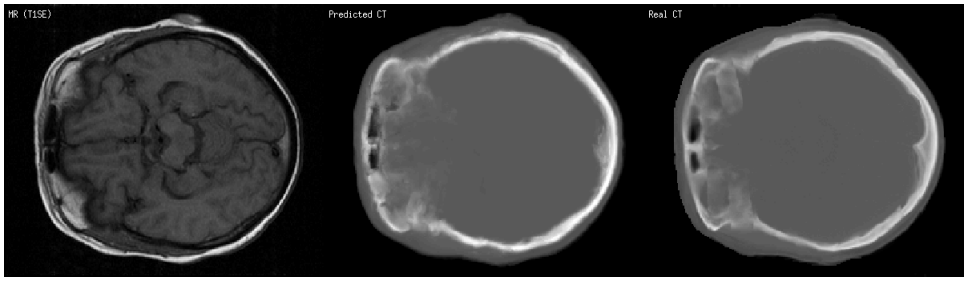


Figure 4.2: Figure extracted from [Hofmann 2008]: Left, MR image, Middle, Predicted CT, Right, Original CT

where ϵ is a random variable. The solution we gave to linear regression assumes a normal distribution to ϵ, that is:

$$\forall i \in \{1, \ldots, N\}, \quad \epsilon_i \sim \mathcal{N}(0, \sigma)$$

And we can write the conditional probability of y given $\mathbf{x}$ as a gaussian distribution:

$$p(y|\mathbf{x}) = \mathcal{N}\left(y \left| \mathbf{x}^T \boldsymbol{\alpha}, \sigma \right.\right) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[\frac{-1}{2\sigma^2}\left(y - \mathbf{x}^T\boldsymbol{\alpha}\right)^2\right] \tag{4.10}$$

this formulation will help us to model mixtures of linear regressions.

4.2 Mixture Models

Mixture models are a good way to model multi-modal probabilities, in the sense where one probability distribution is not enough to fit the data appropriately. Discussions on Mixture models can be found in [Bishop 2006]. Mixture models are the linear combination of several uni-modal probability distributions. They make the assumption of a hidden mixture variable z, that defines for each sample, which probability distribution it follows. A mixture model can be represented by a very simple two nodes graph, as shown in figure 4.3.
The joint probability of this model writes:

$$p(x, z) = p(z)p(x|z) \tag{4.11}$$

z is a latent variable so we do not have access to it, if we assume that z is a discrete variable and we marginalize over it, we get the probability of x:

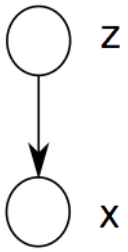


Figure 4.3: Two nodes generative model

$$p\left(x\left|\theta\right.\right)=\sum_{i}p\left(z=i\left|\theta_{i}^{1}\right.\right)p\left(x\left|\theta_{i}^{2}\right.\right)$$

where θ_i^1 and θ_i^2 are the parameters of the distribution of z and x in the mixture i. θ is the concatenation of these parameters.

If we have N samples drawn independently and identically distributed then the probability of the random variable $X=(x_1,\ldots,x_N)$ writes:

$$p\left(X\left|\theta\right.\right)=\prod_{n}\sum_{i}p\left(z_{n}=i\left|\theta_{i}^{1}\right.\right)p\left(x_{n}\left|\theta_{i}^{2}\right.\right)$$

Estimation of the parameters by maximum likelihood is not trivial due to the summation after the multiplication that stays after application of the logarithm. Estimation of the parameters is done by Expectation Maximization.

4.2.1 Expectation maximization

If $Z=(z_1,\ldots,z_N)$ could be observed, then finding the parameters would amount to maximizing the *complete log likelihood* :

$$\ell_{c}\left(\theta\left|x,z\right.\right)=\log p\left(x,z\left|\theta\right.\right)$$

Z is not observed though and we have to marginalize to get the *log likelihood* :

$$\ell\left(\theta\left|x\right.\right)=\log p\left(x\left|\theta\right.\right)=\log\sum_{z}p\left(x,z\left|\theta\right.\right)$$

The summation is not easily tractable for a maximization problem. In the complete log likelihood what happens is that since Z is not observed, ℓ_c is a random variable. We can however average ℓ_c using an averaging distribution $q(z|x)$.

We define the *expected complete log likelihood* as:

$$\langle \ell_c(\theta \,|x, z)\rangle_q = \sum_z q(z|x,\theta) \log p(x, z|\theta)$$

Maximizing a lower bound on the log likelihood

$$\begin{aligned}
\ell(\theta \,|x) &= \log p(x|\theta) \\
&= \log \sum_z p(x, z|\theta) \\
&= \log \sum_z q(z|x) \frac{p(x, z|\theta)}{q(z|x)} \\
&\geq \sum_z q(z|x) \log \frac{p(x, z|\theta)}{q(z|x)} = \mathcal{L}(q, \theta)
\end{aligned}$$

In the last equation, we used the *Jensen's Inequality*

The EM algorithm consist in maximizing this lower bound, to that end, a coordinate ascent is sufficient :

$$\begin{aligned}
\text{(E step)} \quad q^{(t+1)} &= \underset{q}{\operatorname{argmax}} \; \mathcal{L}(q^{(t)}, \theta^{(t)}) \\
\text{(M step)} \quad \theta^{(t+1)} &= \underset{\theta}{\operatorname{argmax}} \; \mathcal{L}(q^{(t+1)}, \theta^{(t)})
\end{aligned}$$

M step

$$\begin{aligned}
\mathcal{L}(q, \theta) &= \sum_z q(z|x) \log \frac{p(x, z|\theta)}{q(z|x)} \\
&= \sum_z q(z|x) \log p(x, z|\theta) - \sum_z q(z|x) \log q(z|x) \\
&= \left\langle \ell_c\left(\theta^{(t)} \,|x, z\right)\right\rangle_q - \sum_z q(z|x) \log q(z|x)
\end{aligned}$$

thus maximizing $\mathcal{L}(q, \theta)$ with respect to θ is equivalent to maximizing the expected complete log likelihood.

E step The maximum of $\mathcal{L}(q, \theta)$ with respect to q is attained when $q^{(t+1)}(z|x) = p(z|x, \theta^{(t)})$ indeed it is easy to see that

$$\mathcal{L}\left(p\left(z\left|x, \theta^{(t)}\right.\right), \theta\right) = \ell\left(\theta^{(t)}\left|x\right.\right)$$

Which is the upper bound on $\mathcal{L}(q, \theta)$. We can consider $p\left(z\left|x, \theta^{(t)}\right.\right)$ as the best current guess on the values of the latent variables given x, this estimation allows to compute an expectation of the complete log likelihood, that will be maximized at the next iteration yielding the new parameters $\theta^{(t+1)}$.

The EM algorithm In the EM iterations, the M step finds the parameters that increase a lower bound on the likelihood, and the E step makes the the bound as close as possible to the actual function to actually act on the log likelihood.

$$\begin{aligned} &\text{(E step)} \quad q^{(t+1)} = p(z|x, \theta^{(t)}) \\ &\text{(M step)} \quad \theta^{(t+1)} = \underset{\theta}{\operatorname{argmax}} \left\langle \ell_c\left(\theta^{(t)}\left|x, z\right.\right)\right\rangle_{q^{(t+1)}} \end{aligned} \tag{4.12}$$

4.2.2 Gaussian Mixture Model

As an example of EM parameter estimation and a good intuition on mixtures of linear regressions discussed later, let us briefly see the parameter estimation of Gaussian mixture models. The normal distribution writes:

$$\mathcal{N}\left(\mathbf{x}\left|\boldsymbol{\mu}, \Sigma\right.\right) = \frac{1}{(2\pi)^{m/2}\left|\Sigma\right|^{1/2}} \exp\left\{-\frac{1}{2}\left(\mathbf{x}-\boldsymbol{\mu}\right)^T \Sigma^{-1}\left(\mathbf{x}-\boldsymbol{\mu}\right)\right\}$$

The probability of a data-set sample following a Gaussian mixture model then writes:

$$p\left(\mathbf{x}_n\left|\theta\right.\right) = \sum_i \tau_n^i \, \mathcal{N}\left(\mathbf{x}_n|\boldsymbol{\mu}_i, \Sigma_i\right)$$

where $\tau^i = p(z = i)$. Parameter estimation via EM leaves us with the problem of the estimation of $p(z|x, \theta^{(t)})$ which is simply solved by application of the Bayes rule:

$$\begin{aligned} p\left(z_n = i \left| \mathbf{x}_n, \theta^{(t)} \right.\right) &= \left(T_n^i\right)^{(t)} \\ &= \frac{p\left(z_n = i, \mathbf{x}_n \left| \theta^{(t)} \right.\right)}{p\left(\mathbf{x}_n \left| \theta^{(t)} \right.\right)} \\ &= \frac{\tau_i^{(t)} \mathcal{N}\left(\mathbf{x} | \boldsymbol{\mu}_i^{(t)}, \Sigma_i^{(t)}\right)}{\sum_j \tau_i^{(t)} \mathcal{N}\left(\mathbf{x} | \boldsymbol{\mu}_j^{(t)}, \Sigma_j^{(t)}\right)} \end{aligned}$$

The parameters in the M step are obtained through simple derivation:

$$\boldsymbol{\mu}_i^{(t+1)} = \frac{\sum_n \left(T_n^i\right)^{(t)} \mathbf{x}_n}{\sum_n \left(T_n^j\right)^{(t)}} \tag{4.13}$$

$$\Sigma_i^{(t+1)} = \frac{\sum_n \left(T_n^i\right)^{(t)} \left(\mathbf{x}_n - \boldsymbol{\mu}_i^{(t+1)}\right) \left(\mathbf{x}_n - \boldsymbol{\mu}_i^{(t+1)}\right)^T}{\sum_n \left(T_n^i\right)^{(t)}} \tag{4.14}$$

$$\tau_j^{(t+1)} = \frac{1}{N} \sum_n \left(T_n^i\right)^{(t)} \tag{4.15}$$

4.2.3 Mixture of regression models

Mixture of regression models or mixture of experts have been introduced in [Jacobs 1991]. The basic idea is to infer the conditional distribution using a mixture of regression models which are local conditional probabilities. The overall conditional probability is obtained by smoothly mixing local conditional distributions. The model is represented by a 3 nodes generative graph shown in figure 4.4.
The probability of the model writes:

$$p(x, y, z) = p(x)p(z|x)p(y|x, z) \tag{4.16}$$

And we get the conditional probability by marginalization over z:

$$p(y|x) = \sum_z p(z|x)p(y|x, z)$$

In [Jacobs 1991] $p(z|x)$ is modeled using a gating network with a soft-max function:

$$p(z^i = 1|\mathbf{x}, \boldsymbol{\xi}) = \tau_i(\mathbf{x}, \boldsymbol{\xi}) = \frac{e^{\boldsymbol{\xi}_i^T \mathbf{x}}}{\sum_j e^{\boldsymbol{\xi}_j^T \mathbf{x}}} \tag{4.17}$$

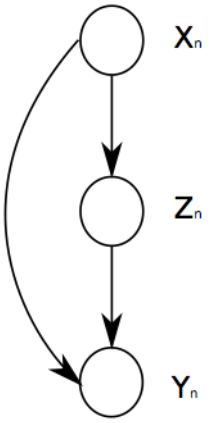


Figure 4.4: Conditional regression model

Mixture of Linear regressions We are now left with the modeling of the term $p(y|\mathbf{x}, z)$ which we can easily do by recalling equation 4.10

$$p\left(y\left|\mathbf{x}, z\right.\right) = \mathcal{N}\left(y\left|\mathbf{x}^T\boldsymbol{\alpha}, \sigma\right.\right)$$

The whole model then writes:

$$p(y|x,\theta) = \sum_i \tau_i(\mathbf{x}, \boldsymbol{\xi})\, \mathcal{N}\left(y\left|\mathbf{x}^T\boldsymbol{\alpha}_i, \sigma_i\right.\right) \tag{4.18}$$

Parameter inference: Parameter inference is once again done through expectation maximization. Let the *complete* data set be $\mathcal{D}_c = \{(\mathbf{x}_n, z_n, y_n) : n = 1, \ldots, N\}$, where $z_n \in \{0,1\}^K$ is a discrete variable, and K is the number of mixture components. The complete log likelihood writes

$$\ell_c\left(\theta\left|\mathcal{D}_c\right.\right) = \sum_n \sum_i z_n^i \log\left[\tau_i\left(\mathbf{x}_n, \boldsymbol{\xi}\right) p\left(y_n\left|Z^i = 1, \mathbf{x}_n, \theta_i\right.\right)\right]$$

Since we don't have access to the hidden variables z_n we write the expected complete log likelihood:

$$\left\langle \ell_c\left(\theta\left|\mathcal{D}_c\right.\right)\right\rangle_q = \sum_n \sum_i q\left(z_n^i = 1\left|x_n\right.\right) \log\left[\tau_i\left(\mathbf{x}_n, \boldsymbol{\xi}\right) p\left(y_n\left|z_n^i = 1, \mathbf{x}_n, \theta_i\right.\right)\right]$$

The expectation step follows directly that of the Gaussian mixture model and using Bayes rule we have:

$$p\left(z_n = i \left| \mathbf{x}_n, y_n, \theta^{(t)} \right.\right) = \left(T_n^i\right)^{(t)} = \frac{\tau_i^{(t)} \ \left(\mathbf{x}_n, \boldsymbol{\xi}\right) \ p\left(y_n \left| z_n^i = 1, \mathbf{x}_n, \theta_i^{(t)} \right.\right)}{\sum_j \tau_j^{(t)} \ \left(\mathbf{x}_n, \boldsymbol{\xi}\right) \ p\left(y_n \left| z_n^j = 1, \mathbf{x}_n, \theta_j^{(t)} \right.\right)}$$

The maximization step is trickier, indeed due to the presence of the soft-max function, there is no closed form solution for the inference of the parameters.

Modeling with a Gaussian Mixture: In [Xu 1995] modelling of $p(z|x)$ is done using a two node generative model like the on in figure 4.3:

$$p(x, z) = p(z)p(x|z)$$

which yields:

$$p(z|x) = \frac{p(z)p(x|z)}{\sum_z p(z)p(x|z)}$$

Now let us assume that z is discrete and that $p(x|z)$ follows a normal distribution we have :

$$p\left(z^i = 1 \left| \mathbf{x} \right.\right) = \frac{p(z^i = 1)\mathcal{N}\left(\mathbf{x}|\theta^i\right)}{\sum_j p(z^j = 1)\mathcal{N}\left(\mathbf{x}|\theta^j\right)} \tag{4.19}$$

Using $p\left(z^i = 1 \left| \mathbf{x} \right.\right)$ as is, does not solve the problem, in [Xu 1995] the approximation is made that the parameters inference of $p\left(z^i = 1 \left| \mathbf{x} \right.\right)$ is done on:

$$p\left(z^i = 1, \mathbf{x} \left| \boldsymbol{\mu}_i, \Sigma_i, \gamma_i \right.\right) = p(z^i = 1)\mathcal{N}\left(\mathbf{x}|\theta^i\right) = \gamma_i \ \mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}_i, \Sigma_i\right) \tag{4.20}$$

But the overall conditional probability is still evaluated with $p\left(z^i = 1 \left| \mathbf{x} \right.\right)$. The expected complete log likelihood rewrites:

$$\left\langle \ell_c\left(\theta \left| \mathcal{D}_c \right.\right)\right\rangle_q = \sum_n \sum_i q\left(z_n^i = 1 \left| x_n \right.\right) \log\left[\gamma_i \ \mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu}_i, \Sigma_i\right) \ p\left(y_n \left| z_n^i = 1, \mathbf{x}_n, \theta_i \right.\right)\right]$$

its maximization through derivation yields:

$$\boldsymbol{\mu}_i^{(t+1)} = \frac{\sum_n \left(T_n^i\right)^{(t)} \mathbf{x}_n}{\sum_n \left(T_n^j\right)^{(t)}} \tag{4.21}$$

$$\Sigma_i^{(t+1)} = \frac{\sum_n \left(T_n^i\right)^{(t)} \left(\mathbf{x}_n - \boldsymbol{\mu}_i^{(t+1)}\right) \left(\mathbf{x}_n - \boldsymbol{\mu}_i^{(t+1)}\right)^T}{\sum_n \left(T_n^i\right)^{(t)}} \tag{4.22}$$

$$\gamma_i^{(t+1)} = \frac{1}{N} \sum_n \left(T_n^i\right)^{(t)} \tag{4.23}$$

$$\alpha_i^{(t+1)} = (M^T T_i^{(t)} M)^{-1} M^T T_i^{(t)} Y \tag{4.24}$$

$$\sigma_i^{(t+1)} = \frac{\sum_n \left(T_n^i\right)^{(t)} \left(y - \mathbf{x}_n^T \boldsymbol{\alpha}_i\right)}{\sum_n \left(T_n^i\right)^{(t)}} \tag{4.25}$$

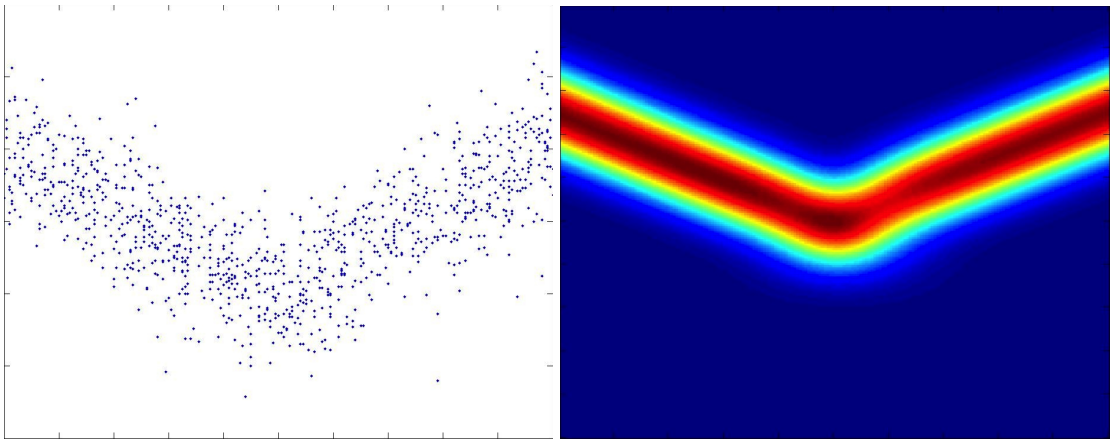
where M has been defined in equation 4.3 and T_i is the matrix with diagonal T_n^i.

4.3 Experiments with regression

We argue here that mixture of experts is very well suited for a medical application. First there is only one parameter to set for the algorithm and it is the number of experts used. As with all learning algorithm, the higher the number of experts the closest we will be to the actual distribution, but also we will be more prone to over-fitting and find ourselves with experts that try to mostly fit noise. The choice of the number of experts is thus a trade-off between accuracy and over-fitting. The structure of mixture of experts accommodates well with non-linearities thanks to the combination of linear regressions, and is flexible on the input data. We will only require that the data is well dimensionned, meaning that there is not one dimension in the input vector that overpowers the others, to that end the data might be redistributed on a unit sphere by centering it (removing its mean) and scaling it (divide by its standard deviation), a prior PCA might be also considered. Second, contrarily to other regression algorithms, the output of mixture of experts is a conditional probability, maximizing it yields the regression function but we have the possibility to exploit the whole modeled information, this is particularly helpful for medical images as will be shown in the next section. Let us first see the performance of mixture of experts on some synthetic and real data sets.

4.3.1 Synthetic Data

Let us have a look at the performance of Mixture of experts on scalar inputs and outputs, this way we can visualize the data. The first experiment in figure 4.5 is very common in non-linear regression, and particularly suited for a mixture of linear regressions. The distribution of points follows two straight lines intersecting in the middle. We expect to have two gating functions one that will act on the left part for a first expert and one that acts on the right part for a second expert. In this experiment, we set the number of experts to 2 which is obviously the optimal number. The obtained optimal conditional probability neatly follows the test distribution.

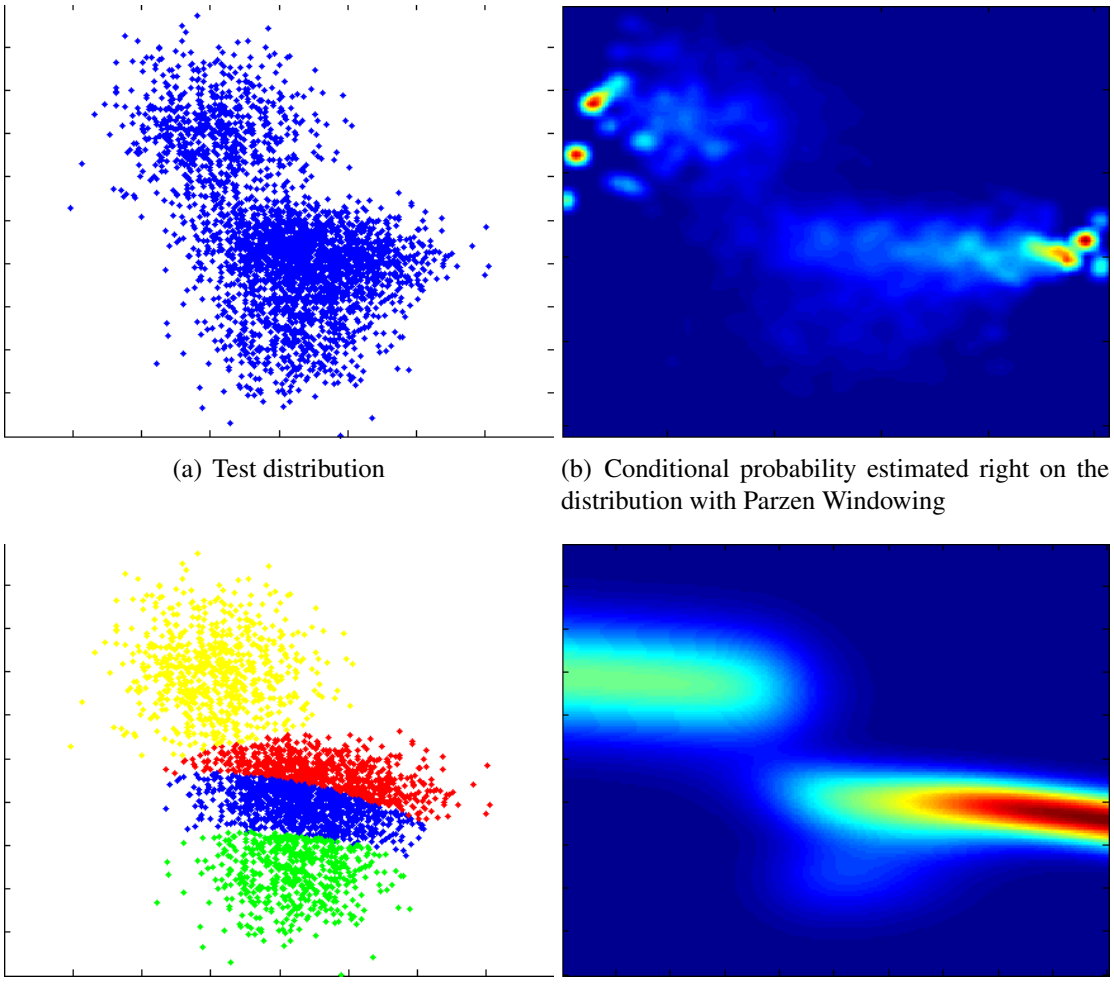


(a) Test distribution (b) Conditional probability

Figure 4.5: One connected cloud of points with two intersecting lines, the color plot on the right represents the estimated densities, gradients of red represent a high density while gradients of blue represent a low density

The second experiment is intended to highlight the fact that not only do we have access to a regression function by maximizing the conditional probability but also we have access to a full probability map that allows us to model the non-functionalities in the data, like when the same input yields different outputs, and it possibly not due to the noise. In order to do so, the next data set is a stacked distribution of Gaussian distributed points, shown in figure 4.6(a). In figure 4.6(b), the conditional probability is estimated right on the points distribution with Parzen Windowing (see [Bishop 2006] on Kernel density estimators for instance) to give an intuition of where the highest concentrations of points are located. The mixture of experts is initialized using a prior clustering, then on each cluster, a linear regression is conducted and all the parameters are estimated that gives us the initial param-

eters. The initial clustering is done using a mixture of Gaussian distributions (initialized with k-means which is common practice) and is displayed in figure 4.6(c). Finally we present the conditional probability estimated by Mixture of experts in figure 4.6(d). We can see that if we draw a vertical line in the middle of the distribution, we will end up with 3 distinct local maxima and while the distribution itself doesn't give us a way to chose between them, we have more that just the maximum of the conditional probability. In the next section we will discuss how we can select the right local maximum for medical imaging applications.



(a) Test distribution

(b) Conditional probability estimated right on the distribution with Parzen Windowing

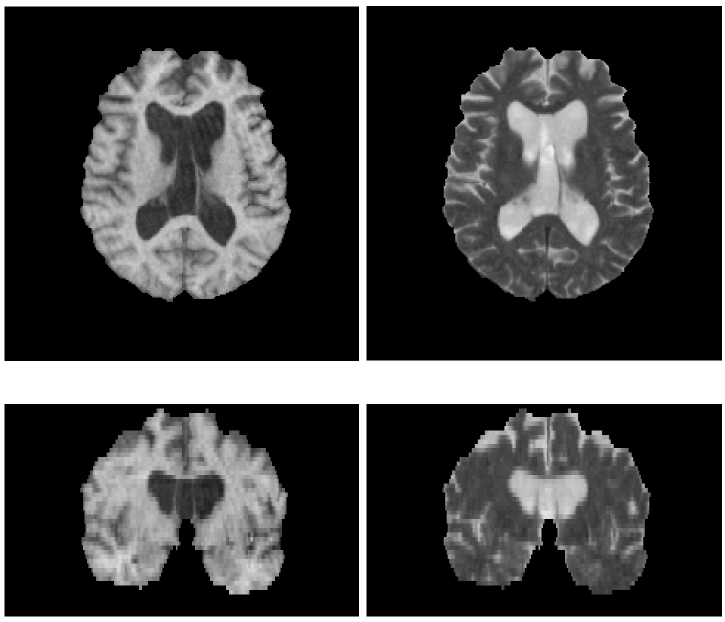
(c) Initial clustering

(d) Estimated conditional probability

Figure 4.6: Stacked mixture of Gaussian distributions

4.3.2 Real Data

To showcase the capabilities of a mixture of linear regressions, we will experiment on two pairs of images both perfectly co-registered. The images are T1 (source) and T2 (target) MRI images of the brain taken during the same session and while the patient's head was immobilized. Images can be seen in figure 4.7. For this experiment we only use one brain, it has to be noted that for more statistical relevance more patients brains should be used for training, and this experiment should be more considered as a proof of concept.



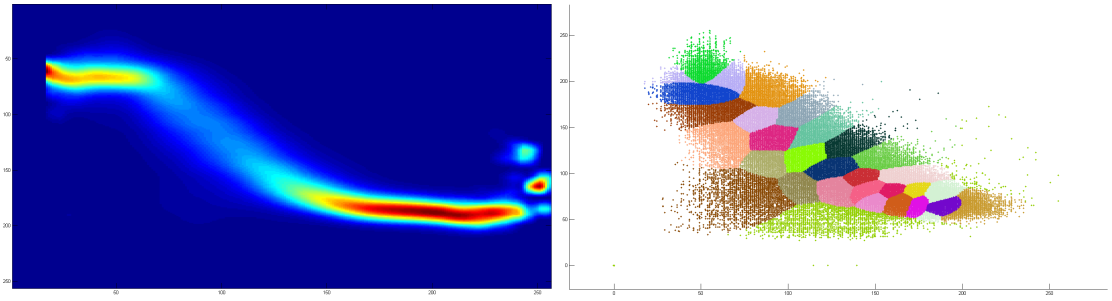
(a) source T1-MRI (b) target T2-MRI

Figure 4.7: Two trainig data set examplar images

The training data-set consists in patches (see section 3.1) of size $5 \times 5 \times 3$, amounting to 75-dimensional feature vectors, that are extracted densely on the source training image, and the scalar intensities extracted on the target training image. The computational efficiency of the algorithm heavily relies on the size of the training set, for that reason the black background (all-zero patches) was removed from the training set. In the case where several brain images are considered, uniform sampling of the data set is required (statistically relevant sampling can also be considered in order to drive a better regression).

A visualization of the data points distribution is not easy in the 76 dimensional space. Here we propose to visualize density of the source intensity (coincidentally also the center

of each patch) plotted against the target intensity as shown in figure 4.8(a). This visualization was also used to perform the initial clustering of the data set using a mixture of 30 Gaussian distributions as can be seen in figure 4.8(b). The clustering of a 76-dimensional data set would be extremely biased towards the source feature vectors, this is why we resorted to cluster on the intensity space, and translate the cluster memberships to the data set elements.



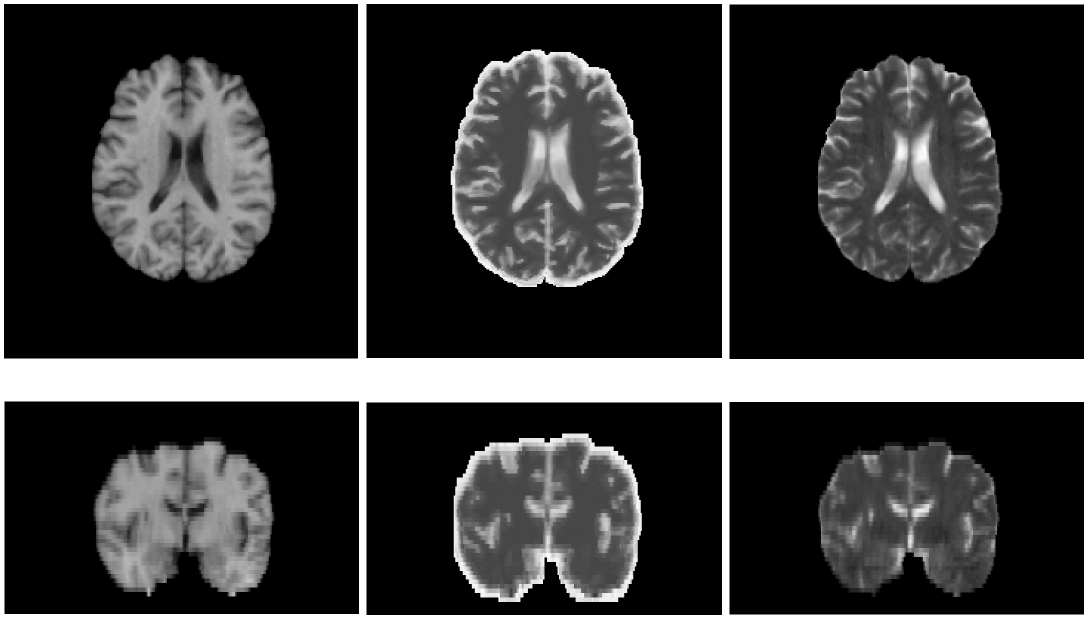
(a) source and target intesities density distribution (b) Initial clustering of the data set for 30 experts

Figure 4.8: Visualization of the densities, on the left is the joint histogram of input and output intensities (we show here only one input intensity for visualization purposes but computations were carried out in a multidimensional input space). On the right we show the initial clustering with 30 experts in the same intensity space as the left for visualization purposes as well.

The testing of the mixture of experts on a new T1-MRI source image can be seen in figure 4.9. Image 4.9(b) has been obtained by the maximization of the conditional probability. We can see that the regressed image is a lot like the actual T2-MRI image in figure 4.9(c), even though the intensity distribution is far from linear as can bee seen in figure 4.8(a). Yet the result image presented here would be hard to accurately register to a T2-MRI, some intensities are clearly off in this image. Most notably there is a white halo around the brain. This is easily explainable by the fact that the background black matches the ventricles black in the T1-MRI, maximizing on the conditional probability just misses the non functionality that the black intensity can map to either a white or a black intensity. This precise issue will be discussed in the next section.

4.4 Solving the one to one problem

Such situations can arise where there is not a one-to-one application between a feature vector and an intensity. If we remind ourselves that Ω is the spatial domain for all images,



(a) T1-MRI test source Image (b) Maximum of the conditional probability (c) Actual T2-MRI target Image

Figure 4.9: Testing Mixture of experts on a new T1-MRI image

$\mathbf{x}$ a position vector in Ω, and that J denotes the source image and I denotes the target image, the aim of image regression is to define an operator f such that:

$$\forall \mathbf{x} \in \Omega, \quad f\left(J\left(\mathbf{x}\right)\right) \simeq I\left(\mathbf{x}\right) \tag{4.26}$$

It is important to note here that f is not dependent on the spatial position $\mathbf{x}$ as is the case in [Hofmann 2008]. Such a model is compact but also ill-posed. The same origin intensity (different spatial positions) could be mapped to numerous different intensities in the target space, or

$$J\left(\mathbf{x}_0\right) = J\left(\mathbf{x}_1\right) \quad \text{and} \quad I\left(\mathbf{x}_0\right) \neq I\left(\mathbf{x}_1\right) \tag{4.27}$$

These situations cannot be modeled by a unique function of the input space, we refer to these situations as non-functionality. To cope with such non-functionalities we adopt a two-component approach. First we augment the information space on which the transport function is defined using a feature vector extracted at the point position $\mathbf{x}$ as input of the regression function. Following the notation we introduced in 3.1, this feature vector extraction will be denoted as $\boldsymbol{\pi}\left(I, \mathbf{x}\right)$. The feature extraction function is assumed to bring

some context to the extracted feature by taking into account the direct neighborhood of the spatial point extraction. This augmentation of the information on each pixel drastically reduces the occurrences of the situation earlier described. There is a trade-off to consider in he design of the feature extraction function, the bigger the neighborhood it will act upon, the less ambiguity we will encounter, but in the same time the less general the learned function will be. Hence we will still face situations where ambiguities arise as explained in figure 4.10. For medical images, this lack of one-to-one mapping is due to the use of different modalities and to the locality of some artifact in images.

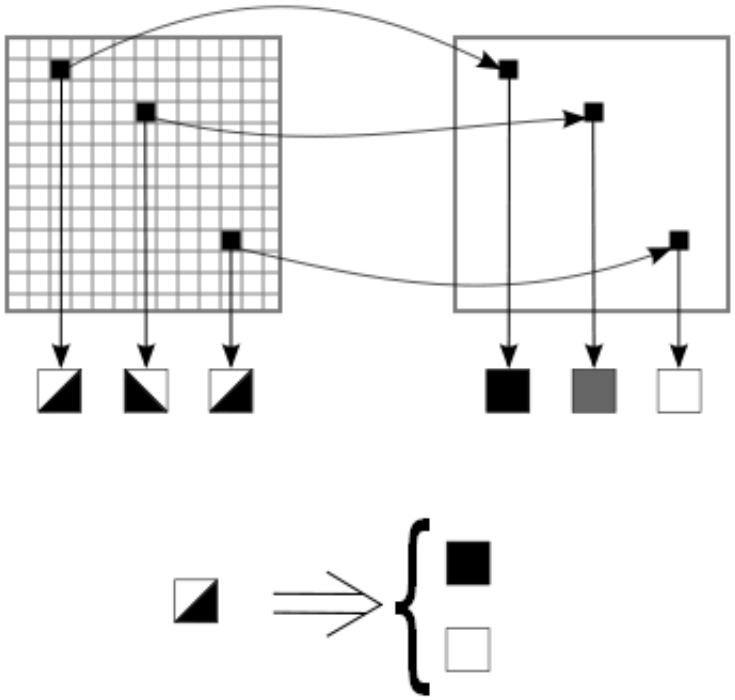
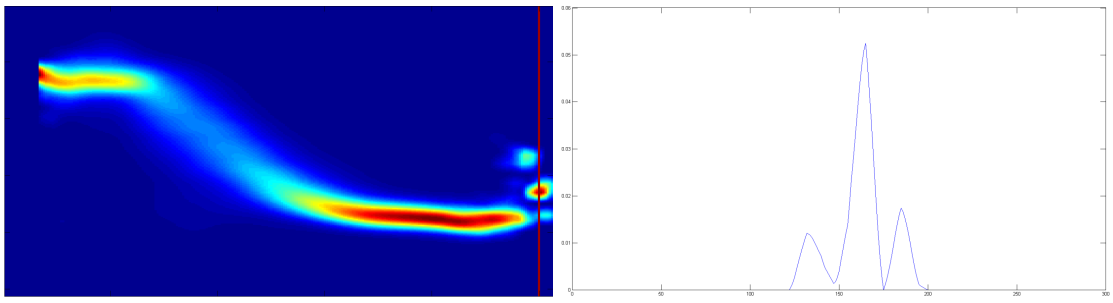


Figure 4.10: The locality of some image features prevents us from assuming a one-to-one correspondence between feature vectors.

We can see this effect in the images used in the previous section, if we take a closer look at figure 4.8(a) we can see that one intensity can map to multiple intensities as shown in figure 4.11. We also can see that this problem still arises even when neighborhood information is taken into account in the form of a feature vector, as we saw in the previous section with the problem of the white halo around the brain.



(a) Graphed line is extracted along the red line

(b) For some input intensities, 3 output intensities can be found

Figure 4.11: Visualization of the local maxima for one input intensity

4.4.1 Markov Random Field smoothing

As we have seen, mixture of experts provides us with a complete conditional probability profile instead of only a regression function. Using this fact, for each and every point location in the image, instead of the maximum of the conditional probability, we can focus on local maxima of the conditional probability. For each local maximum we have access to the conditional probability of its occurrence which we can easily transform into a score that will help us chose the right local maximum. The decision of taking only local maxima, instead of the full probability profile is only a choice of discretization of the problem with computational efficiency in mind. Optimizing the score would obviously lead to the having the maximum a posteriori (MAP) of the conditional probability and give rise to the same image as the one in the previous section. Instead, we chose to balance this score with a smoothing constraint that forces a decision on an intensity in one position of the image to be consistent with decisions in a defined neighborhood of the image.

Let us assume that we retain M local maxima, for each pixel location $\mathbf{x} \in \Omega$. Now let us assume that those maxima are ordered according to their probability, and then labeled with $L = \{\ell_1(\mathbf{x}), \ldots, \ell_N(\mathbf{x}) : \ \forall \mathbf{x} \in \Omega\}$, where ℓ_1 denotes the maximum with highest probability. If we consider the function $lmax$ that extracts the local maxima of the conditional probability and order them: $lmax\left(p\left(I\left(\mathbf{x}\right)|J\left(\mathbf{x}\right), \boldsymbol{\theta}\right), \ell_n\left(\mathbf{x}\right)\right)$, that will be written $lmax_n(\mathbf{x})$ for short, then we have:

$$p\left(I\left(\mathbf{x}\right) = lmax_1(\mathbf{x})|J\left(\mathbf{x}\right), \boldsymbol{\theta}\right) \geq \ldots \geq p\left(I\left(\mathbf{x}\right) = lmax_N(\mathbf{x})|J\left(\mathbf{x}\right), \boldsymbol{\theta}\right) \tag{4.28}$$

Now let us consider the discrete Markov Random Field energy (we refer the reader to section: 2.3.2):

$$E(L) = \sum_{\mathbf{x}\in\Omega} -\log\left(p\left(I\left(\mathbf{x}\right) = lmax(\mathbf{x})|J\left(\mathbf{x}\right), \boldsymbol{\theta}\right)\right) + \gamma \sum_{\mathbf{x}\in\Omega}\sum_{\mathbf{y}\in\mathcal{N}(\mathbf{x})} |lmax(\mathbf{x}) - lmax(\mathbf{y})| \quad (4.29)$$

where the first part of the energy, the unary term, is the data fidelity part of the energy, and minimizing it alone would yield the MAP. The second part of the energy, the pairwise term, is the smoothing part of the energy, when minimized it enforces a continuity among chosen local maxima in a neighborhood of $\mathbf{x}$ denoted as $\mathcal{N}(\mathbf{x})$. Minimization of this kind of energy has been discussed in section 2.3.2. In this work we chose to solve this problem with the now widely used Fast-PD solver [Komodakis 2007, Komodakis 2008], that has proven computational efficiency on this kind of problem. This method leaves us with 3 new parameters to set: the number of retained local maxima M, the smoothness balancing term γ and the neighborhood paradigm $\mathcal{N}(\mathbf{x})$.

4.5 Results

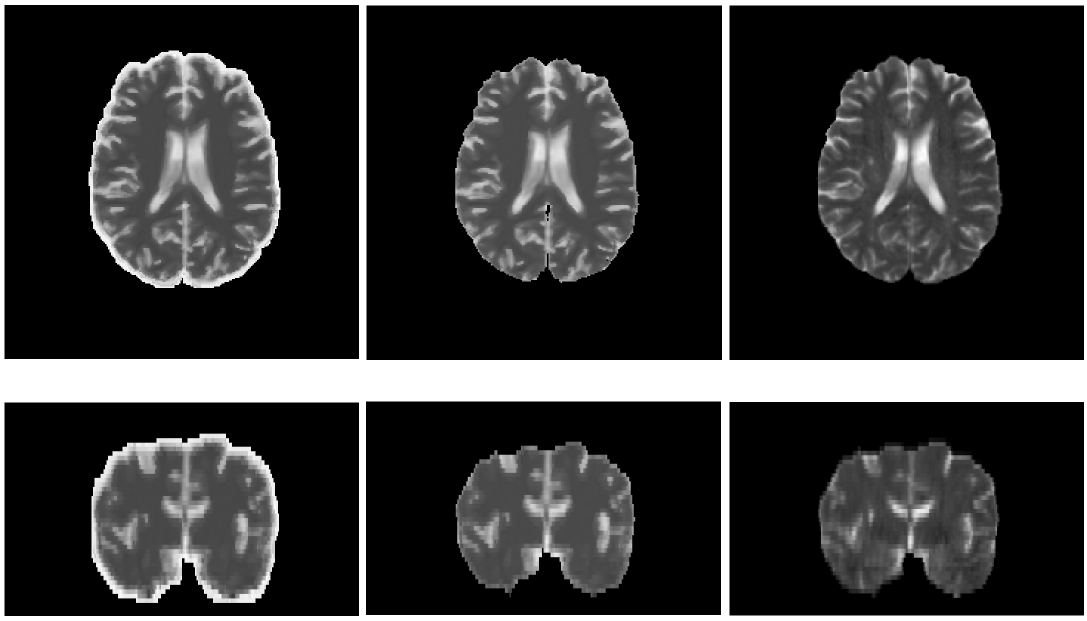
Some of the results reported here were published in the International symposion on biomediacla imaging (ISBI) [Michel 2010]. Two different types of data sets were considered to evaluate the potential of our method. The first data set consists in MRI images of the brain with 3 MRI modalities (T1, T2 and Proton density -PD-) acquired in the same session and thus perfectly co-registered, of 10 patients, resulting in a total of 30 images. These images were kindly provided by *Professor Christos Davatzikos* and his team[1]. The second data set consists in 4 images of perfectly aligned whole body correction attenuated PET images and CT images that were acquired concurrently (attenuation correction is done using the CT image). We chose to only work with the chest sections of the images due to the easily spotable features such as the lungs. This data set was kindly provided by the french based company Intrasense[2]. In all cases all images were rigidly registered to one image of the data set in order to remove the rigid registration component of the equation.

For all experiments, we found that setting the number of experts to 30 yields good results without going into the pitfall of over fitting. The value of 30 experts was set using 10-fold cross-validation on a random subset of the patches. The number of allowed local maxima was set to 3 as it was the maximal number of concurrent local maxima encountered. We experimented with several neighborhood paradigms, linking each node to

[1] http://www.rad.upenn.edu/sbia/
[2] www.intrasense.fr

6 of its neighbors already yields good results in the smoothing, in the experiments that we run, having a larger neighborhood resulted in over smoothed results and deteriorated the overall quality. Finally the parameter γ regulating the trade off between smoothing and the data fitness, was set using leave one out cross-validation on the comparison with the actual expected data (mean absolute error).

Following the results obtained in figure 4.9, the effect of the MRF smoothing on the image can be seen in figure 4.12. We can see that the white halo around the brain has disappeared, some artifacts of lesser importance have appeared though like the black 'hole' that appears just under the ventricles. These artifacts appear when the probabilities between two different output intensities are very close and the decision by the MRF solver amounts to a 'coin flip'. Overall the quality of the image has been improved.



(a) Maximum of the conditional probability (b) MRF smoothed image (c) Actual T2-MRI target Image

Figure 4.12: Effect of the MRF smoothing

4.5.1 Evaluation on brain MRI data set

In this thesis we will discuss the results on two type of multi-modal registrations: T1 to T2-MRI registration and T1 to PD-MRI registration, the reverse case of PD to T1-MRI

registration is discussed in [Michel 2010]. In each case we used 4 images for training, leaving 6 images in the testing database.

Evaluation of the registration is done in an inter-patient setting: one of the patient is taken as the target image and we register all of the 5 remaining images to this patient. In a clinical case we would have an image of one patient acquired with one modality and an image of another patient acquired with another modality. We would for instance have one T2 image for the target patient and 5 T1-MRI images to register it to. In such a setting, evaluation of the registration can be done with landmarks that have been placed by an expert (typically a clinician), or using segmentations that have been performed on the same organ on all the images to see if after deformation the segmentations align well, using the Dice coefficient for instance. The Dice coefficient measures the proportion of overlap between two regions compare to their overall area, if the two regions are denoted Ω_A and Ω_B the the dice coefficient $D(\Omega_A, \Omega_B)$ is:

$$D(\Omega_A, \Omega_B) = \frac{2\,|\Omega_A \cap \Omega_B|}{|\Omega_A| + |\Omega_B|} \tag{4.30}$$

This type of evaluation will be carried out using segmentation of the ventricles on all images.

But a second type of evaluation can also be carried out in our unusual setting. Since for each T1 image in our example we have access to the corresponding T2 image, we can also apply the deformation on this image and directly compare both the target T2 image and the deformed T2 image. This comparison will be carried out using the Mean absolute difference of images.

As we have seen, very few algorithms have used an approach similar to ours to solve this problem, so we will compare ourselves to state of the art multi-modal similarity measures such as Mutual Information (MI) and Normalized Mutual Information (NMI). As we have seen in section 2.1, registration is an ill-posed problem and this is only solved by the adjunct ion of a regularization term. Two solutions given by two different algorithms are only comparable at the same level of regularization. When we compare the mean absolute differences, a range of different values for the regularization parameter has been considered to remove the dependence on this parameter. When comparing the Dice coefficients we used another coefficient the *Harmonic energy* [Yeo 2009] to quantify the amount of regularization. The dice coefficient is often used in segmentation to compare two different segmentations. In registration we argue that looking at the increase in the dice coefficient before and after registration gives a much better view of the performance of the algorithm. Indeed when the starting Dice coefficient is very low, even the best registration algorithm will not get a Dice coefficient close to 1. Yet we can still compare the performance of different registration algorithms on such images by comparing the increase in the Dice

coefficient. Here only increases in the Dice coefficient are presented. Two transformation have the same amount of regularization when their harmonic energy are equal. The harmonic energy of a deformation field $\mathbf{u}$ writes:

$$HE(\mathbf{u}) = \frac{1}{|\Omega|} \int_{\Omega} \|J(\mathbf{u}(\mathbf{x}))\|^2 \, \mathrm{d}\mathbf{x} \tag{4.31}$$

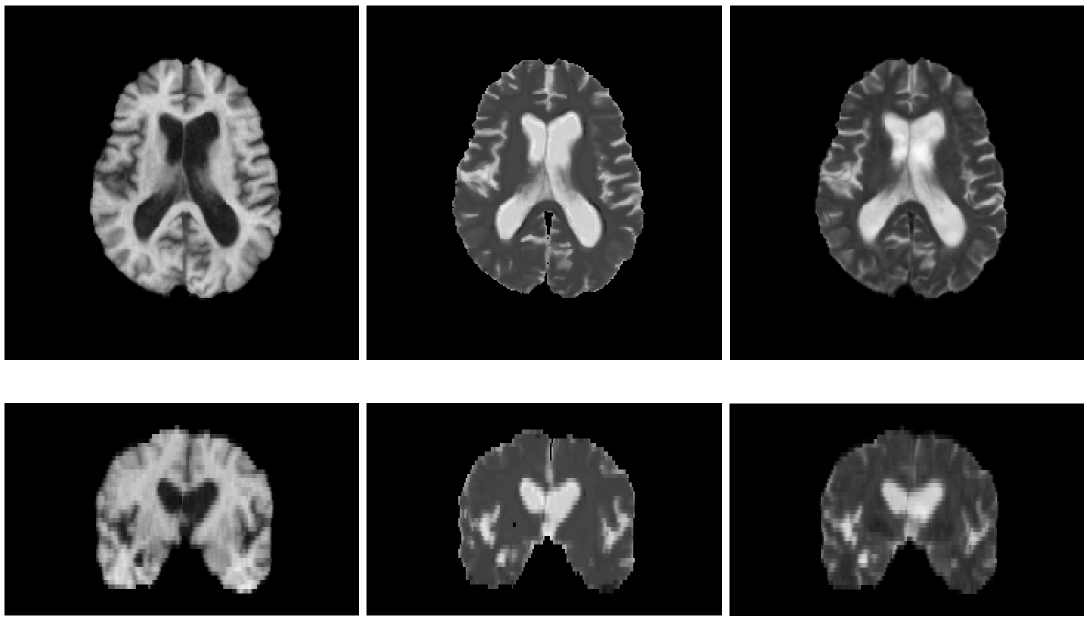
where $J(\mathbf{u}(\mathbf{x}))$ is the Jacobian of the deformation field. The lower the harmonic energy, the more rigid and smooth the transformation is. Again, since we have access to the T2 (in this example) images for all the patients, we can directly compare ourselves to the cases where we register those images uni-modally, which would be the oracle in this case. We performed 5 registration experiment for each harmonic energy level, for 20 different harmonic energy levels in total amounting to 200 registration experiment for each similarity measure.

T1 to T2-MRI image registration In figure 4.13 an examplar simulated image of the testing set is shown. Then our method is compared to mutual information and the ideal case of the unimodal SSD in figure 4.14. We can see that our algorithms performs much better than Mutual information on this data set and also that the results are very close to the uni-modal case which confirms the visual clue we have by looking at the simulated images. In figure 4.15 we compare the results on the dice cofficient for our method and the Normalized mutual information similarity measure. Showing again the superiority of our method over state of the art similarity measure at all harmonic energies.

T1 to PD-MRI image registration T1 to PD-MRI registration is a more challenging case due to the peculiar intensity distribution of the PD images. As can be seen in figure 4.16. Yet we can see in figure 4.17 and figure 4.18 that our method still yields better results than Mutual information and normalized mutual information, especially when we look at the ventricles segmentation performance since the simulation of this part is particularly accurate.

4.5.2 Evaluation on chest PET-CT data set

Here we present result images on a much harder data set, the chest PET-CT data set. Here the response intensities in the PET image are sometimes not related at all to the input intensities of the CT, but are the result of proton emissions induced by the tracer. This renders the task of simulating one image from the other impossible as can be seen in figure 4.19.



(a) T1-MRI source image (b) MRF smoothed image (c) Actual T2-MRI target Image

Figure 4.13: Image of the testing data set (same subject across all columns) after learning on 4 images

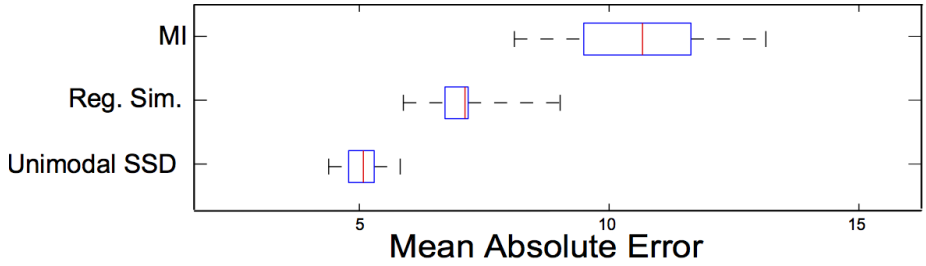


Figure 4.14: Boxplot showing the mean absolute differences between the deformed target image and the actual target image for mutual information, our metric and the ideal case of unimodal SSD.

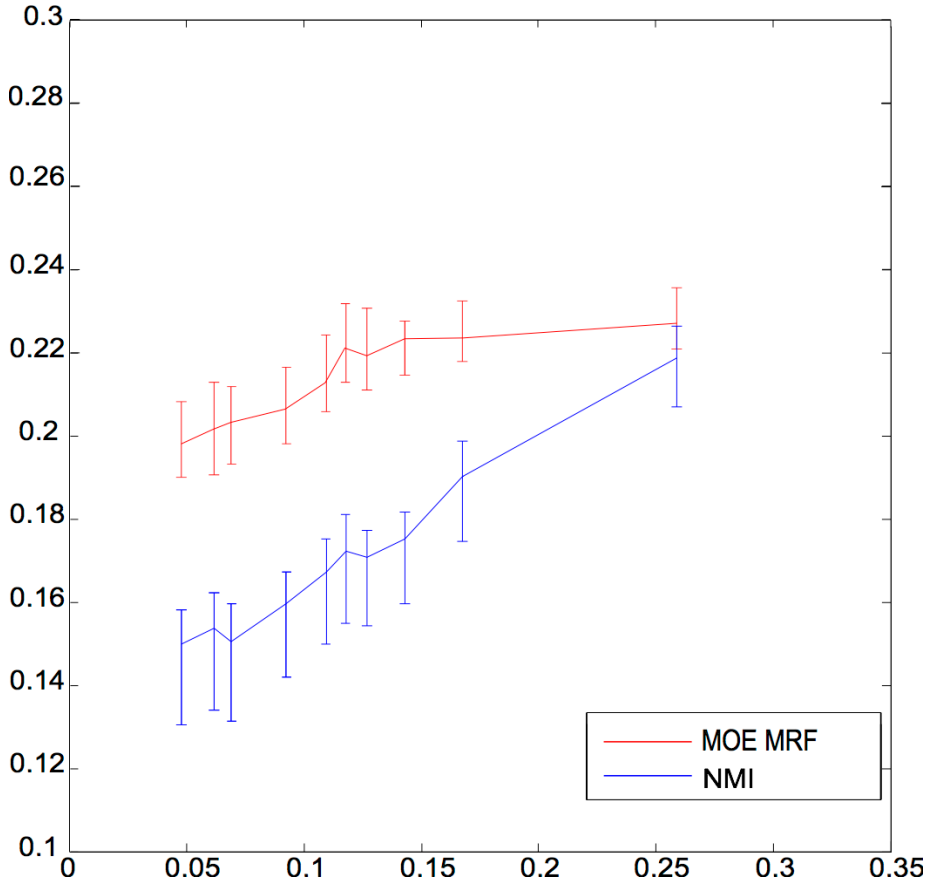
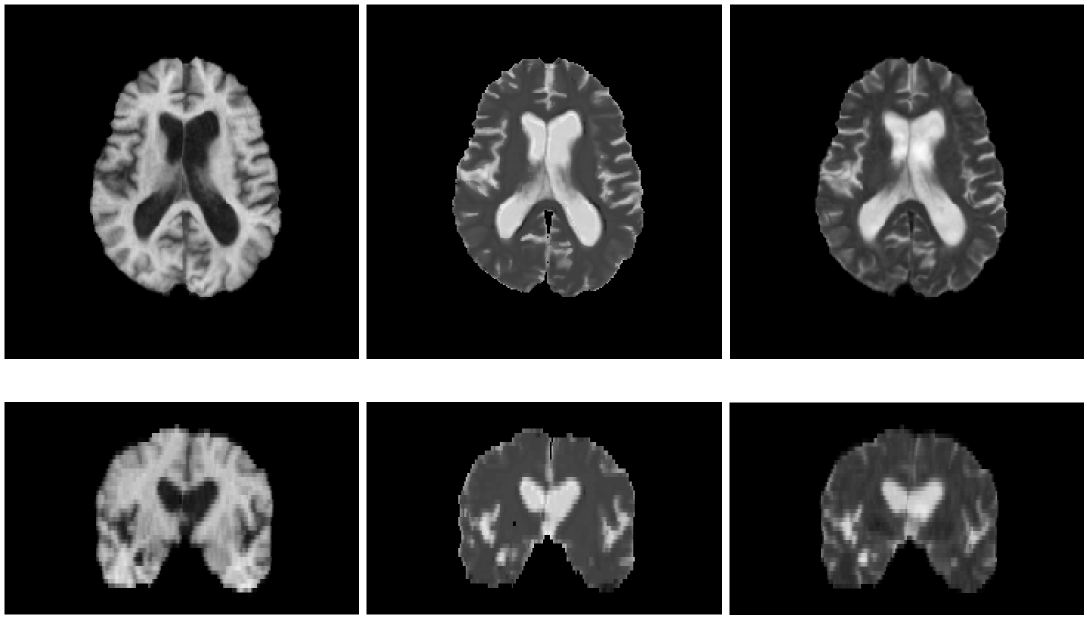


Figure 4.15: Evolution of the increase of the Dice coefficient as a function of the Harmonic Energy. The solid lines represent the average lines over all the experiments while the whiskers represent the lowest and highest values.



(a) T1-MRI source image (b) MRF smoothed image (c) Actual T2-MRI target Image

Figure 4.16: Effect of the MRF smoothing

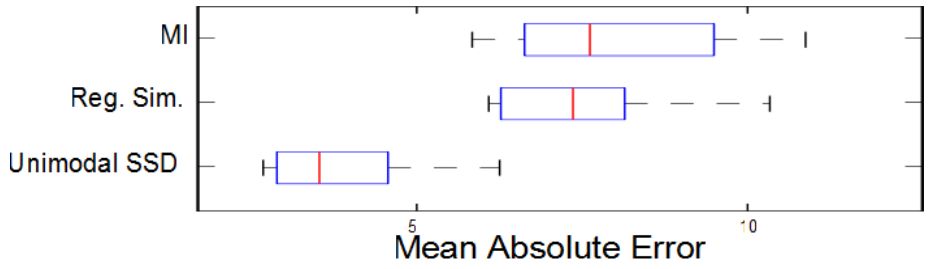


Figure 4.17: Boxplot showing the mean absolute differences between the deformed target image and the actual target image for mutual information, our metric and the ideal case of unimodal SSD.

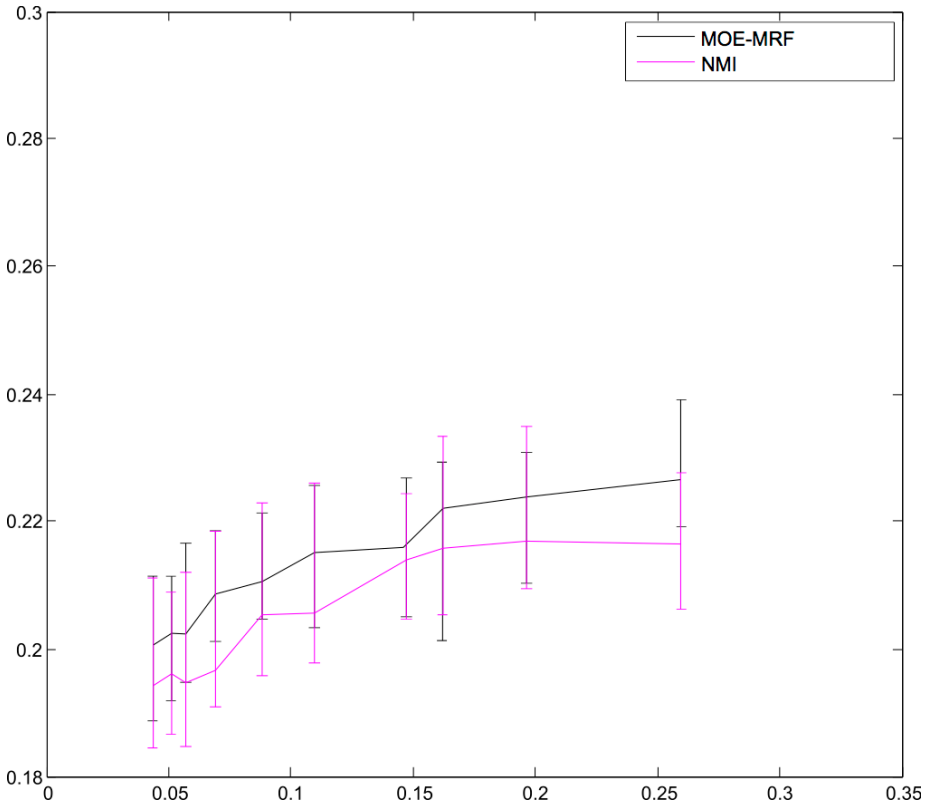
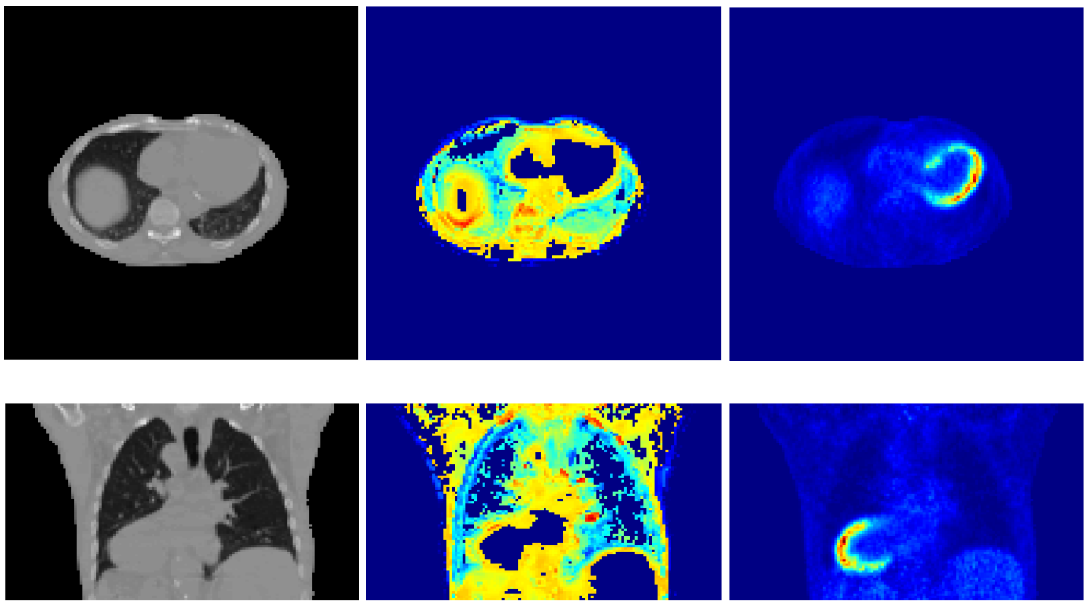


Figure 4.18: Evolution of the increase of the Dice coefficient as a function of the Harmonic Energy. The solid lines represent the average lines over all the experiments while the whiskers represent the lowest and highest values.



(a) CT source image (b) Result of the algorithm after MRF smoothing (c) Actual attenuation corrected PET target Image

Figure 4.19: Attempt on PET-CT data set, we would have liked the image in the midll to look as much as possible like the image on the right, both images have the same intensity scale.

Chapter 5

Metric Learning

Interest for *Metric Learning* (ML) started very recently and can be dated back to 2004. Furthermore if *Manifold Learning* techniques are counted as metric learning techniques, in that case, interest for ML dates back roughly to 1994. The idea of metric learning is fairly simple: given a data set of training data on which some sort of similarity information is given, either in the form of a full scalar distance between samples, or a ranking or some proximity of some samples to others or just a separation between samples that are deemed similar and dissimilar, metric learning techniques try to find a distance function that reproduces the relationships given on the training set, and generalizes well on unforeseen examples.

But before going deeper in the understanding of Metric Learning, we first need to mathematically define a distance function, and see some of the relaxations that can be done on this definition.

5.1 Distance Function

Let us consider a non–empty set $\mathcal{X}$, a distance function d on $\mathcal{X}$ is a mapping $d : \mathcal{X}\times \mapsto \mathbb{R}$ such that for all $x, y, z \in \mathcal{X}$ we have:

- symmetry: $d(x, y) = d(y, x)$
- identity of indiscernibles: $d(x, y) = 0 \iff x = y$
- triangle inequality: $d(x, z) \leq d(x, y) + d(y, z)$

from these properties, one has immediately the positivity property: $\forall x, y \in \mathcal{X},\ d(x, y) \geq 0$, since:

$$0 = d(x, x) \leq d(x, y) + d(y, x) = 2d(x, y)$$

A non–empty set endowed with a distance function d is called a *metric space*. A distance function is a very general function on a manifold. Distance functions can be considered a 'generalization' of the notion of norm for a *normed vector space*, and indeed, for any norm $\|\cdot\|_{\mathcal{X}}$, we can define a distance function d such as:

$$\forall x, y \in \mathcal{X},\ d(x, y) = \|x - y\|_{\mathcal{X}}$$

since in most case we work with normed vector spaces, we will rely heavily on this latter fact.

5.1.1 Relaxations to the notion of distance

Sometimes, distance functions are too restrictive, and only some of the properties stated before can be respected. Two relaxations are fairly common:

1. **pseudometrics**:
   - symmetry: $d(x, y) = d(y, x)$
   - **semi-separation**: $d(x, y) = 0 \Longrightarrow x = y$
   - triangle inequality: $d(x, z) \leq d(x, y) + d(y, z)$
2. **quasimetrics**:
   - identity of indiscernibles: $d(x, y) = 0 \iff x = y$
   - triangle inequality: $d(x, z) \leq d(x, y) + d(y, z)$

   In this case, symmetry is just dropped.

5.1.2 Examples of distance functions

By far the most used distance function in computer vision is the **euclidean distance** on vector spaces. This distance is of course based on the Euclidean norm denoted here as L_2 norm, or $\|\cdot\|_2$, in the case where the space is of finite dimension, it is not unusual to refer to the ℓ_2 norm, or $\|\cdot\|_{\ell_2}$, to specify the summation is finite. In this work, only finite summations will be considered. The euclidean distance on the vector space $\mathcal{X}$ of dimension N, is expressed as:

$$\forall \mathbf{x}, \mathbf{y} \in \mathcal{X} \quad d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_{\ell_2} = \sqrt{(\mathbf{x} - \mathbf{y})^T (\mathbf{x} - \mathbf{y})} = \sqrt{\sum_{i=1}^{N} (\mathbf{x}_i - \mathbf{y}_i)^2} \tag{5.1}$$

Another very well known distance function, that plays a very important role in Metric Learning is the **Mahalanobis distance**. In the assumption that all elements of $\mathcal{X}$ are generated from the same distribution, we can reliably estimate the expectation of a random variable vector $\mathbf{X}$, here the expectation of $\mathbf{X}$ is denoted as $\mathbb{E}[\mathbf{X}]$. Then the covariance matrix of $\mathbf{X}$ denoted here S, is the element wise variance and covariance matrix of $\mathbf{X}$:

$$S = \mathbb{E}\left[(\mathbf{X} - \mathbb{E}[\mathbf{X}])(\mathbf{X} - \mathbb{E}[\mathbf{X}])^T\right] \tag{5.2}$$

It has to be noted that S is a *positive semi-definite matrix* (p.s.d.), which is characterized by a non-negative spectrum (the set of its eigenvalues). P.S.D matrices can be decomposed in the product of their square roots meaning that there exists a square matrix R such that:

$$S = RR^T \tag{5.3}$$

Also if S is positive definite (positive spectrum), then S is invertible and R is also invertible. For the purpose of this work we will assume the covariance matrix to be invertible as long as the sample size is sufficiently large (at least larger than the dimension of the space). The case where the matrix is not invertible is a degenerate case where some dimensions follow Dirac distributions, this is solved numerically (inversion in the image space of the covariance matrix through singular value decomposition).
The Mahalanobis distance is a reweighting of the euclidean distance in such a way that all dimensions in the vector space play a role according to the spread of their distribution:

$$\forall \mathbf{x}, \mathbf{y} \in \mathcal{X} \quad d(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T S^{-1} (\mathbf{x} - \mathbf{y})} \tag{5.4}$$

This distance is already very useful as is, but using equation (5.3), one can see that the Mahalanobis distance can be interpreted in terms of the euclidean distance:

$$\begin{aligned}
\sqrt{(\mathbf{x} - \mathbf{y})^T S^{-1} (\mathbf{x} - \mathbf{y})} &= \sqrt{(\mathbf{x} - \mathbf{y})^T (RR^T)^{-1} (\mathbf{x} - \mathbf{y})} && (5.5)\\
&= \sqrt{(\mathbf{x} - \mathbf{y})^T R^{T^{-1}} R^{-1} (\mathbf{x} - \mathbf{y})} && (5.6)\\
&= \sqrt{(R^{-1}(\mathbf{x} - \mathbf{y}))^T R^{-1} (\mathbf{x} - \mathbf{y})} && (5.7)\\
&= \left\| R^{-1}\mathbf{x} - R^{-1}\mathbf{y} \right\|_{\ell_2} && (5.8)
\end{aligned}$$

This last equation shows that the Mahalanobis distance is just the euclidean distance performed in the transformed space $R^{-1}\mathcal{X}$. In this space, all elements are in a distance that is proportionate to the spread of their distributions.

5.1.3 Kernels and RKHS

The notion of distance is way more versatile than the absolute value of a vector in a feature space [Smola 1998]. It allows to develop methods without knowing the exact representation of an element in the feature space but only a comparison of that element to other elements in the feature space. A kernel is such a comparison function, a kernel k is a function from $\mathcal{X} \times \mathcal{X}$ to $\mathbb{R}$, and a set of n elements $\{x_1, \dots, x_n\}$ is represented by the $n \times n$ matrix K, the general element of which is:

$$K_{i,j} = k(x_i, x_j) \tag{5.9}$$

The kernel k is positive definite on $\mathcal{X}$ if and only if kernel matrix K is positive semi definite for any subset of $\mathcal{X}$. The *Aronszajn* theorem states that:

Theorem 1. *The kernel k is positive definite if and only if there exists a Hilbert space $\mathcal{H}$ and a mapping $\phi : \mathcal{X} \mapsto \mathcal{H}$ such that for any $\mathbf{x}, \mathbf{x}'$ in $\mathcal{X}$*

$$k(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle_{\mathcal{H}} \tag{5.10}$$

Here $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ is the inner product on $\mathcal{H}$. This theorem is at the source of all kernel methods, it allows to have access to a possibility infinite feature space $\mathcal{H}$ and to the projection ϕ without explicitly knowing one or the other, only the kernel expression is needed.

Kernel methods also rely on another important result on the *Reproducing Kernel Hilbert Spaces* (RKHS).

Definition 1. *The kernel k is called a reproducing kernel of $\mathcal{H}$ if:*

1. *$\mathcal{H}$ contains all functions of the form:*

$$\forall \mathbf{x} \in \mathcal{X}, \quad k_{\mathbf{x}} : \mathbf{t} \mapsto k(\mathbf{x}, \mathbf{t})$$

2. *for every $\mathbf{x} \in \mathcal{X}$ and $f \in \mathcal{H}$:*

$$f(x) = \langle f, k_{\mathbf{x}} \rangle_{\mathcal{H}}$$

If a reproducing kernel exists, then $\mathcal{H}$ is called a Reproducing Kernel Hilbert Space.

A reproducing kernel is unique to a RKHS and conversely. Lastly, we have the result:

Theorem 2. *A kernel k is positive definite if and only if it is a reproducing kernel*

Finally, the *kernel trick* links a kernel to a distance function, considering the previous results we have:

$$\begin{aligned} d(\mathbf{x},\mathbf{y}) &= \sqrt{\|\mathbf{x}-\mathbf{y}\|_{\mathcal{H}}^2} \\ &= \sqrt{\|\mathbf{x}\|_{\mathcal{H}}^2 + \|\mathbf{y}\|_{\mathcal{H}}^2 - 2\langle \mathbf{x},\mathbf{y}\rangle_{\mathcal{H}}} \\ &= \sqrt{k(\mathbf{x},\mathbf{x}) + k(\mathbf{y},\mathbf{y}) - 2k(\mathbf{x},\mathbf{y})} \end{aligned} \tag{5.11}$$

Examples of positive definite kernels: some kernels are very widely used since they can introduce a lot of non-linearity and are very easy to implement, among those two stand out:

1. For any $p \in \mathbb{N}$ the polynomial kernel is positive definite:

$$k(\mathbf{x},\mathbf{y}) = \langle \mathbf{x},\mathbf{y}\rangle^p \tag{5.12}$$

   where $\langle \cdot,\cdot \rangle$ is the euclidean inner product.

2. The Gaussian radial basis function, also known as the Gaussian kernel is positive definite:

$$k(\mathbf{x},\mathbf{y}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}\right) \tag{5.13}$$

5.2 Metric Learning and Space Embedding

As we saw earlier, be it with the Mahalanobis distance or with kernel derived distances, metric learning and space embedding are intrinsically linked. Defining and learning a metric amounts in most cases to learn a projection into a new space where the elements have the desired similarity properties. Two main approaches can be distinguished in the literature. *Unsupervised learning* approaches where there is no more information given to us than the training samples, which are often represented as a set of feature vectors $\{\mathbf{x_1},\ldots,\mathbf{x_N}\} \subset \mathcal{X}$. This is opposed to *Supervised learning* approaches, where the sampled training data is enriched with information given by the user, this information can come in the form of labels in the cases of classification where given a set of samples and their classes one tries to recover the class of a new sample, in this case, the training set is of the form $\{(\mathbf{x_i},\ell_i)|i \in \{1,\ldots,N\}\} \subset \mathcal{X}\times\mathbb{N}$. But in the specific case of metric learning,

data can also come with similarity information, and for instance we are given two sets of pairs of samples, the dissimilar and the similar pairs. This information is given by the user before training and the role of the algorithm is to identify the structure of similarity and given two new samples be able to identify them as similar or dissimilar. Supervised learning refers to all the algorithms where on top of the data there is a user defined structure that the algorithms tries to identify and recreate.

A good, but rather outdated survey on *Metric Learning* can be found in [Yang 2006]. Let us now review some fundamental and state of the art *Metric Learning* algorithms and space embedding algorithms.

5.2.1 Unsupervised Learning

As we have seen from equation (5.5), *Metric Learning* and space embedding are closely linked, since learning a metric in turn might just amount to finding an embedding space in which the properties we are interested in are projected. Unsupervised learning for metric learning is the most primitive form of learning since we can't impose information constraints on the learned space, but this set of techniques come in handy when this supplementary information is too costly to produce or that only recovering the intrinsic structure of the space might suffice. This discipline is named *Manifold Learning* and an extensive literature describes it. Manifold Learning techniques all revolve around the central idea that the data in the highly dimensional space $\mathcal{X}$ is in actuality laying on a much lower dimension manifold, which theses techniques try to recover. This idea is traditionally represented with the "Swiss roll" exemplar problem, as illustrated in figure (5.1): in this example the data is embedded in a 3D space, yet obviously is supported by a 2D manifold which is this seemingly rolled sheet of paper, the 'Swiss roll'. If we try to compare two points in the 3D space using the euclidean distance, two points that are completely different due to the inherent structure of the roll might appear very close. Manifold Learning techniques are used to unroll the manifold and embed it in its true space. In this space, the euclidean distance should be a good approximate of the inherent structured distance of the data.

It is out of the scope of this thesis to fully describe the extent of Manifold Learning, instead we are going to describe the most popular Manifold Learning algorithms and give an intuition as to how they can be used in the same way as metric learning techniques.

Multi–Dimensional Scaling (MDS) [Cox 2001] is one of the first successful manifold embedding algorithms. The main idea of MDS is to only rely on the inner distances of the training set, thus removing all considerations on the samples themselves. MDS finds the embedding with the lowest dimension that maps the samples all while keeping their relative distances. In theory, if the original embedding is of dimension M, then a distance

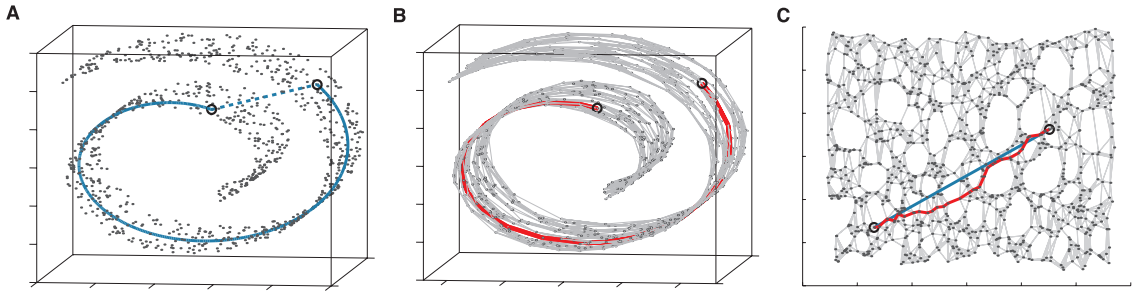


Figure 5.1: Figure extracted from [Tenenbaum 2000]: Left: Manifold distance compared to euclidean distance, Middle: Geodesic distance as used by ISOMAP, Right: unrolling of the 'Swiss roll' and both Manifold distance and geodesic distances compared.

preserving embedding is of maximum dimension $M-1$. Such a space reduction is not enough in practice, so irrelevant dimensions to the distance are discarded. In [Cox 2001], Cox *et al.* show how to go from a matrix of distances to a matrix of inner products between points of the embedding, to finally the positions of the actual points in the embedding, up to some isomorphic (that doesn't operate on distances) transformations. Then a development is made to handle discrepancy matrices, which are matrices of user defined distances. Let us have a look at how this is done:

1. start with a matrix $D = \{D_{ij}\}$ of the inner–sample discrepancies.
2. Compute the intermediate matrix $A = \{A_{ij}\} = \{-\frac{1}{2}D_{ij}^2\}$
3. Find the matrix of inner-products $B = \{B_{ij}\} = \{A_{ij} + \frac{1}{N}\sum_i A_{ij} + \frac{1}{N}\sum_i A_{ij} + \frac{1}{N^2}\sum_{ij} A_{ij}\}$
4. Find the eigenvalues $\lambda_1, \ldots, \lambda_{N-1}$ and associated eigenvectors $\mathbf{v_1}, \ldots, \mathbf{v_{N-1}}$ and $\forall i, \mathbf{v_i}^T\mathbf{v_i} = \lambda_i$. Eigenvalues are in a descending order.
5. Choose an appropriate number of dimensions p, and the coordinates of the points are given by $v_{ij}, i \in 1, \ldots, p; j \in 1, \ldots, N$

One has to note that if D is a matrix of distances, then MDS is very close to PCA.

Isomap [Tenenbaum 2000]is one of the first non-linear manifold learning techniques able to deal with the swiss roll problem (figure 5.1). The idea of Isomap is to extend on MDS, by using geodesic distances extracted in the original space.

The algorithm works in 3 steps:

1. Using K-nearest neighbors, an adjacency graph is constructed, in which the weight of the edges is $d_{\mathcal{X}}(\mathbf{x}_i, \mathbf{x}_j)$ (the euclidean distance) if i and j are neighbors, and infinity if they are not.

2. To find the geodesic distances $d_{\mathcal{H}}(\mathbf{x}_i, \mathbf{x}_j)$, shortest paths are computed: initialize

$$\forall i, j, \quad d_{\mathcal{H}}(\mathbf{x}_i, \mathbf{x}_j) = d_{\mathcal{X}}(\mathbf{x}_i, \mathbf{x}_j)$$

   Then $\forall i, j$ and each value of k in $1, \ldots, N$ set

$$d_{\mathcal{H}}(\mathbf{x}_i, \mathbf{x}_j) = \min\left(d_{\mathcal{H}}(\mathbf{x}_i, \mathbf{x}_j), d_{\mathcal{H}}(\mathbf{x}_i, \mathbf{x}_k) + d_{\mathcal{H}}(\mathbf{x}_k, \mathbf{x}_j)\right)$$

3. Use MDS with the matrix $D = \{D_{ij}\} = \{d_{\mathcal{H}}(\mathbf{x}_i, \mathbf{x}_j)\}$

As with any non-linear algorithm, there is the introduction of a non-linearity parameter, here it is the K of KNN. The setting of K can be crucial in finding the right embedding, as setting K too small might lead to over–fitting of the manifold noise, and setting K too large might not get as close to the true geodesic distance on the manifold.

Locally linear embedding (LLE) [Roweis 2000] finds an embedding that focuses on the local structure in the manifold. The construction is done in such a way that only geometrical properties of the manifold are kept, so the final embedding mimics the local geometric relationship between samples.

First, for each sample $\mathbf{x}_i \in \mathcal{X}$ of dimension d, k neighboring samples are selected constituting $\mathcal{N}_i$, the projection of $\mathbf{x}_i$ onto the new space will only rely on these k neighbors. Then a weighting parameter is learned such that the geometric interactions of $\mathbf{x}_i$ with its neighbors are preserved. This is done by minimizing the following cost:

$$\begin{aligned} \min_{W} \quad & \sum_i \left(\mathbf{x}_i - \sum_{j \in \mathcal{N}_i} W_{ij}\mathbf{x}_j\right)^2 \\ \text{s.t.} \quad & \sum_{j \in \mathcal{N}_i} W_{ij} = 1 \end{aligned} \tag{5.14}$$

The weight W_{ij} characterizes the contribution of the j^{th}sample to the reconstruction of $\mathbf{x}_i$. By design W is rotation and scale invariant, and the sum to 1 constraint makes W translation invariant. In this way, W only captures intrinsic geometric properties of the

local embedding. Reconstruction of the new embedding is done in the same way, the new set of vectors $h_i \in H$ of dimension $m \ll d$ is found by minimizing the functional:

$$\min_{H} \quad \sum_i \left(\mathbf{h}_i - \sum_{j \in \mathcal{N}_i} W_{ij} \mathbf{h}_j \right)^2 \tag{5.15}$$

$$\text{s.t.} \quad \begin{cases} \frac{1}{N} \sum_i \mathbf{h}_i \mathbf{h}_i^T = I \\ \sum_i \mathbf{h}_i = 0 \end{cases}$$

Where I is the identity matrix. The first constraint prevents from having degenerate representation while the second constraint prevents translations around the embedding centroid. Note here that by setting $W_{ij} = 0$ when $j \notin \mathcal{N}_i$, then we can define the $d \times d$ matrix W of general element W_{ij}, and we have

$$\sum_i \left(\mathbf{x}_i - \sum_{j \in \mathcal{N}_i} W_{ij} \mathbf{x}_j \right)^2 = \sum_i \left(\mathbf{x}_i^T \left(I - W \right)^T \left(I - W \right) \mathbf{x}_i \right)^2 \tag{5.16}$$

$$= \sum_i \left\| \mathbf{x}_i \right\|^2_{(I-W)^T (I-W)} \tag{5.17}$$

With equation (5.16), we can see that finding W is effectively minimizing the Mahalanobis weighted norm of the samples and effectively leads to a Mahalanobis weighted distance between the samples in the learned space.

Laplacian Eigenmaps [Belkin 2003] is another manifold learning technique, that focuses on local interactions just like LLE. The use of the *Laplace-Beltrami* opreator and the heat kernel allows to smooth out irregularities and noise in the manifold which leads to a smoother and more consistent embedding. Laplacian eigenmaps, first build an adjacency graph, weighted by heat kernels and the embedding is computed using the Laplacian-Beltrami operator:

1. Construct the adjacency graph using K-nearest neighbors by putting an edge between sample i and sample j if i and j are neighbors.

2. Choose the weighting of the graph, either soft weighting:

$$\begin{cases} W_{ij} = \exp\left(-\dfrac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma}\right) & \text{if } i \text{ and } j \text{ are connected} \\ W_{ij} = 0 & \text{otherwise} \end{cases}$$

or hard weighting:

$$\begin{cases} W_{ij} = 1 & \text{if } i \text{ and } j \text{ are connected} \\ W_{ij} = 0 & \text{otherwise} \end{cases}$$

3. Have $D = \{D_{ii}\} = \{\sum_j W_{ji}\}$ the diagonal matrix of weights, and $L = D - W$, the laplacian matrix, and compute the eigenvalues λ_i and eigenvectors $\mathbf{v}_i$ to the generalized eigenvector problem, sorted in ascending order according to their eignevalues:

$$L\mathbf{v} = \lambda D\mathbf{v}$$

$\mathbf{v}_0$ is dropped since linked to eigenvalue 0, the next r values are kept for an embedding in r dimensions. The representer $\mathbf{h}_i$ of $\mathbf{x}_i$ in the new space is the vector:

$$\mathbf{h}_i = (\mathbf{v}_1(i), \ldots, \mathbf{v}_m(i))$$

It can be shown that this embedding minimizes the objective function:

$$\sum_{ij} \|\mathbf{h}_i - \mathbf{h}_j\|^2 W_{ij} \tag{5.18}$$

under appropriate constraints. This objective function penalizes neighboring points that are mapped far apart in the embedding according to their weight W.

The missing link of all the Manifold Learning methods with metric learning, is the ability to reproduce the result on unforeseen examples. This is called the *out-of-sample problem* or how to project a new point onto the already existing embedding. In the general case, this is an open problem, but some manifold learning techniques have been provided with out-of-sample extensions. And this is the case for all the manifold learning algorithms presented here. In [Bengio 2004] Bengio *et al.* provide an out-of-sample extension for LLE, ISOMAP, MDS, eigenmaps and spectral clustering, by first acknowledging a common framework to the methods and then devising a weighting function used to map new points in the embedding.

Relevant components analysis (RCA) [Shental 2006]is interesting because it is one of the works that sits in between Unsupervised and Supervised learning, it is referred to by the

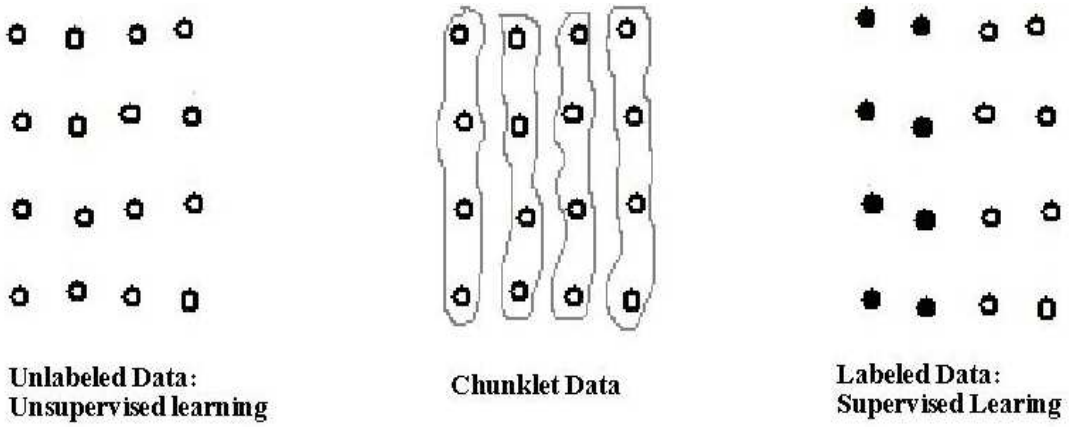


Figure 5.2: Figure extracted from [Shental 2006]

authors as *Adjustment Learning*. Adjustment learning is mainly targeted at classification. The main idea is that the only information we have on the data set is that it is organized in chunks or *chunklets* of data, in which all elements have the same unknown label (see figure 5.2). A chunklet may contain only one element.

Considering we have M chunklets, and $P_m,\ m \in 1, \ldots, M$ samples per chunklet, the mean sample over each chunklet is represented by $\hat{\mu}_m,\ m \in 1, \ldots, M$ and a sample from chunklet m is represented by $\mathbf{x}_m^i,\ i \in 1, \ldots, P_m$ then RCA is defined as follows:

1. Compute S_{ch}:
$$S_{ch} = \frac{1}{|\Omega|} \sum_{m=1}^{M} \sum_{i=1}^{P_m} \left(\mathbf{x}_m^i - \hat{\mu}_m\right) \left(\mathbf{x}_m^i - \hat{\mu}_m\right)^T$$
and find r the number of singular values of S_{ch} that are significantly larger than 0.

2. Compute S_T the total covariance matrix of the original data and project the data using PCA to its r largest dimensions.

3. Project S_{ch} onto the reduced dimensional space, and compute the corresponding whitening transformation R as the square root matrix of S_{ch}

4. Apply R to the original data in the reduced space

This transformation magnifies the directions that are relevant to each chunklet, while masking inner-chunklet variability. PCA is done to not magnify directions that have no spread. A kernelized version of RCA was proposed in [Tsang 2005]. Later in [Bar-Hillel 2003,

Bar-Hillel 2006] it has been shown that a full fledged metric can be devised by first doing Fisher Discriminant Analysis in the original space followed by RCA in the reduced space, metric which maximizes the mutual information between the original data and its representation in the embedding.

5.2.2 Supervised Learning

Supervised learning is the best setting for metric learning. Given a sample data set, and some similarity information on the samples, we are aiming to learn a distance function that reproduces this sense of similarity on unforeseen samples. The information of similarity can be given in various ways we will see. A very popular way of expressing the similarity is by considering two sets of pairs:

$$S = \{(x_i, x_j) \in \mathcal{X}^2, (i,j) \in \{1, \dots, N\} \quad \text{s.t. } x_i \text{ is semantically close to } x_j\}$$

and

$$D = \{(x_i, x_j) \in \mathcal{X}^2, (i,j) \in \{1, \dots, N\} \quad \text{s.t. } x_i \text{ is semantically distant from } x_j\}$$

S and D are user defined. The first successfull attempt at solving this problem was made in [Xing 2002], where the learning of a Mahalanobis–like distance is made:

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x}_i - \mathbf{x}_j\|_A = \sqrt{(\mathbf{x} - \mathbf{y})^T A (\mathbf{x} - \mathbf{y})} \tag{5.19}$$

The notion of distance is relaxed here to a pseudo–metric as the matrix A is only required to be positive semi–definite. Matrix A is found by minimizing the following problem:

$$\min_A \quad \sum_{(x_i,x_j)\in S} \|\mathbf{x}_i - \mathbf{x}_j\|_A^2 \tag{5.20}$$

$$\text{s.t.} \quad \begin{cases} \sum_{(x_i,x_j)\in D} \|\mathbf{x}_i - \mathbf{x}_j\|_A \geq 1 \\ A \quad \text{p.s.d.} \end{cases}$$

Matrix A is found such that the Mahalanobis distance between the similar pairs is as small as possible while the the distance between dissimilar pairs is maintained reasonably large. The arbitrary constant 1 can be changed but only resulting in a scaling of the matrix. The problem is obviously convex and the constrains are also convex. The resolution of this problem is done iteratively in two steps, first the matrix is found by gradient descent, then projection is done on the two constrain sets.

A kernelized version of this method has been presented in [Kwok 2003], where $\mathbf{x}$ and $\mathbf{y}$ are first projected in feature spaces, allowing for non-linearity in the metric, the optimization method is also revized with the use of linear programming with the objective of circumventing the problem of local minima that can arise from gradient descent.

More recently, Goldberger *et al.* proposed *Neighbourhood Component Analysis* (NCA) [Goldberger 2004]. Even though NCA was originally designed for *k-nearest neighbor classification* (KNN), NCA was used successfully in other applications [Wang 2008, Keller 2006] and even in an unsupervised setting [Yuan 2007]. Given a training set of labeled data $S_l = \{(\mathbf{x}_1, \ell_1), \ldots, (\mathbf{x}_N, \ell_N) \in \mathcal{X} \times \{C_1, \ldots, C_K\}\}$, where $\{C_1, \ldots, C_K\}$ are the K different classes, NCA learns a Mahalanobis–like distance weighted by matrix $A = Q^T Q$ that optimizes the leave one out performance of KNN on the training data. Neighbourhood assignment in this case is made differentiable with a soft-max activation function:

$$\begin{cases} p_{ij}^A &= \dfrac{\exp\left(-\|Q\mathbf{x}_i - Q\mathbf{x}_j\|^2\right)}{\displaystyle\sum_{k \neq i} \exp\left(-\|Q\mathbf{x}_i - Q\mathbf{x}_k\|^2\right)} \\ p_{ii}^A &= 0 \end{cases} \tag{5.21}$$

Maximizing the leave-one-out performance of KNN under soft-max is the same as maximizing the expected number of points correctly classified under soft-max, as a function of Q:

$$f(Q) = \sum_i \sum_{j \in C_i} p_{i,j}^A \tag{5.22}$$

In practice, f is maximized using a gradient based optimizer. Which is again sensitive to local minima and initialization.

Following the work of [Goldberger 2004] on NCA, Globerson and Roweis show a convex alternative to this method [Globerson 2006] with *Maximally Collapsing Metric Learning* (MCML). The whole idea of MCML is that in the same class, samples should be mapped infinitely close to each other by the distance and in between classes, as far away as possible. This is done by minimizing the Kullback-Leibler distance between p_{ij}^A of NCA and a p_{ij}^0 defined as:

$$p_{ij}^0 = \begin{cases} 1 & \mathbf{x}_i = \mathbf{x}_j \\ 0 & \mathbf{x}_i \neq \mathbf{x}_j \end{cases} \tag{5.23}$$

The objective function we try to minimize is then:

$$\min_{A} \quad D_{KL}\left(p_{ij}^{A} \middle\| p_{ij}^{0}\right) \tag{5.24}$$

$$\text{s.t.} \qquad A \quad \text{p.s.d.}$$

This function is convex in A with convex constraint, so the minimization converges to a global optimum. Here optimization is done with gradient based optimizer followed by projection on the space of P.S.D. matrices.

Scalability The main shortcoming of the previously mentioned algorithms stems from the fact that the computational complexity grows quadratically with the number of samples which becomes intractable very quickly. Also the estimation of the eigenvector problems linked to the projection on the space of p.s.d matrices is of cubic complexity with respect to the dimension of the space. These two major problems introduce the search for algorithms that are scalable to the number of data and their dimensions.

One of the first algorithm to deal with scalability is *Similarty Sensitive Hashing* (SSH) [Shakhnarovich 2005, Ren 2005]. SSH finds a mapping $H : \mathcal{X} \to \mathcal{H}$ such that the L_1 distance between samples in $\mathcal{H}$ reflects a user defined similarity that is given in the form of labels given to pairs of samples. Pairs of samples in S are given label 1 and pairs of samples in D are given label -1. $\mathcal{H}$ is defined as a *Hamming* space, i.e. a space of binary vectors, here, the vectors are considered weighted by a parameter vector $\boldsymbol{\alpha}$:

$$H(\mathbf{x}) = [\alpha_1 h_1(\mathbf{x}), \ldots, \alpha_M h_M(\mathbf{x})] \tag{5.25}$$

With the projection function h defined as a threshold function. Here we will only display the case were h is a linear projection function of $\mathbf{x}$, with matrix R, and R_m a row vector of R:

$$h_m(\mathbf{x}) = \begin{cases} 1 & \text{if } R_m^T \mathbf{x} \leq 0 \\ 0 & \text{if } R_m^T \mathbf{x} \geq 0 \end{cases} \tag{5.26}$$

A *weak classifier* can then be written as:

$$\begin{aligned} c_m(\mathbf{x}_i, \mathbf{x}_j) &= \begin{cases} 1 & \text{if } h(\mathbf{x}_i) = h(\mathbf{x}_j) \\ 0 & \text{otherwise} \end{cases} \\ &= sign\left(R_m^T \mathbf{x}_i\right) sign\left(R_m^T \mathbf{x}_j\right) \end{aligned} \tag{5.27}$$

We then have the identity:

$$\begin{aligned} \|H(\mathbf{x}_i) - H(\mathbf{x}_j)\|_{\ell_1} &= \sum_{m=1}^{M} \alpha_m \left|h_m(\mathbf{x}_i) - h_m(\mathbf{x}_j)\right| \\ &= \frac{M}{2} - \frac{1}{2}\sum_{m=1}^{M} \alpha_m c_m(\mathbf{x}_i, \mathbf{x}_j) \end{aligned} \tag{5.28}$$

this last equation made the *strong classifier* appear:

$$C(\mathbf{x}_i, \mathbf{x}_j) = sign\left(\sum_{m=1}^{M} \alpha_m c_m(\mathbf{x}_i, \mathbf{x}_j)\right) \tag{5.29}$$

The optimization strategy on C follows the boosting approach [Schapire 1999a, Collins 2002]. At each iteration, optimization on a new c_m is done that maximizes the correlation between labels and prediction. Then the misclassified samples are boosted with a weight to have more effect on the optimization of c_m at the next iteration. The α_m are also computed by optimization in the boosting procedure. Boosting is known to be scalable for a large number of samples and has been shown to have excellent generalization performance, without a tendency to over-fitting. However, boosting is very sensitive to label noise, and the overall metric isn't differentiable due to the use of the $L1$ distance.

One of the first Mahalanobis-like distances that takes into account the scalability is *Information Theoretic Metric Learning* (ITML) [Davis 2007] in which the distance is interpreted in terms of the Gaussian distribution of mean $\boldsymbol{\mu}$ that generates it:

$$p(\mathbf{x}; A) = \frac{1}{Z}\exp\left(-\frac{1}{2}\|\mathbf{x} - \boldsymbol{\mu}\|_A^2\right) \tag{5.30}$$

Given this, they propose to find the matrix A yielding a distribution as close as the one given by a chosen matrix A^0 (e.g. $A^0 = I$ for a distance as close as possible to the euclidean distance) under the constraints of separation of S and D. This is done with the use of the Kullback-Liebler divergence with the optimization of the problem:

$$\min_{A} \quad D_{KL}\left(p(\mathbf{x}; A)\,\middle\|\,p(\mathbf{x}; A^0)\right) \tag{5.31}$$

$$\text{s.t.} \quad \begin{cases} \|\mathbf{x}_i - \mathbf{x}_j\|_A^2 \leq u & (i,j) \in S \\ \|\mathbf{x}_i - \mathbf{x}_j\|_A^2 \geq v & (i,j) \in D \end{cases}$$

In order to ensure that the solution always exists, slack variables are introduced. The optimization method repeatedly computes projections of the current solution onto a single

constraint with an update on the matrix A. The complexity of the algorithm is in $\mathcal{O}(cd^2)$ where c is the number of constraints and d the dimensionality of the space $\mathcal{X}$.

Large Margin Methods One of the major shortcomings of the distance metric learning method thus far is their lack of robustness to label and sample noise. However real life cases always display sample and label noise, be it because the operator mislabeled a sample or because the sample itself is subject to measurement noise. *Large Margin Methods* (LMM) for metric learning are an attempt to deal with the latter problem. The idea behind large margin methods is to put as much distance as possible between the samples and the decision frontier, this distance is called the margin. If a new sample is corrupted by noise, then chances are that it still falls on the right side of the decision frontier since it is supposedly closer to similar samples. LMM have proven to be more successful and more robust to noise than their counterparts

One of the first *Large Margin Metric Learning* algorithm was presented in [Weinberger 2006, Weinberger 2009] with *Large Margin Nearest Neighbor* (LMNN) classification. This algorithm was originally designed for K-Nearest Neighbor classification. LMNN finds a mapping and a Mahalanobis distance function such that the nearest neighbors of a sample $\mathbf{x}_i$ are effectively its target neighbors, identified by the fact that they have same label. The idea of LMNN is to establish a perimeter around each sample in which there are no impostors, *i.e.* no samples should be of different class in this perimeter. Even more, the distance between impostors and the perimeter (the margin) should be maintained large. During learning, the impostors are pushed out of the perimeter with sufficient margin, while target neighbors are pulled in the perimeter:

$$\min_A \quad \underbrace{(1-\lambda) \sum_{i,j \in \mathcal{N}(i)} \|\mathbf{x}_i - \mathbf{x}_j\|_A}_{\text{pull}} + \underbrace{\lambda \sum_{i,j \in \mathcal{N}(i)} \sum_k (1 - \delta_{\ell_i = \ell_k}) \max\left(1 + \|\mathbf{x}_i - \mathbf{x}_j\|_A - \|\mathbf{x}_i - \mathbf{x}_k\|_A, 0\right)}_{\text{push}} \quad (5.32)$$

$\max(x, 0)$ is the standard hinge loss function, if the discrepancy between distances in the class and out of the class is smaller than one then the push term is active, otherwise it gets inactivated, this is how the margin is enforced. The choice of a unit margin, is merely a convention as choosing another value would only scale A. This problem is again expressed into a constrained problem (A p.s.d. is of course one of the constraints) with slack variables and solved with sub–gradient descent and projections on the constraint spaces.

In [Torresani 2007] an extension to LMNN is proposed, *Large Margin Component Analysis* (LMCA), in which dimension reduction is also carried out and the minimization of equation (5.32) is done on a rectangle matrix R of dimension $r \times d$ where d is the dimension of the space $\mathcal{X}$ and r is the maximum rank of the resulting matrix $A = R^T R$. This constrain on the rank of A constitutes the dimension reduction, and the optimization is done directly by gradient descent on the space of matrices. Using this method, kernelization of the method is possible and presented in the paper.

Since LMNN, large margin metric learning methods have gained a lot of attention. Recently [Shen 2009, Shen 2012], was proposed a take on large margin metric learning with the use of boosting like methods, thus alleviating the problems of scalability. Training triplets $\{(\mathbf{x}_r, \mathbf{x}_r^+, \mathbf{x}_r^-)\}_r$ are considered, such that $d(\mathbf{x}_r, \mathbf{x}_r^+) < d(\mathbf{x}_r, \mathbf{x}_r^-)$ for all r and they are associated with the margin:

$$\begin{aligned} \mu_r &= \left\|\mathbf{x}_r - \mathbf{x}_r^-\right\|_A^2 - \left\|\mathbf{x}_r - \mathbf{x}_r^-\right\|_A^2 \\ &= (\mathbf{x}_r - \mathbf{x}_r^-)^T A(\mathbf{x}_r - \mathbf{x}_r^-) - (\mathbf{x}_r - \mathbf{x}_r^+)^T A(\mathbf{x}_r - \mathbf{x}_r^+) \\ &= \operatorname{tr}\{B_r^T A\} = \langle B_r, A\rangle \end{aligned} \tag{5.33}$$

where $B_r = (\mathbf{x}_r - \mathbf{x}_r^-)(\mathbf{x}_r - \mathbf{x}_r^-)^T - (\mathbf{x}_r - \mathbf{x}_r^+)(\mathbf{x}_r - \mathbf{x}_r^+)^T$ and $\langle \cdot, \cdot \rangle$ denotes the standard inner product on the space of matrices. The margin μ_r quantifies how well the projection R (or, equivalently, the positive semi-definite matrix matrix A) separates the positive pair from the negative pair in training sample r. Then the following log-exponential cost is minimized:

$$\min_A \quad \log\left(\sum_r \exp\left(-\mu_r\right)\right) + \epsilon \operatorname{tr}\{A\} \tag{5.34}$$

$$\text{s.t.} \quad \begin{cases} \mu_r = \langle B_r, A\rangle \\ A \text{ p.s.d.} \end{cases}$$

ϵ is a small positive parameter used to avoid arbitrary scaling of A. The trace norm term involving it further promotes low-rank solutions, which adds a dimensionality reduction flavor to the approach. The major innovation of the method is to use an observation made in [Shen 2008], that any positive semi-definite matrix can be decomposed into a linear positive combination of trace-one rank one matrices. Use of this fact is made in a boosting approach in which a single row of the projection matrix R is added at each iteration. With A expressed as:

$$A = \sum_j \omega_j \boldsymbol{\xi}_j \boldsymbol{\xi}_j^T \tag{5.35}$$

with $\omega_j \geq 0$ and$\|\boldsymbol{\xi}_j\| = 1$. Now the minimization problem can be rewritten:

$$\min_{\omega, \boldsymbol{\xi}} \quad \log\left(\sum_r \exp\left(-\mu_r\right)\right) + \epsilon \sum_j \omega_j \tag{5.36}$$

$$\text{s.t.} \quad \begin{cases} \mu_r = \sum_j \omega_j \boldsymbol{\xi}_j^T B_r \boldsymbol{\xi}_j \\ \omega_j \geq 0 \\ \|\boldsymbol{\xi}_j\| = 1 \end{cases}$$

Minimization over ω and $\boldsymbol{\xi}$ is performed using a procedure similar to *Adaboost* classification that treats $\boldsymbol{\xi}_j^T B_r \boldsymbol{\xi}_j$ as a weak learner and increases the influence of training samples B_r violating the constraints the most.

Online Metric Learning Another strand of metric learning techniques developed alongside, the previously discussed methods, the online metric learning methods. Like all online methods, online metric learning methods learn their parameters and get better adjusted to the task as they are fed with samples. Formally, online metric learning methods start with a set of initial parameters and are given a new pair of samples $(\mathbf{x}_1, \mathbf{x}_1') \in \mathcal{X}^2$ and make a prediction on their similarity. At iteration 2 the label ℓ_1 corresponding to iteration 1 is given, along side a new pair $(\mathbf{x}_2, \mathbf{x}_2')$ to make prediction on. The actualization of the parameters will take into account the discrepancy between the last prediction and the ground truth given by the label, and then make a new prediction.

One of the first method to attract attention was *Pseudo-metric Online Learning Algorithm* (POLA) [Shalev-Shwartz 2004], in which a mahalanobis-like pseudo metric is learned, and at each iteration, the hinge loss over all the previously seen samples is minimized:

$$L_t(A, b) = \max\left\{0, \ell_t\left(\|\mathbf{x}_t - \mathbf{x}_t'\|_A^2 - b\right) + 1\right\} \tag{5.37}$$

At each iteration, two parameters are modified, the matrix A and the threshold parameter b. When the Mahalanobis distance between two samples goes over b they are considered dissimilar.
With each new sample pair, A and b are projected onto two sets, first the set of permissible solutions:

$$C_t = \{(A, b) \text{ s.t. } L_t(A, b) = 0\} \tag{5.38}$$

and then onto the set of constraints:

$$C_c = \{(A, b) \text{ s.t. } A \text{ p.s.d, } b \geq 1\} \tag{5.39}$$

projection on the space of p.s.d. matrices is achieved through eigenvector decomposition such as in [Xing 2002]. The problem of such a decomposition is that it is computationally expensive which is a drawback when considering online methods that should be designed for near real-time performances.

In [Davis 2007] an online version of ITML was also presented, this version doesn't require the eigenvector decomposition to be made, but still presents quite costful computations. This problem was recently addressed in [Jain 2008], where the update on the Mahalanobis matrix is completely made by a gradient descent step, the algorithm is referred to as *LogDet Exact Gradient Online* (LEGO). The loss function considered at each iteration is:

$$L_t(A) = \frac{1}{2}\left(\|\mathbf{x}_t - \mathbf{x}'_t\|^2_A - y_t\right)^2 \tag{5.40}$$

where y_t is the groundtruth value of the measurement on $\|\mathbf{x}_t - \mathbf{x}'_t\|^2_A$. Then following [Davis 2007], the update rule on A follows:

$$A_{t+1} = \underset{A \text{ p.s.d.}}{\operatorname{argmin}} \left\{D_{KL}\left(p(\mathbf{x_t}; A) \,\|p(\mathbf{x_t}; A_t)\right) + \lambda L_t(A)\right\} \tag{5.41}$$

Careful derivation of the previous cost leads to the gradient update on A, it is shown that with iterations A will keep p.s.d. and bounds on the error are given.

Recently, a new approach to online metric learning gained a lot of attention. The approach named *Online Algorithm for Scalable Image Similarity learning* (OASIS) [Chechik 2009, Chechik 2010] iterestingly steps back from the Mahalanobis distance learning approach and learns a simillarity S_W that is a simple bilinear form:

$$S_W(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T W \mathbf{x}_j \tag{5.42}$$

at each iterations, a triplet of samples in $\mathcal{X}$ is considered such that $S_W(\mathbf{x}_i, \mathbf{x}_i^+) > S_W(\mathbf{x}_i, \mathbf{x}_i^-)$, this is enforced by the margin equation:

$$S_W(\mathbf{x}_i, \mathbf{x}_i^+) > S_W(\mathbf{x}_i, \mathbf{x}_i^-) + 1 \quad \forall \mathbf{x}_i, \mathbf{x}_i^+, \mathbf{x}_i^- \in \mathcal{X} \tag{5.43}$$

The update equation is following the Passive-Aggressive algorithm:

$$W_i = \underset{W}{\operatorname{argmin}} \frac{1}{2} \left\| W - W^{i-1} \right\|_F^2 + C\xi \tag{5.44}$$

$$\text{s.t.} \begin{cases} l_w = \max(0, 1 - S_w(\mathbf{x}_i, \mathbf{x}_i^+) + S_w(\mathbf{x}_i, \mathbf{x}_i^-) \leq \xi \\ \xi \geq 0 \end{cases}$$

Where $\|\cdot\|_F$ is the Froebenius norm of the matrix (the sum of all the squared elements) this is solved by a simple gradient descent like update:

$$W_i = W_{i-1} + \lambda \frac{\partial l_W}{\partial W} \tag{5.45}$$

$$\lambda = \min \left\{ C, \frac{l_W}{\left\| \frac{\partial l_W}{\partial W} \right\|^2} \right\} \tag{5.46}$$

C is the trad-off at each iterations between how close the new matrix is to the previous one and the respect of the new constraint. The Whole algorithm is very fast and scalable. In the resolution, sparcity of the elements of $\mathcal{X}$ is taken into account to augment the computationla efficiency.

The similarity matrix W learned by OASIS is not guaranteed to be positive or symmetric. And this might be an advantage to some applications that are known to be non-symmetric, such as the demonstrated ranking of images by semantic relevance to a given image query. However, variants of the algorithm are proposed where the Mahalanobis distance is considered, then the approach resembles online LMNN, in this case, projection onto the space of p.s.d matrices is done. This is shown to reduce overfitting, however the performance of the algorithm is shown to decrease due to the fact that the Null space of the matrix is hard to determine numerically (when the projection on the space of p.s.d. matrices is done) when there is noise in the samples.

5.3 Multi-Modal Metric Learning

Multi-modal metric learing builds onto the unimodal cases that were described earlier in this chapter. Instead of dealing with samples from $\mathcal{X}$ and trying to find the best suited metric on $\mathcal{X}$ we are now faced with elements from $\mathcal{X} \subset \mathbb{R}^{d_1}$ and $\mathcal{Y} \subset \mathbb{R}^{d_2}$, with d_1

usually different from d_2, two different spaces with elements possibly completely different in nature and try to find a metric on the product space $\mathcal{X} \times \mathcal{Y}$. This metric will be learned on a training set representing the similarity on the spaces that we want to learn. This will usually present itself as a set of labeled pairs, with pairs given a label 1 if similar and -1 if dissimilar, or even a more general sense of similarity with triplets with a sample from $\mathcal{X}$ and two samples from $\mathcal{Y}$ with the requirement that one is semantically closer to the element in $\mathcal{X}$ than the other.

We can see that this problem is much harder than the usual metric learning problem, because here we don't have any natural metric on the product space $\mathcal{X} \times \mathcal{Y}$, which we could then modify as is the case with Mahalanobis type metrics in the unimodal case.

However we will see that some unimodal problems can be modified to account for multi-modality, or even that a wide range of unimodal algorithms can be adapted to the multi-modal case.

5.4 Cross-Modality Sililarity Sensitive Hashing

The first idea for metric learning in this chapter is to find a common embedding space $\mathcal{H} \subset \mathbb{R}^d$ for metric learning. This is depicted in figure 5.3, we want to learn two projection functions $f : \mathcal{X} \to \mathcal{H}$ and $g : \mathcal{Y} \to \mathcal{H}$, such that elements of $\mathcal{X}$ and $\mathcal{Y}$ labeled as similar end up close, in the L_1 measure sense,in the common embedding space $\mathcal{H}$, while dissimilar elements end up as far away as possible in the embedding. Then given two new samples from $\mathcal{X}$ and $\mathcal{Y}$, the distance is computed with the application of the projection functions followed by the distance computation in the embedding space.

Using the work of [Shakhnarovich 2005, Ren 2005] for unimodal metrics, we show here how we can extend it to the multimodal case. This work has been presented in [Bronstein 2010] and an application to 3D images and Gabor features embedding was presented in [Michel 2011].

5.4.1 Extension on Similarity sensitive Hashing

Let us consider $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^{d_1}$ and $\mathbf{y} \in \mathcal{X} \subseteq \mathbb{R}^{d_2}$ with $d_1 \neq d_2$. The M-dimensional *Hamming* embedding for each image space can be thought as binary vectors $\xi(\mathbf{x})$ and $\eta(\mathbf{y})$ with

$$\xi(\mathbf{x}) = \begin{pmatrix} \xi_1(\mathbf{x}) \\ \vdots \\ \xi_M(\mathbf{x}) \end{pmatrix} \quad \text{and} \quad \eta(\mathbf{y}) = \begin{pmatrix} \eta_1(\mathbf{y}) \\ \vdots \\ \eta_M(\mathbf{y}) \end{pmatrix} \tag{5.47}$$

we model $\xi_m(\mathbf{x})$ and $\eta_m(\mathbf{y})$ such that

$$\xi_m(\mathbf{x}) = \begin{cases} 0 & \text{if} \quad f_m(\mathbf{x}) \leq 0 \\ 1 & \text{otherwise} \end{cases} \qquad \text{and} \qquad \eta_m(\mathbf{y}) = \begin{cases} 0 & \text{if} \quad g_m(\mathbf{y}) \leq 0 \\ 1 & \text{otherwise} \end{cases} \tag{5.48}$$

where f_m and g_m are the components of the vector functions f and g, we assume here that f and g are linear projections and so we have:

$$f_m(\mathbf{x}) = p_m^T \mathbf{x} + a_m \quad \text{and} \quad g_m(\mathbf{y}) = q_m^T \mathbf{y} + b_m \tag{5.49}$$

The ℓ_1 distance on the *Hamming* space defines the *Hamming* distance denoted $\mathrm{d}_{\mathcal{H}}$:

$$\mathrm{d}_{\mathcal{H}}\left(\xi(\mathbf{x}), \eta(\mathbf{y})\right) = \left\|\xi(\mathbf{x}) - \eta(\mathbf{y})\right\|_{\ell_1} \tag{5.50}$$

$$\mathrm{d}_{\mathcal{H}}\left(\xi(\mathbf{x}), \eta(\mathbf{y})\right) = \sum_{m=1}^{M} \left|\xi_m(\mathbf{x}) - \eta_m(\mathbf{y})\right| \tag{5.51}$$

Now for each dimension m, we can define a weak binary classifier:

$$\begin{aligned} c_m(\mathbf{x}, \mathbf{y}) &= \begin{cases} 1 & \text{if} \quad \xi_m(\mathbf{x}) = \eta_m(\mathbf{y}) \\ -1 & \text{otherwise} \end{cases} \\ &= \left(2\xi_m(\mathbf{x}) - 1\right)\left(2\eta_m(\mathbf{y}) - 1\right) \\ &= \operatorname{sgn}\left(f_m(\mathbf{x})\right) \operatorname{sgn}\left(g_m(\mathbf{y})\right) \\ &= \operatorname{sgn}\left(p_m^T \mathbf{x} + a_m\right) \operatorname{sgn}\left(q_m^T \mathbf{y} + b_m\right) \end{aligned} \tag{5.52}$$

We can also rewrite $\mathrm{d}_{\mathcal{H}}$:

$$\mathrm{d}_{\mathcal{H}}\left(\xi(\mathbf{x}), \eta(\mathbf{y})\right) = \frac{M}{2} - \frac{1}{2} \sum_{m=1}^{M} c_m(\mathbf{x}, \mathbf{y}) \tag{5.53}$$

Now if we define the Similarity Classifier C:

$$C(\mathbf{x}, \mathbf{y}) = \operatorname{sgn}\left(\sum_{m=1}^{M} c_m(\mathbf{x}, \mathbf{y})\right) \tag{5.54}$$

then $\mathrm{d}_{\mathcal{H}}$ is small for $C(\mathbf{x}, \mathbf{y}) = +1$ and large for $C(\mathbf{x}, \mathbf{y}) = -1$ with high probability.

Resolution with AdaBoost We find the parameters of the projection functions, using Adaboost [Freund 1995] to solve the classification problem. Let us assume we are working with N sample pairs $\{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{X} \times \mathcal{Y}\}_{i\in\{1,\dots,N\}}$ associated with N classification labels $(\ell(i))_{i\in\{1,\dots,N\}}$

Following the Adaboost update rule, we have the weighted correlation of labels with prediction:

$$\begin{aligned} r_m &= \sum_{i=1}^{N} W_m(i)\ell(i)c_m(\mathbf{x}_i, \mathbf{y}_i) \\ &= \sum_{i=1}^{N} W_m(i)\ell(i)\text{sgn}\left(p_m^T\mathbf{x} + a_m\right)\text{sgn}\left(q_m^T\mathbf{y} + b_m\right) \end{aligned} \tag{5.55}$$

A reasonable objective of the weak learner at iteration m is to maximize r_m Our boosted cross-modality similarity learning algorithm can be summarized as follows:

Algorithm 5.1 Boosted cross-modal similarity-sensitive embedding

Require: K pairs (x_k, y_k) labeled by $s_k = s(x_k, y_k)$

{Initialize weights} $w_{1k} = 1/K$

for $i = 1$ **to** n **do**

Select ξ_i and η_i such that c_i in (5.52) maximizes:

$$r_i = \sum_{k=1}^{K} W_i(k)\ell(k)c_i(x_k, y_k). \tag{5.56}$$

Set $\alpha_i = \frac{1}{2}\log(1 + r_i) - \frac{1}{2}\log(1 - r_i)$.

Update weights according to

$$w_{i+1,k} = W_i(k)e^{-\alpha_i\ell(k)c_i(x_k,y_k)} \tag{5.57}$$

and normalize by sum.

end for

return maps $\xi_i : X \to \{0, 1\}$ and $\eta_i : Y \to \{0, 1\}$, and scalars α_i, $i = 1, \dots, n$.

The algorithm follows very much the standard AdaBoost procedure. It consists of two steps, where first the maximization of the weighted correlation r_i of labels with the outputs of the weak classifier is addressed. This step is followed by the selection of α_i that minimizes the exponential loss [Freund 1995]. In case the unweighted version of the Hamming metric is used, s skipped, fixing $\alpha_i = 1$.

Projection selection Details of projection selection specific to our cross-modality similarity learning problem are concentrated in the first step.

Substituting the affine projections f_i and g_i into (5.56), we obtain

$$r_i = \sum_{k=1}^{K} W_i(k)\ell(k)\text{sign}\,(p_i^{\mathrm{T}} x_k + a_i)\text{sign}\,(q_i^{\mathrm{T}} y_k + b_i). \tag{5.58}$$

Maximizing r_i with respect to the projection parameters is difficult because of the sign function. However, this maximizer is closely related to the maximizer of a simpler function,

$$\hat{r}_i \;\; = \;\; \sum_{k=1}^{K} v_k (p_i^{\mathrm{T}} \overline{x}_k)(q_i^{\mathrm{T}} \overline{y}_k), \tag{5.59}$$

where $\overline{x}_k$ and $\overline{y}_k$ are x_k and y_k centered by their weighted means, and $v_k = W_i(k)s_k$. Rewriting the above yields

$$\hat{r}_i \;\; = \;\; p_i^{\mathrm{T}} \left(\sum_{k=1}^{K} v_k \overline{x}_k \overline{y}_k^{\mathrm{T}} \right) q_i = p_i^{\mathrm{T}} C q_i, \tag{5.60}$$

where C can be thought of as the difference between weighted covariance matrices of positive and negative pairs of the training data points.

Unit projection directions p_i and q_i maximizing $\hat{r}_i$ correspond, respectively, to the largest left and right singular vectors of C. In practice, since the minimizers of $\hat{r}_i$ and r_i are not identical, we project x_k and y_k onto the subspaces spanned by M largest left and right singular vectors. Selecting $M \ll m, m'$ allows to greatly reduce the search space complexity. In our experiments, M was empirically set to 5; further increase of M did not bring significant improvement.

In order to find the best projection directions p_i and q_i in the two reduced M-dimensional search spaces, the following concept was used: N pairs of M-dimensional random vectors are generated. Each such pair forms a candidate for the pair of projection directions p_i and q_i; for each candidate, we project the training data points obtaining two sets of scalars $x'_k = p_i^{\mathrm{T}} x_k$ and $y'_k = q_i^{\mathrm{T}} y_k$. Next, we search for the scalar parameters a_i and b_i maximizing r_i. For that purpose, for every pair of scalars (a, b), we define the cumulative sum

$$S(a, b) = \sum_{k=1}^{K} \mathbf{1}(x'_k + a \leq 0)\mathbf{1}(y'_k + b \leq 0)v_k, \tag{5.61}$$

where $\mathbf{1}$ denotes an indicator function. In this notation, r_i can be expressed as $r_i(a, b) = 4S(a, b) + S(-\infty, -\infty) - 2S(a, -\infty) - 2S(-\infty, b)$. In order to find (a, b) maximizing r_i, we quantize the space of candidate pairs (a, b) on a grid of $B \times B$ bins and evaluate $S(a, b)$ and, hence, $r_i(a, b)$ in each bin.

5.4.2 Similarity Map Experiment

For our experiments, we used the MR brain images of ten patients. For each patient, perfectly co-registered T1-, T2-weighted and Proton-Density (PD) images were available. Four pairs of images were used for training; the rest was used for testing. The training dataset was designed using the groundtruth correspondence between the multi-modal images: feature vectors at corresponding location in two different modalities were considered similar, while two feature vectors extracted at a location distant 14 to 16 pixels from the groundtruth correspondence location were considered dissimilar. For the training set, we randomly picked features vectors in the four image pairs, with $|\mathcal{P}| = 20 \times 10^3$ positive and $|\mathcal{N}| = 200 \times 10^3$ negative pairs.

To visually assess the validity of the learned measure, we plot the learned metric from a point in the image in one modality to all the points in the image in second modality in Figure 5.4 (since the data are 3D, two 2D slices are shown). It is interesting to observe that for some very distinctive points in the image, the distance is close to 0 in a very limited area around the point position (first row of Figure 5.4), while in less distinctive image areas, the distance profile is more shallow around the point. We can note that the size of the valley around the point of interest in the latter case is around 15 voxels in radius, which is consistent with the training set creation.

5.5 Maximum-Margin Cross-Modal Metric Learning

Cross-Modality Similarity Sensitive Hashing was one of the first method to bring metric learning to the multi-modal case. Its formulation allows for very fast computation of the distance thanks to the hamming space embedding which allows to make comparison of feature vectors through their representation in the Hamming space which thus consists in the comparison of binary vectors.

However this practicality comes with a cost, namely using sign functions in the expression of the distance prevents us from having a differentiable measure. Unfortunately as we have seen in Chapter 2, many registration methods need a differentiable cost to perform the optimization on the transformation. We thus would like to have access to a learned distance metric that is based on a L_2 distance.

A second problem of the previous method is that it is based on the well known algorithm AdaBoost. Even though the usage of AdaBoost makes the computation of the projection functions very efficient, it is well known that AdaBoost is very sensitive to label noise as shown in [Dietterich 2000]. Lately, the usage of Maximum Margin methods in learning has become very popular, and metric learning is no exception as we have seen in section 5.2.2. Maximum Margin methods aim at maximizing the distance (the margin)

between the decision boundary and the training samples. This is usually done with the usage of slack variables that allow some samples in the training set to fall within margin and thus renders the problem feasible. Maximizing the distance between the samples and the decision boundary allows for a larger security on the testing set, and prevents the algorithm from the sensitivity to the label noise. Like in support vector classification, this improves the generalization properties of the metric and constitutes a provably powerful mechanism against overfitting.

In this section we are going to explore a new paradigm for cross-modal metric learning. Instead of learning at the same time the new embedding and the metric in this space, we take a two-step approach in which we first embed both feature spaces in the same space, then learn the embedding in this space. This approach has an obvious advantage on the previous one, since we learn the embedding on a common space the probelm can be considered as a unimodal problem, and thus can be solved with off-the-shelf metric learning algorithms.

Let us explore this solution in details.

5.5.1 Learning a common space embedding

Extending the idea of metric learning to the multimodal setting, let $\mathcal{X} \subseteq \mathbb{R}^m$ and $\mathcal{Y} \subseteq \mathbb{R}^n$ denote two different representation spaces, and let $\{(\mathbf{x}_r, \mathbf{y}_r^+, \mathbf{y}_r^-)\}_r$ denote the training triplets with $(\mathbf{x}_r, \mathbf{y}_r^+)$ being positive pairs and $(\mathbf{x}_r, \mathbf{y}_r^-)$ being negative ones. A possible approach would be to look for two embeddings $f : \mathcal{X} \longrightarrow \mathcal{H}$ and $g : \mathcal{Y} \longrightarrow \mathcal{H}$ of the input data into a common representation space such that $d_{f,g} = d_{\mathcal{H}} \circ (f \times g)$ is again small for negative pairs and large for positive ones. This problem is arguably harder to solve than the unimodal problem, and the methods discussed before are not readily amenable to this framework. As an alternative, we propose to build a translation function $t : \mathcal{Y} \longrightarrow \mathcal{X}$ simultaneously with the embedding $f : \mathcal{X} \longrightarrow \mathcal{H}$. The resulting distance $d_{f,t} = d_{\mathcal{H}} \circ (f \times (f \circ t))$ fits well into the unimodal setting. In what follows, we present a multimodal maximum margin metric learning procedure based on this idea. To the best of our knowledge, this is the first time such an extension is considered.

As before, we assume f to be given by an embedding matrix $\mathbf{P}$. We furthermore assume t to be given by another matrix $\mathbf{T}$ such that

$$d_{f,t}(\mathbf{x}, \mathbf{y}) = \|\mathbf{P}\mathbf{x} - \mathbf{P}\mathbf{T}\mathbf{y}\|_2^2 \tag{5.62}$$

Considering the $\mathbf{T}\mathbf{y}$'s as samples in $\mathcal{X}$, the multimodal metric learning problem involving $d_{f,t}$ can be viewed as a conventional unimodal metric learning problem coupled with optimization over $\mathbf{T}$. While such a coupling can be done in various metric learning problems,

here we adopt the LMCA formulation due to its simplicity. We solve

$$\begin{aligned} \min_{\mathbf{P},\mathbf{T}} \quad & \textstyle\sum_r \|\mathbf{P}\mathbf{x}_r - \mathbf{P}\mathbf{T}\mathbf{y}_r^+\|_2^2 + c\sum_r h\left(1-\mu_r\right) \\ \text{s.t.} \quad & \mu_r = \|\mathbf{P}\mathbf{x}_r - \mathbf{P}\mathbf{T}\mathbf{y}_r^-\|_2^2 - \|\mathbf{P}\mathbf{x}_r - \mathbf{P}\mathbf{T}\mathbf{y}_r^+\|_2^2, \end{aligned} \tag{5.63}$$

by alternatingly fixing one of the matrices and solving for the other.

The process is initialized by setting $\mathbf{P}$ and $\mathbf{T}$ to be orthogonal projection matrices. A few gradient descent iterations are performed to obtain $\mathbf{T}$. We use an early stopping strategy in order not to fall far away from the minimizer for $\mathbf{P}$. Next, $\mathbf{T}$ is fixed and $\mathbf{P}$ is computed using unimodal maximum margin metric learning in which $\mathbf{x}_r^+$ and $\mathbf{x}_r^-$ are replaced with $\mathbf{T}\mathbf{y}_r^+$ and $\mathbf{T}\mathbf{y}_r^-$, respectively. At each iteration, the equal error rate (EER, defined as the rate at which false positive rate equals false negative rate) is evaluated on a validation set. The process is stopped when the latter ceases to decrease significantly or starts increasing as a consequence of overfitting. See Algorithm 5.2 for details.

Algorithm 5.2 Alternating minimization for multimodal maximum margin metric learning

Require: Training triplets $\{(\mathbf{x}_r, \mathbf{y}_r^+, \mathbf{y}_r^-) \in \mathcal{X} \times \mathcal{Y} \times \mathcal{Y}\}$; target embedding dimension p

{Initialize matrices to orthogonal projections} $\mathbf{T} = \mathbf{I}_{n,m}$, $\mathbf{P} = \mathbf{I}_{p,n}$.

while change in EER $> \epsilon$ **do**

Find $\mathbf{T}$ by gradient descent with early stopping:

$$\mathbf{T} \leftarrow \mathbf{T} - \lambda \frac{\partial C}{\partial \mathbf{T}} \tag{5.64}$$

where C is the cost function in (5.63) and λ is a step size found by line search.

Find $\mathbf{P}$ by unimodal maximum margin metric learning on $\{(\mathbf{x}_r, \mathbf{T}\mathbf{y}_r^+, \mathbf{T}\mathbf{y}_r^-)\}$.

end while

return A $p \times n$ matrix $\mathbf{P}$, and an $m \times n$ matrix $\mathbf{T}$

The method is relatively easy to implement, and allows to incorporate off-the-shelf unimodal metric learning algorithms. Depending on the unimodal method, this algorithm can be very fast.

In the rest of this chapter, for the cross-modal maximum margin techniques, we used two different margin maximization algorithms, they have been presented in section 5.2.2, and are the algorithms LMCA by [Torresani 2007] and Boosted Max Margin by [Shen 2012].

With a sample size of $N = 200,000$, dimensions m and n reaching 120 and $p = 64$, learning time did not exceed 30 minutes on an Intel Xeon W3530.

5.5.2 Training Dataset Creation

Though in theory the learning of the metric can be done directly on the image pixels [Lee 2009], image pixels are not very well suited for medical image analysis. Medical images often present high level of noise ; furthermore, in MR images the luminosity is not normalized, meaning that the range of gray levels may vary between acquisitions. The projection of the images into a space that is robust to those artifacts allows for a better learning of the embedding, and a better reproducibility of the results. We opted for the Invariant Gabor feature space that was presented in section 3.2.

In the same way as for Cross-Modality similarity sensitive hashing, the training dataset was designed using the groundtruth correspondence between the multi-modal images: feature vectors at corresponding location in two different modalities were considered similar, while two feature vectors extracted at a location distant (14 to 16 pixels in our case) from the groundtruth correspondence location were considered dissimilar. For the training set, we randomly picked features vectors in four image pairs, with a total of 200×10^3 samples.

To visually assess the validity of the learned distance, we plot the learned metric from a point in an image in one modality to all the points in an image in another modality in figure 5.5 (3D are depicted as 2D slices), in this figure the case of the distance between T1-MRI and T2-MRI is considered, other datasets include distance between T1-MRI and PD-MRI and distance between PET scan and CT scan and results will be presented in the next section. We observe that for almost all the points in the image, the distance is close to 0 in a very limited area around the point position.

The comparison side by side with the similarity map in figure 5.4 clearly shows that this similarity yields better and more located minima, which in turn will help the registration minimization.

Convergence of the Alternating Minimization Method Since the multimodal metric learing objective function (5.63) is non-convex in $\mathbf{P}$ and $\mathbf{T}$ simultaneously, we empirically tested the convergence of the proposed method with different unimodal metric learning problems used in the second step. The alternating minimization thus yields a steady decrease in the EER, until the minimizer of both objectives are reached. When this happens, an oscillatory behavior is exhibited. We discovered that while the procedure was derived from the LMCA algorithm, other margin maximization approaches also work very well, for instance the Boosted max margin metric learning detailed in section 5.2.2, yields even better results in terms of EER and later of registration than LMCA for the considered datasets.

We present here the evolution of the equal error rate for both methods, in the case of two of our datasets in figures 5.6 and 5.7

5.6 Results

For registration, the algorithm by [Glocker 2008] presented in section 2.3.2 has been used. The discrete Markov Random Fields approach allows us to make use of non-differentiable similarity measures such as our cross-modal similarity sensitive hashing. Optimization strategies such as FastPD[Komodakis 2007, Komodakis 2008] allow to find a good approximation to the global minimum of the labeling problem regardless of the similarity cost. These recent advances in optimization have made this work possible.

Two experiments were performed on two sets of images. In the first experiment, MRI brain scans of ten patients containing perfectly co-registered T1-weighted, T2-weighted, and Proton Density (PD) images were used for training and testing. In the second experiment, a second set of images composed of whole body scans of four patients acquired with synchronous CT and PET scans was used for training. All PET data were corrected for attenuation. In this paper, we focused on the chest part of the body scans.

5.6.1 Multi-Modal MRI image data set

In the case of multimodal MRI registration, two pairs of modalities were considered, T2 and T1 as well as T1 and PD. Given that we performed the training on four patients images, the registration tests were carried out on the remaining six. One of the six images was taken as the target image to which all other five images were registered. In order to remove any rigid transformation bias, an affine alignment was first performed. Non-rigid registration was performed between each of the five T2 images and the T1 target image; T1 to PD registration was evaluated in the same way. The same validation protocol was used as the one used in section 4.5.1. Manual segmentations of the ventricles were used to validate the alignment.It is to be noted that at the time of writing there is no publicly available data-base of multimodal images that are co-registered that we could use for training. The deformation field obtained from the alignment was used to warp a segmentation of the ventricle of the moving image. The deformed segmentation was compared to the ventricle segmentation of the target using the DICE coefficient measuring the proportion of overlap between two segmentations. We don't present here (as is often the case) the bare dice coefficient but its increase before and after registration as we believe that it is a fairer way to compare between registration algorithms (see section 4.5.1). Since the registration problem is fundamentally an ill-posed problem, a regularization term is usually introduced to penalize for irregular transformations and make the problem solvable. However, this smoothing term common to most registration algorithms prevents from having any reliable measure on the validity of the registration, since the setting of this term changes radically the quality of fit. In this paper, we quantify the smoothness of the warp

using the harmonic energy defined in [Yeo 2009] as the squared Frobenius norm of the Jacobian of the displacement field averaged over all voxels. The lower is the harmonic energy, the more rigid and smooth is the transformation. Deformations should be compared for the same values of harmonic energy.

Figure 5.8 and 5.9 depict the increase in the DICE coefficient as the result of the registration as function of the harmonic energy.

We compare our method to the most commonly used metrics in multi-modal medical image alignment in the case of T1-T2 and T1-PD alignment, and for reference, provide the results obtained in the unimodal case (T1-T1) with a learned unimodal metric using boosted maximum margin metric learning (Unimodal BMM)and also with the correlation ratio (Unimodal CR). Each curve represents a different method and consists of 20 point, factored in with 5 patients (this yields 100 experiments for one single curve). The solid line represents the average curve, and the whiskers around the line represent the maxima and minima for the method at the specific harmonic energy.

We can see on these datasets that our methods outperforms all commonly used similarity criterion even in the ideal unimodal case (figure 5.8), this is to be expected since, the amount of information conveyed by the feature vectors is much larger than the pixel intensity information used by these metrics.

We can also see that our second method adapting state of the art unimodal Max Margin metric learning algortihms to the multimodal case outperforms the first algorithms, the cross modality similarity sensitive hasing. We could already have a sens of this result by inspecting the similarity maps presented in figure 5.4 and 5.5.

We also provide in figure 5.10 a visual assesment of the results in the case of T1 to T2 registration with Cross-modality similarity sensitive hashing.

5.6.2 PET-CT image data set

For this experiments, we used a set of four images composed of whole body scans of four patients acquired with synchronous CT and PET. Following the leave-one out evaluation process, for each evaluation, we used three images for training and one image for registration testing. All four images were alternatively used for testing. In this paper, we focused on the chest part of the body scans. All PET data was corrected for attenuation. Due to the fact that PET data intensity distribution depends on the time after the injection of the tracer and the very high level of noise in the image, learning on raw PET data is very challenging. To have a more uniform training set, we used the Midway equalization algorithm described in [Delon 2004], which uses the average cumulative histogram of the training images as an image equalization tool.

In order to evaluate the registration, we artificially deformed the images using a grid

of 27 points randomly moved in a vicinity of 10 pixels, the interpolation was recovered using Thin Plate Splines (TPS) [Bookstein 1989]. Registration was performed between an undeformed CT image and a TPS deformed PET image. Two measures of the registration error were considered. First, we show the mean of the absolute intensity error between the undeformed image and the image recovered through registration. Second, we warped randomly distributed points in the image with the TPS deformation, then applied the recovered transformation to these points, the second measure of the error is the mean distance between the undeformed points and the recovered points.

Figure 5.11 depicts the two registration error measures as function of the harmonic energy. Each curve represents a different method and consists of 15 points, averaged over the 4 patients (this yields 60 experiments for one single curve). Comparison is made with Normalized Mutual Information which is commonly used in the multi-modal alignment of PET and CT.

Figure 5.12 shows the fused images before and after registration, in an alignment made using multi modal boosted maximum margin (CMBMM).

5.7 Conclusion

In this Chapter, we presented two very novel approaches for metric learning towards multi-modal image fusion. The first approach, Cross Modality Similarity Sensitive Hashing is a generalization of similarity-sensitive hashing to multi-modal data. To the best of our knowledge, this is the first attempt to approach the challenging problem of cross-modality similarity learning as an embedding problem. We showed that using cross-modality similarity learning allows to efficiently perform alignment of medical images acquired with different modalities. While in retrieval applications the Hamming embedding is advantageous due to its low computational and storage complexity and easy integration into existing database managements systems, the Hamming metric is discrete-valued and involves a non-differentiable non-linearity.

This is why we developed our second approach that extends large margin component analysis to deal with the multi-modal case and adopts boosted max margin concepts. The resulting metric is continuous, differentiable and is computationally efficient. Furthermore, it seems to inherit strong discrimination power and outperforms other learning-based methods. Further improvement of the method such as the use of convex criteria to determine the embeddings and the metric could also add theoretical stability in the process. The use of non-linear data assumptions can be easily encoded in the process through kernel-based methods and is currently under investigation. Last but not least, the use of context through local interactions between observations could make the process more robust and help differentiate between cases where similar features are observable in different

anatomical structures.

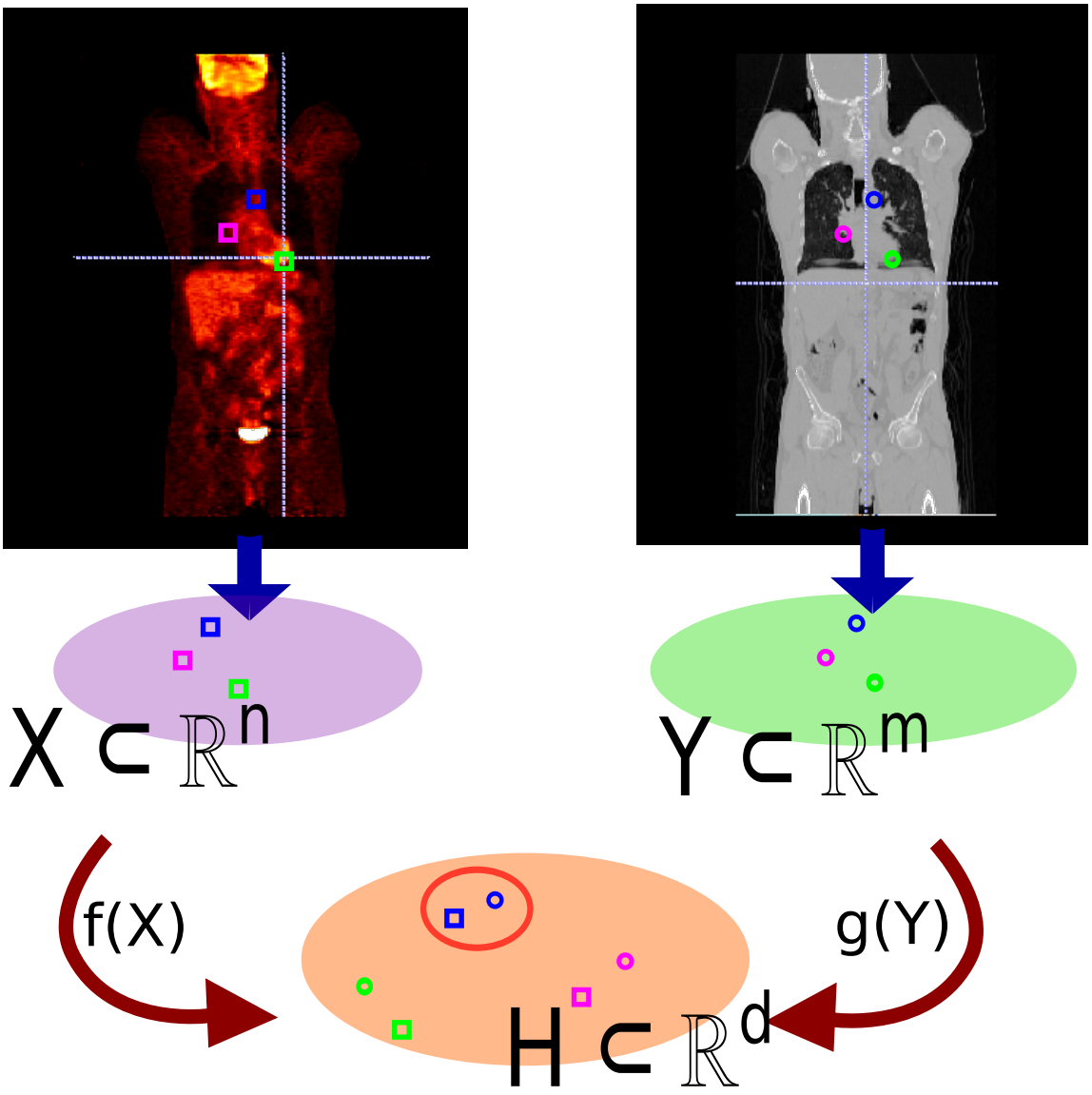


Figure 5.3: Creation of a common embedding space: features are extracted from a set of two perfectly aligned images, this feautres are each embeded in different spaces X and Y. Using similarity sensitive hashing we aim to lear two projection functions f and g that will map the elements from X and Y respectively into a common space H in which elements that were labeled as similar in the training set are embedded close to each other (red circle) while dissimlar pairs are embeded as far away as possible. The dimension of the embedding space H is a parameter of the algorithm

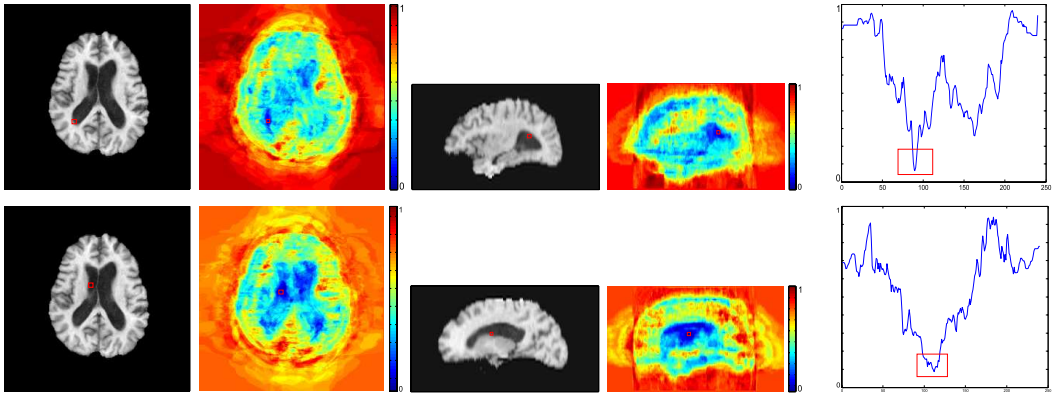


Figure 5.4: Distance map: plot of the learned distance taken between the feature vector extracted in the red square position on the left on the T1-MRI and all of the feature vectors extracted on the corresponding co-registered T2-MRI image. Far right is a profile extracted on the same line as the reference position. Bottom row presents the less distinctive case, notice the 15 voxels neighborhood around the extraction position.

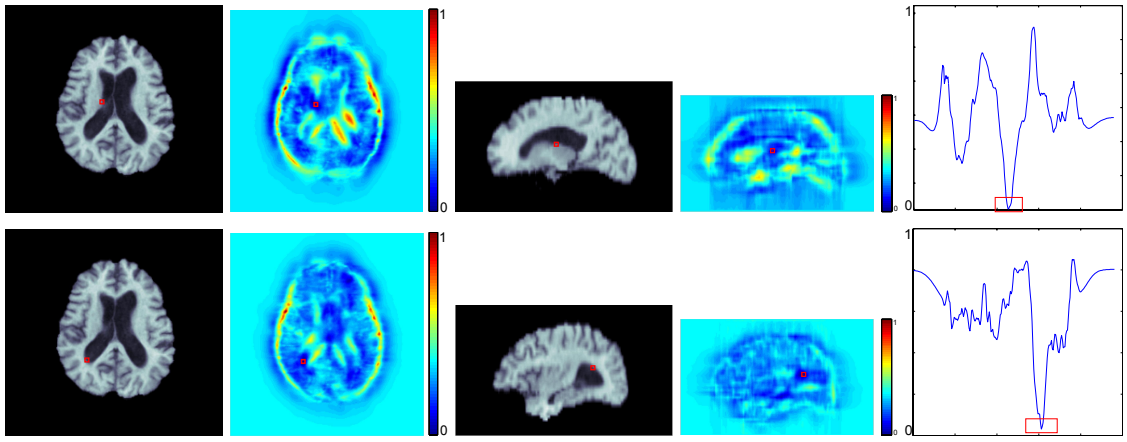


Figure 5.5: Distance map: plot of the learned distance taken between the feature vector extracted in the red square position on the left on the T1-MRI and all of the feature vectors extracted on the corresponding co-registered T2-MRI image. Far right is a profile extracted on the same line as the reference position.

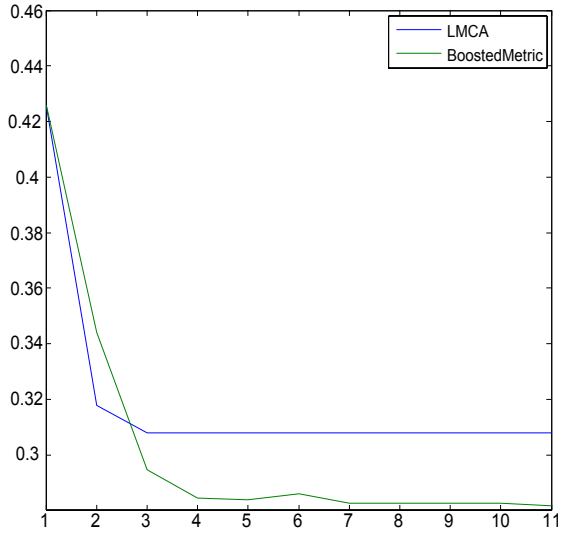


Figure 5.6: Evolution of the Equal Error Rate (EER) with the iterations of the alternate minimization for the PET CT dataset

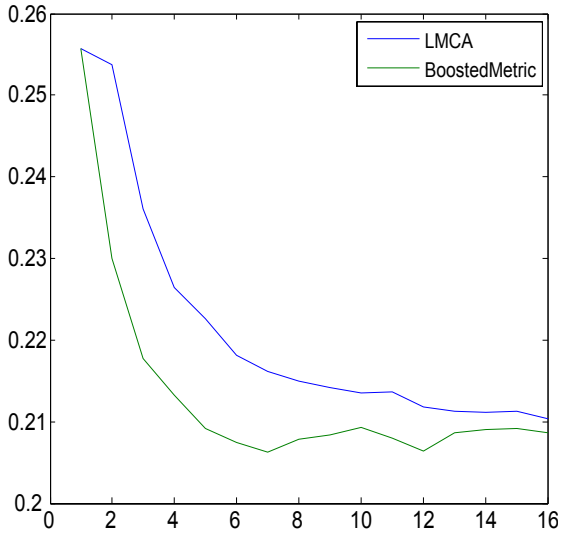


Figure 5.7: Evolution of the Equal Error Rate (EER) with the iterations of the alternate minimization for the T1-MRI T2-MRI dataset

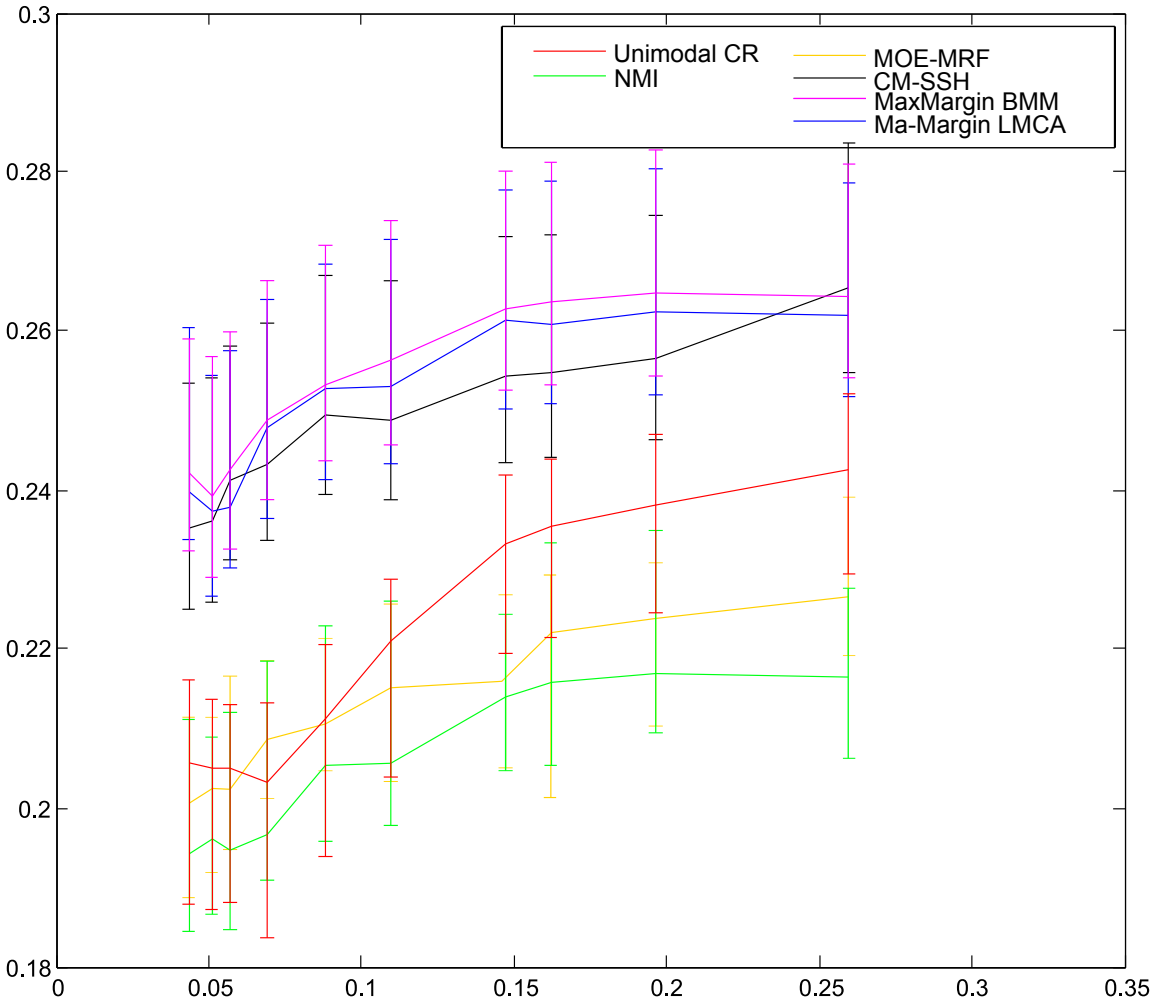


Figure 5.8: Evolution of the difference in dice coefficient as a function of the harmonic energy, each single curve factors in 100 experiments, the solid line curve represents the average dice coefficient increase while the whiskers ends represent the minimum and maximum increase in the dice coefficient. Here is presented the case of T1 to PD MRI registration, presented are the results with Normalized mutual information (NMI), Unimodal Correlation Ration (Unimodal CR), Mixture of experts with MRF (MOE-MRF), our Cross-modality similarity sensitive hashing (CM-SSH), our two adapted measures Corss Modal Max margin Boosted Max MArgin (Max-Margin BMM), and with LMCA (Max-Margin LMCA)

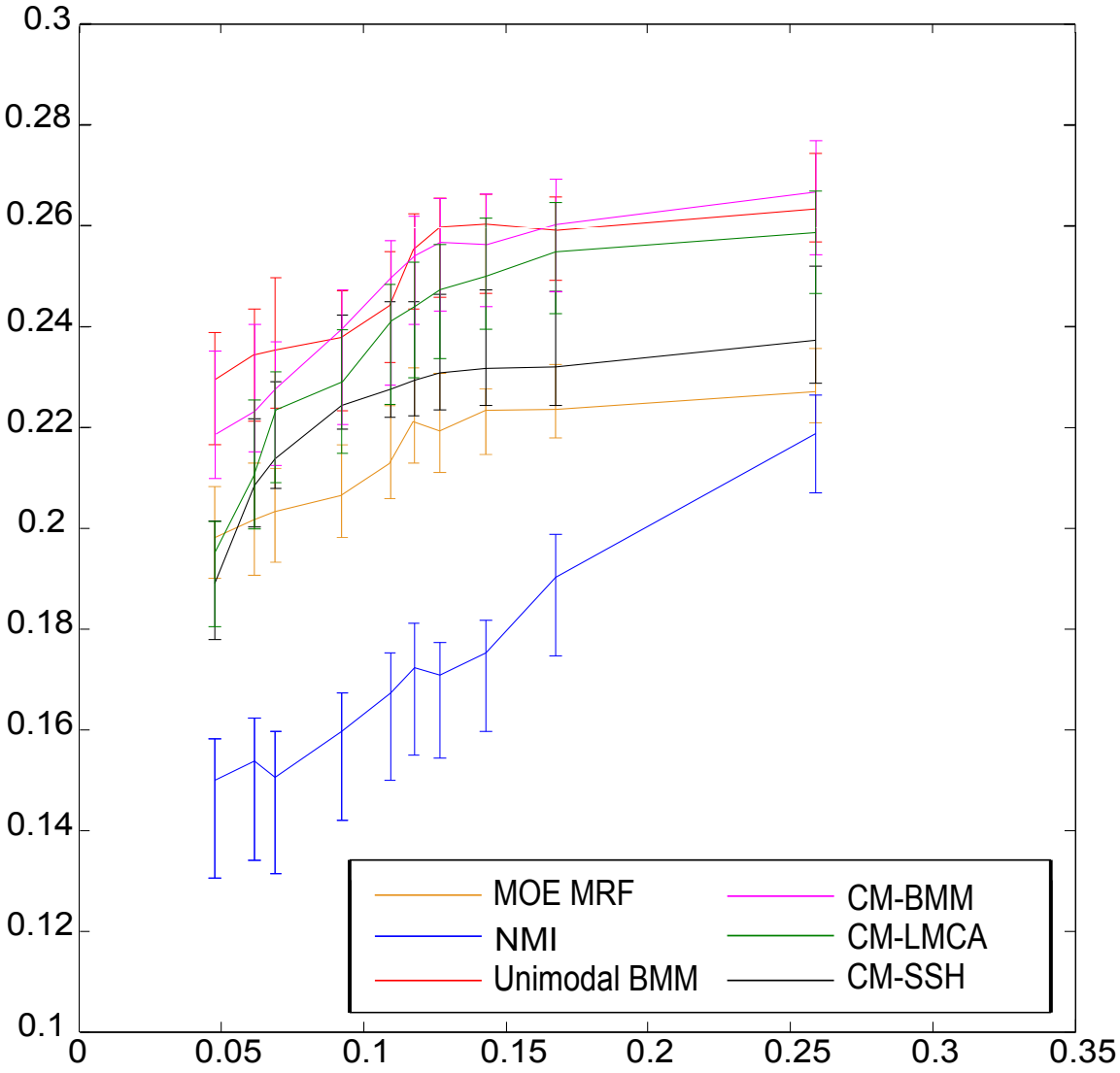


Figure 5.9: Evolution of the difference in dice coefficient as a function of the harmonic energy, each single curve factors in 100 experiments, the solid line curve represents the average dice coefficient increase while the whiskers ends represent the minimum and maximum increase in the dice coefficient. Here is presented the case of T1 to T2 MRI registration, presented are the results with Normalized mutual information (NMI), Unimodal Bosted Max Margin (Unimodal BMM) which represents the Metric Learning ideal case, Mixture of experts with MRF (MOE-MRF), our Cross-modality similarity sensitive hashing (CM-SSH), our two adapted measures Corss Modal Max margin Boosted Max Margin (CM BMM), and with LMCA (CM-LMCA)

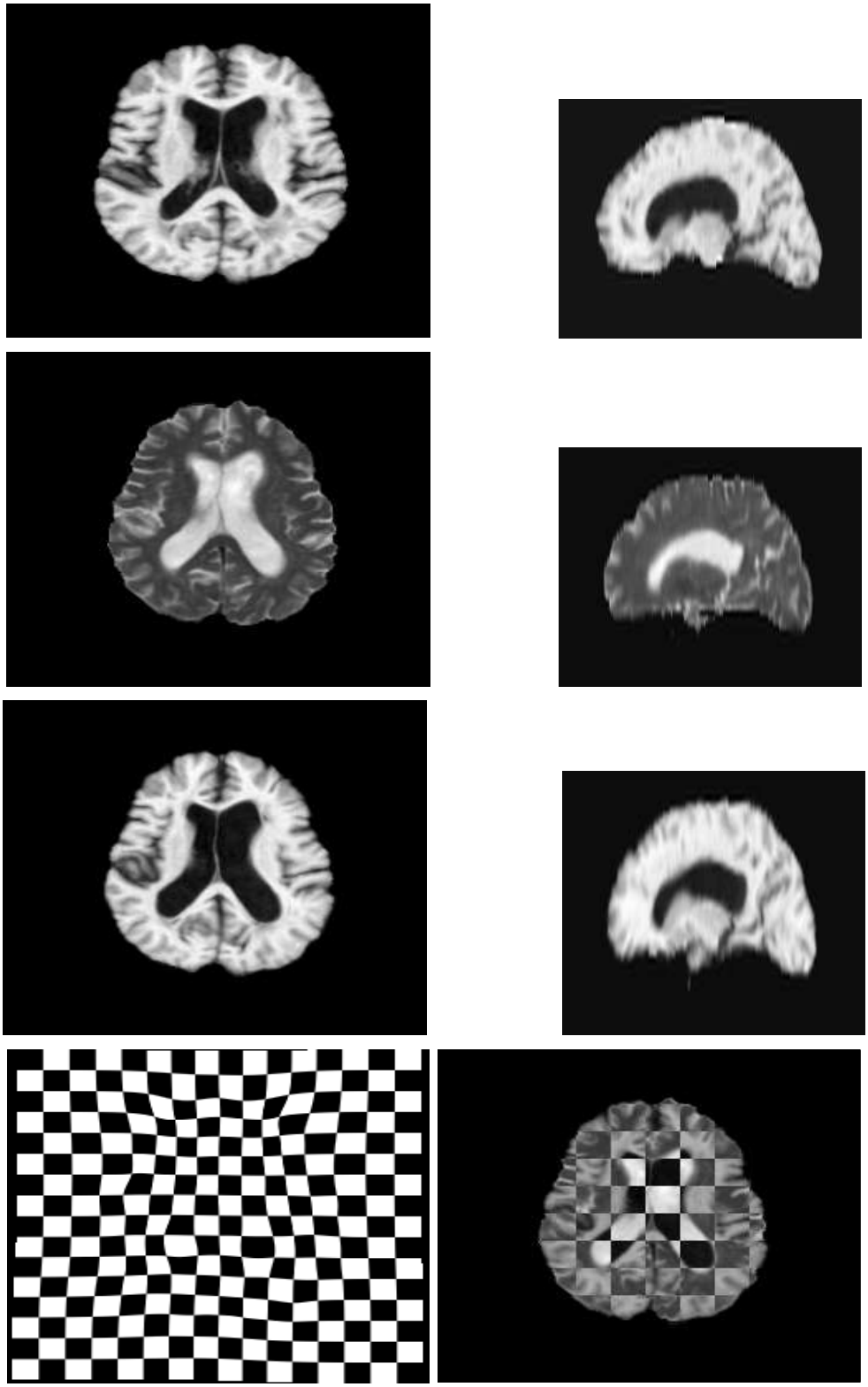


Figure 5.10: Sample of the registration results obtained for T1-T2 registration with Cross modality similarity sensitive hashing. Top row: Source Image T1-MRI image. Second Row: target T2-MRI image. Third Row: deformed image after multi-modal deformable registration. Bottom Row: left, deformation field of the registration, right, checker-board image between the target and the deformed source.

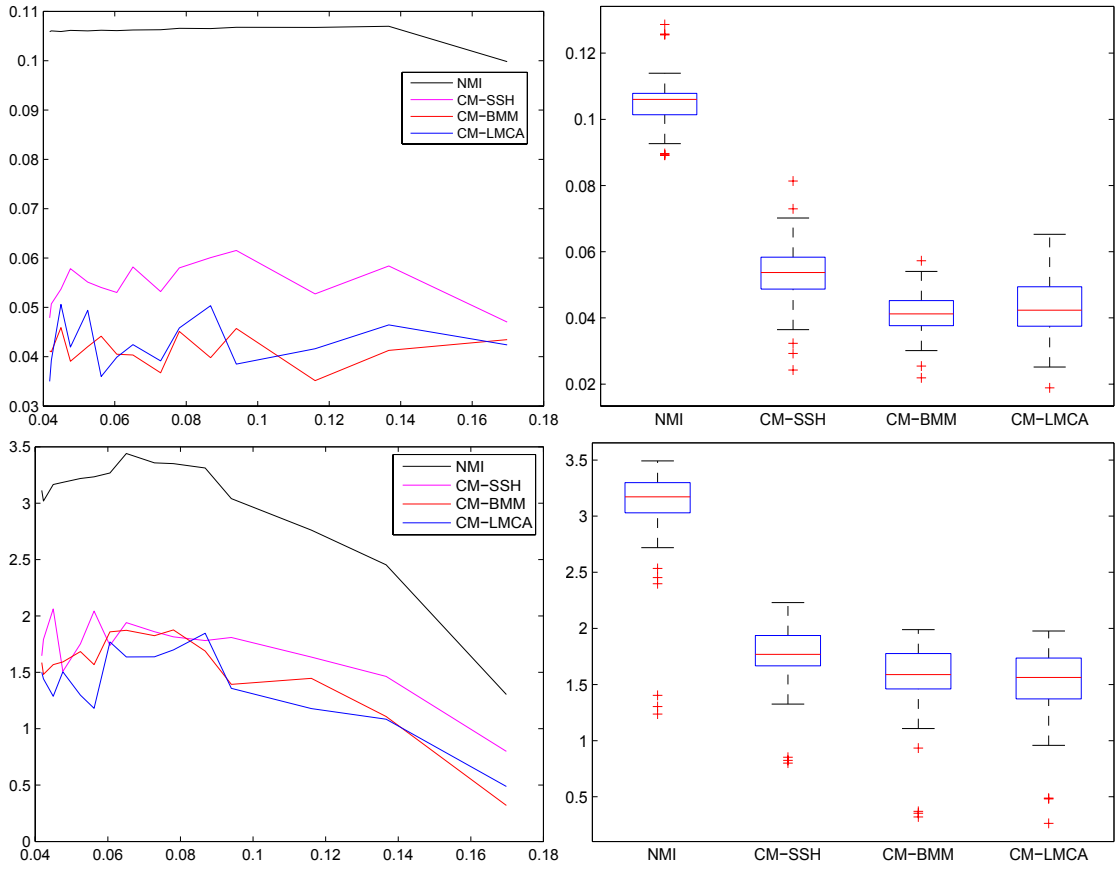


Figure 5.11: Error measure as a function of the Harmonic Energy, our methods are denoted as CM-BMM and CM-LMCA. Top row: mean absolute difference of the images. Bottom row: mean distance between undeformed points and points after transformation recovery

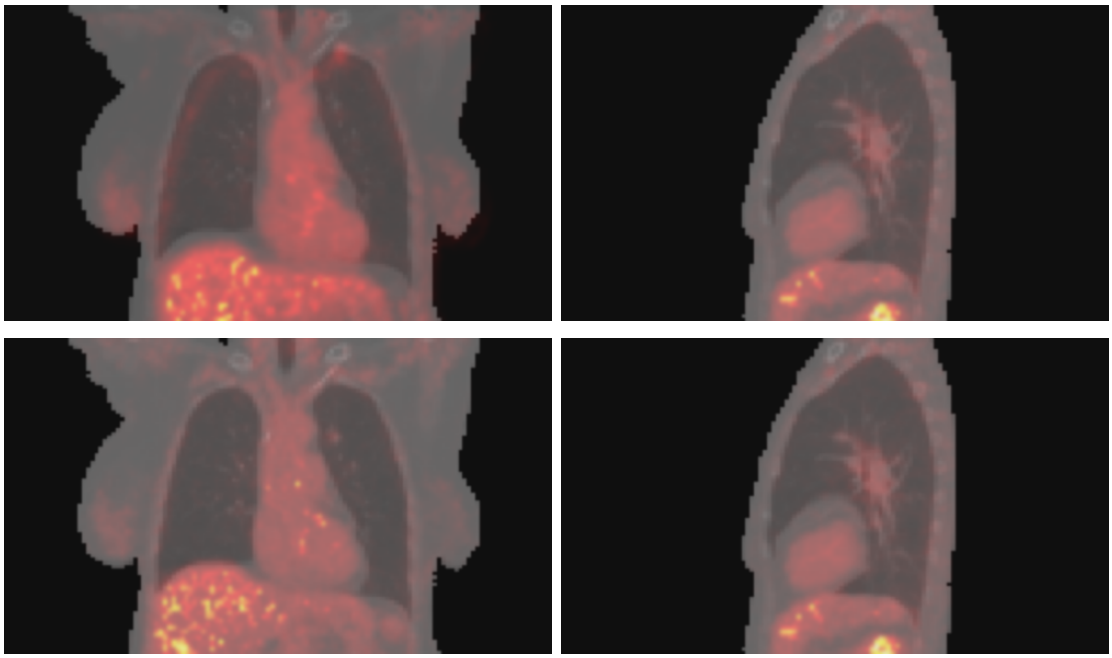


Figure 5.12: Sample from the PET CT registration data set. Registration was performed here with multi modal boosted maximum margin (CMBMM). Top row: fused images before registration, Bottom row: after registration

Chapter 6

Markov Random Field Training for Image Registration

The previously discussed similarity metrics (Chapter 4 and 5) both suffer from the same drawback. In order to learn the similarity metric we need to have at our disposition a data set of perfectly registered images. This predicament is usually quite seldom in clinical cases, where simultaneous acquisition is usually infeasable.

This fact led us to try and relax this constraint and work by matching organs boundaries. The registration process we have in mind involves detecting the boundaries of an organ and use this information as the driving force of the registration.

However boundary detection itself is already a challenge in the case of medical images. In this work we use a training data set of pairs of images that are not co-registered and we ask to also have manual segmentations of the organs of interest. Using these segmentations we learn probability distributions on the organs by boosting (section 6.1), and use Markov Random Fields training (section 6.3) to learn the correct amount of boundary smoothing to apply locally to get organ boundaries as close to the manual segmentation as possible.

In a sense the algorithm we present here performs concurrent segmentation and registration of the organs. Concurrent segmentation and registration has first been investigated in the neuroimaging community [Ashburner 1997]. Recent work include [Xiaohua 2004] where a Maximum a Posteriori Model is computed to take into account both segmentation and registration, [Gooya 2011] where Expectation maximization is used to incorporate information of tumor growth in the registration process, in [Lu 2011] where bayseian model of both registration and segmentation are learned and then assembled using a conditional modeling. All these approaches deal with inefficient bayesian modeling of the interactions which tend to be slow especially with large data such as medical images. Recently was proposed an interesting approach [Parisot 2012b] that uses a two leveled Markov Random Field to model the segmentation. One level controls the segmentation and the other con-

trols the registration. This method yields significantly faster performance. While being closer to this last work our method however lays the emphasis on the registration part, and even subpar segmentation results can lead to decent registration performances.

6.1 Boosting

The idea of boosting follows closely the one of Mixture of Experts (see section 4.2.3), in the setting of classification. For now let us focus on binary classification where the goal is to apply a label -1 or 1 to an unforeseen data sample, given learning on a labeled data-set.

Intuitively, let us assume we have access to an expert (in [Freund 1999] the authors present a horse race expert), we want to gain his knowledge and have access to a collection of data (that would be the health of the horse, the state of the track, the number of wins...). For each of these data, the expert is able to give a rule of thumb (play a horse that won in the last race for instance). On the whole the expert won't know how he makes his decisions, they are based on a lot of factors that he weights according to his own experience, but we can use this set of rule of thumbs to make an expert of our own. Some rule of thumbs might be contradictory, so we need to apply the right rule of thumb on the right data, and know which one to trust the most for all possible input data.

The idea of boosting is that given a data set on which we know the outcomes (a labeled data set), and given a base function (a weak learner in the boosting framework) which gives us a rule of thumb, we are going to learn an additive function, sum of several rule of thumbs on weighted data. Each round will add a rule of thumb to the decision function. The weights will be chosen according to the outcomes of each decision round, in order to emphasize the samples that have been misclassified in the previous rounds. The weights are boosting the misclassified samples, hence the name of these techniques.

6.1.1 AdaBoost

Let us have a look at one of the first very successful boosting algorithm: AdaBoost [Freund 1995] and is given in Algorithm 6.1. AdaBoost is a discrete boosting algorithm in the sense that the final classification function is a sum of weak learners that have their output values in $\{-1, 1\}$. Boosting methods with real valued weak learners have been also investigated and we will discuss one such method in section 6.1.2. Here is only discussed the boosting methods, discussions on the weak learner will be provided in section 6.1.3.

AdaBoost maintains a list of boosting weights $W_t(i)$ during iterations. The choice of α is done to make the new weighted problem maximally difficult for the weak learner as shown in equation 6.1. AdaBoost (or *Adaptive Boosting*) is a method that adapts to the error rates of the weak hypotheses.

Algorithm 6.1 Adaboost

Require: $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N) \in (\mathcal{X}, \{-1, 1\})$

{Initialization} $\quad \forall i \in \{1, \ldots, N\}, W_1(i) = \dfrac{1}{N}$

for $t = 1$ **to** T **do**

- Train weak learner using distribution W_t
- Get weak hypothesis $h_t : \mathcal{X} \to \{-1, 1\}$ with error

$$\epsilon_t = \sum_{\{i | h_t(\mathbf{x}_i) \neq y_i\}} W_t(i)$$

- Update α

$$\alpha_t \leftarrow \frac{1}{2} \log \left(\frac{1 - \epsilon_t}{\epsilon_t} \right)$$

- Update W

$$W_{t+1}(i) \leftarrow \frac{W_t(i)}{Z_t} \exp \left(-\alpha_t y_i h_t(\mathbf{x}_i) \right)$$

Where Z_t is chosen such that $\sum_i W_{t+1}(i) = 1$

end for

return

$$H(\mathbf{x}) = \text{sign} \left(\sum_{i=1}^{T} \alpha_t h_t(\mathbf{x}) \right)$$

$$\alpha_t = \underset{\alpha}{\operatorname{argmax}} \sum_i W_t(i) \exp\left(-\alpha y_i h_t(\mathbf{x}_i)\right) \tag{6.1}$$

Moreover in [Freund 1995] it is proven that the training error drops exponentially fast as soon as the weak learners are better than random, that is that the weighted expectation for the classification of the training set with the weak learner is better than $1/2$. If we denote by γ_t the distance of ϵ_t to $1/2$ then the error drops is bounded by the exponential of the square of γ_t.

Unlike many learning algorithms, boosting is said to be robust to over-fitting with the increase in T. In [Freund 1999], the authors present a bound with high probability on the generalization error that is independent on the number of rounds T, giving an insight on the claims of robustness to over-fitting.

In [Friedman 2000], an in-depth analysis of AdaBoost is given, where the problem is seen as the minimization of a functional with an additive logistic regression model. It is indeed proven that AdaBoost builds an additive logistic regression model via Newton-like updates for the minimization of the functional $J(H)$:

$$J(H) = E\left[\exp\left(-yH(\mathbf{x})\right)\right] \tag{6.2}$$

Where $E[\cdot]$ denotes the expectation. It immediately follows that $J(H)$ is minimized at:

$$H(\mathbf{x}) = \frac{1}{2} \log \frac{P\left(y = 1 \,|\mathbf{x}\right)}{P\left(y = -1 \,|\mathbf{x}\right)} \tag{6.3}$$

A very interesting result, that gives us access to the classification probabilities, which we will use in the development of this chapter:

$$P\left(y = 1 \,|\mathbf{x}\right) = \frac{\exp\left(H(\mathbf{x})\right)}{\exp\left(-H(\mathbf{x})\right) + \exp\left(H(\mathbf{x})\right)} \tag{6.4}$$

$$P\left(y = -1 \,|\mathbf{x}\right) = \frac{\exp\left(-H(\mathbf{x})\right)}{\exp\left(-H(\mathbf{x})\right) + \exp\left(H(\mathbf{x})\right)} \tag{6.5}$$

6.1.2 GentleBoost

Using the remarks done on AdaBoost, two new Boosting algorithms are presented in [Friedman 2000], namely *LogitBoost* and *Gentle AdaBoost* or *GentleBoost*, the idea of which is to circumvent the use of log-ratios (found in the setting of α) that can be unstable in extreme case of the error value ϵ. GentleBoost also makes use of real valued weak learners, the output of which then gives a confidence level on how well the classification

is done for each sample. A weak learner is then in essence a regression function. In the AdaBoost procedure, we chose h_t that minimizes the cost:

$$J(H_{t-1}(\mathbf{x}) + h_t(\mathbf{x})) = E\left[\exp\left(-yH_{t-1}(\mathbf{x}) + yh_t(\mathbf{x})\right)\right] \tag{6.6}$$

GentleBoost considers minimizing the Taylor approximation to this cost:

$$J(H) \propto E\left[\exp\left(-yH_{t-1}(\mathbf{x})\right)\left(y - h_t(\mathbf{x})\right)^2\right] \tag{6.7}$$

GentleBoost algorithm (6.2) is empirically shown to be more stable than other boosting algorithms.

Algorithm 6.2 GentleBoost

Require: $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N) \in (\mathcal{X}, \{-1, 1\})$

{Initialization} $\quad \forall i \in \{1, \ldots, N\}, W_1(i) = \dfrac{1}{N} \quad H(x) = 0$

for $t = 1$ **to** T **do**

- Fit the regression function $h_t(\mathbf{x})$ by weighted least-squares of y to $\mathbf{x}$ with weights W_t
- Update H

$$H(\mathbf{x}) \leftarrow H(\mathbf{x}) + h_t(\mathbf{x})$$

- Update W

$$W_{t+1}(i) \leftarrow \frac{W_t(i)}{Z_t} \exp\left(-y_i h_t(\mathbf{x}_i)\right)$$

Where Z_t is chosen such that $\sum_i W_{t+1}(i) = 1$

end for

return

$$\text{sign}\left(H(\mathbf{x})\right) = \text{sign}\left(\sum_{i=1}^{T} h_t(\mathbf{x})\right)$$

For the rest of this chapter, GentleBoost will be the algorithm of choice.

6.1.3 Choice of the weak learner

In the case of real valued weak learner, two types of functions are the most common. First and by far the most used is the regression stump function expressed as:

$$h_t(\mathbf{x}) = a1\left[\mathbf{x}_k < \theta\right] + b1\left[\mathbf{x}_k \geq \theta\right] \tag{6.8}$$

where $1[\cdot]$ is the indicator function and $\mathbf{x}_k$ denotes the k$^{\text{th}}$component of $\mathbf{x}$, four parameters have to be set for this function: a, b, θ and k. Finding these four parameters in a weighted least square setting is very fast, making the regression stump usually the function of choice with GentleBoost.

However a smoother function can lead to a better fit to the data, which then tends to diminish the required number of iterations T, unfortunately fitting a smoother function is usually more computationally expensive. Such a function can be a generalized logistic function:

$$h_t(\mathbf{x}) = a + \frac{b-a}{\left(1 + \theta \exp\left(-\lambda\left(\mathbf{x}_k - x_0\right)\right)\right)^{1/\nu}} \tag{6.9}$$

where 7 parameters have to be found.

In the discrete case, weak learners usually are decision trees, from the binary decision stump to more complex decision trees as can be found in [Freund 1996].

6.1.4 Multiclass Bossting

Until now we only have seen binary classification algorithms, and indeed boosting algorithms are intrinsically binary. However, very early on strategies have been presented to use boosting algorithms in a multiclass setting. *AdaBoost.M1* and *AdaBoost.M2* were presented in [Freund 1995], and *AdaBoost.MH*, *AdaBoost.MO* and *AdaBoost.MR* were presented in [Schapire 1999b].

Overall, two types of strategies are used. Seldom used is the *One against One* strategy, where one model is constructed for each pair of classes. Then the classification of an unknown sample is done by maximum voting where each model votes for one class. More commonly used is the *One against All* strategy, where there is one model constructed per class, and is trained to distinguish the samples from this class against the samples of all the other classes. Classification of an unknown sample is done by finding the maximum output of all models. Various variants of these methods have been used, such as the Hamming loss coding that is used in *AdaBoost.MH*. In our experiments we used a simple *One against All* strategy with yielded good performance.

It has to be noted that the energy that is minimized by this strategy then becomes:

$$J(H) = \sum_{i=1}^{C} E\left[\exp\left(-y_i H_i(\mathbf{x})\right)\right] \tag{6.10}$$

where C is the number of classes. Then through the minimizer of J, following equation 6.4, the probability of a sample to belong to a class C_i is given by:

$$P(\mathbf{x} \in C_i) = \frac{\exp\left(H_i(\mathbf{x})\right)}{\sum_{j=1}^{C} \exp\left(H_j(\mathbf{x})\right)} \tag{6.11}$$

6.1.5 Advantages and weaker aspects of boosting methods

One of the main advantages of boosting methods is the ease of implementation and the relative computational efficiency of the methods since AdaBoost for instance is of complexity $\mathcal{O}(dNT)$ where d is the dimension of $\mathcal{X}$. The second main advantage that we already discussed is the apparent robustness of Boosting methods to over-fitting which means in practice less time spent tuning one of the only parameters that is the number of rounds T.

However it has to be understood that these advantages come with a price. First, although boosting type of algorithms tend to have a naturally large margin as discussed in [Koltchinskii 2002] it is clear that the boosting methods such as AdaBoost and its variants were not designed to maximize the margin between samples and the decision boundary. And the choice of a weak learner also impacts on the margins, giving a trade off between over-fitting and margin maximization. The lack of explicit margin maximization leads boosting methods to be extremely sensitive to labeling noise, as shown in [Dietterich 2000].

6.1.6 Experiments with Multiclass GentleBoost

In this chapter we' will tackle the problem of CT to T2-MRI image registration of the liver. Due to lack of a proper image database and harsh memory requirements as will be seen later, all the experiments are conducted in 2D.

In the experiments on multiclass boosting we considered of a training data set of one 3D CT image of one patient consisting in 365 slices, of which we retained 35 for training. In each of the slices manual segmentation of the liver and the background has been performed, resulting in three classes, the liver, the background and the rest of the abdomen. An exemplar slice and its corresponding segmentation can be seen in figure 6.1. For each image we densely extract Gabor Features as explained in section 3.1, the extracted Gabor features have 15 orientations and 4 scale levels, and we extract 5×5 patches of the image. The Patches are rendered intensity shift invariant by removing the average intensity of the image (The DC component of the patch filter, see section 3.1). Altogether we end up with a 85 dimensional feature vector ($15 \times 4 + 5 \times 5$) for each pixel position in each of the 35 images. We use the manual segmentations as a labeling on the feature vector space and train the GentleBoost Multiclass Classifier on this data set.

In this experiment the testing data set consists in on 2D slice extraceted form the CT scaner of a second patient's liver. This image and a manual segmentation of it are provided in figure 6.2. Finally, we provide in figure 6.3 the probabilities estimated on the test image and the resulting segmentation. It has to be noted that the resulting classification result roughly detects the liver position, and this result alone could be satisfying for many applications (like detection). However, the resulting segmentation is very jittery and we would like to get closer to the manual segmentation, this will be the purpose of the next two sections.

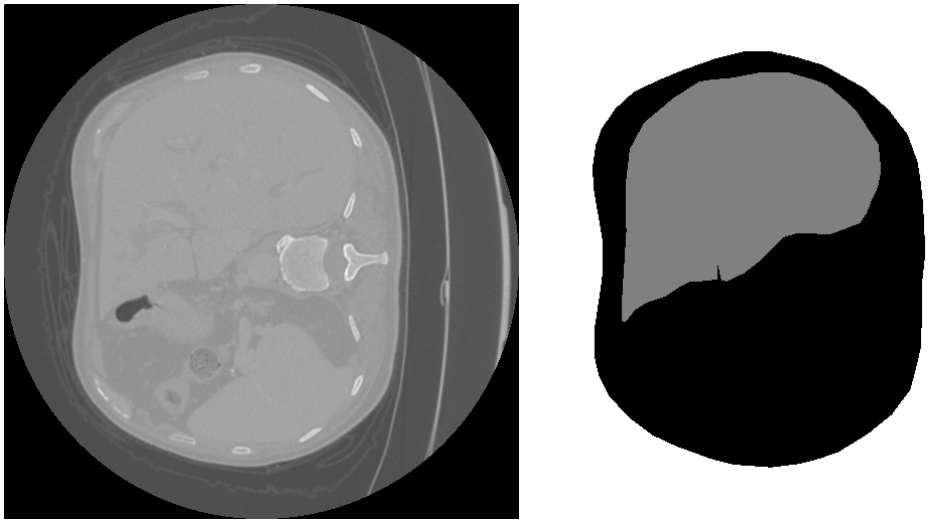


Figure 6.1: Exemplar image extracted from the CT image training data set and the companion segmentation

As we want to tackle the problem of CT to T2-MRI registration, we conducted the exact same experiment on T2 MRI images. For the training data set we used a 39 slice T2-MRI image of one patient of which we extracted 35 slices for training. An exemplar image alongside a manual segmentation of it can be found in figure 6.4. Testing was done on one slice of a second patient. This testing data set can be found in figure 6.5. Finally, estimated probabilities on the test image can be found in figure 6.6.

6.2 Markov Random Field Smoothing

Classification methods like boosting make one major assumption on the data, they assume that it is independent and identically distributed (iid). However when dealing with images (medical or not), it is obvious that any data extracted at one pixel position is not iid with

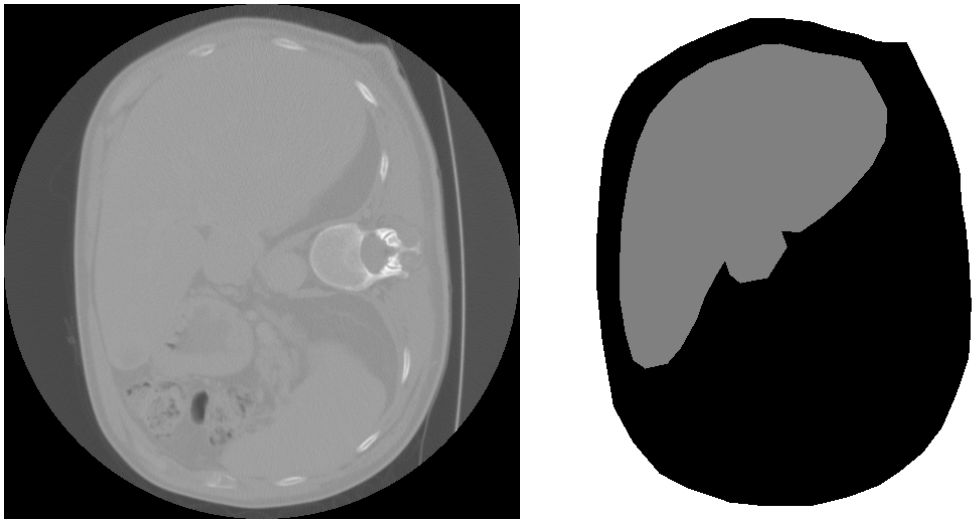


Figure 6.2: Testing data-set along side a manual segmentation of it

regards to data extracted in neighboring locations. This is due to the spatial coherence of an image, given one pixel value in one position it is often easy to predict the values of the neighboring pixels.

Markov Random Fields are very suitable to model this type of interaction. Indeed a Markov random field models each pixel label as a random variable. The Markovian property of this field states that each variable is only dependent on a neighborhood of variables (and not on all variables). Markov Random Field modeling for image processing was explored very early in [Geman 1984], but the algorithmic complexity for solving continuous Markov random fields problem made this solution impractical.

Advances in the domain of graph-cuts (as in [Boykov 2001]) and discrete Markov Random Fields (MRFs) (as in [Kolmogorov 2006, Komodakis 2008]) made the use of Markov Random Fields models tractable (we refer the reader to section: 2.3.2 for more discussions on discrete Markov Random Fields). Notably, MRFs have been used in many image segmentation applications as a way to propagate certainties and uncertainties among neighboring pixel locations [Boykov 2006]. In practice taking into account neighboring pixels dependence acts as a smart and localized smoothing of the segmentation map, hence the title of this section. MRF segmentation has been explored in medical applications notably in [Lee 2008, Besbes 2009].

In the remainder of this chapter we will consider a discrete Markov Random Field Model built on a graph G, the nodes of which correspond to the pixel locations in the image, the node system will be denoted $\mathcal{V}$. The edge system $\mathcal{E}$ will be discussed later in this section. Here we will only consider MRFs in which a node shares an interaction with

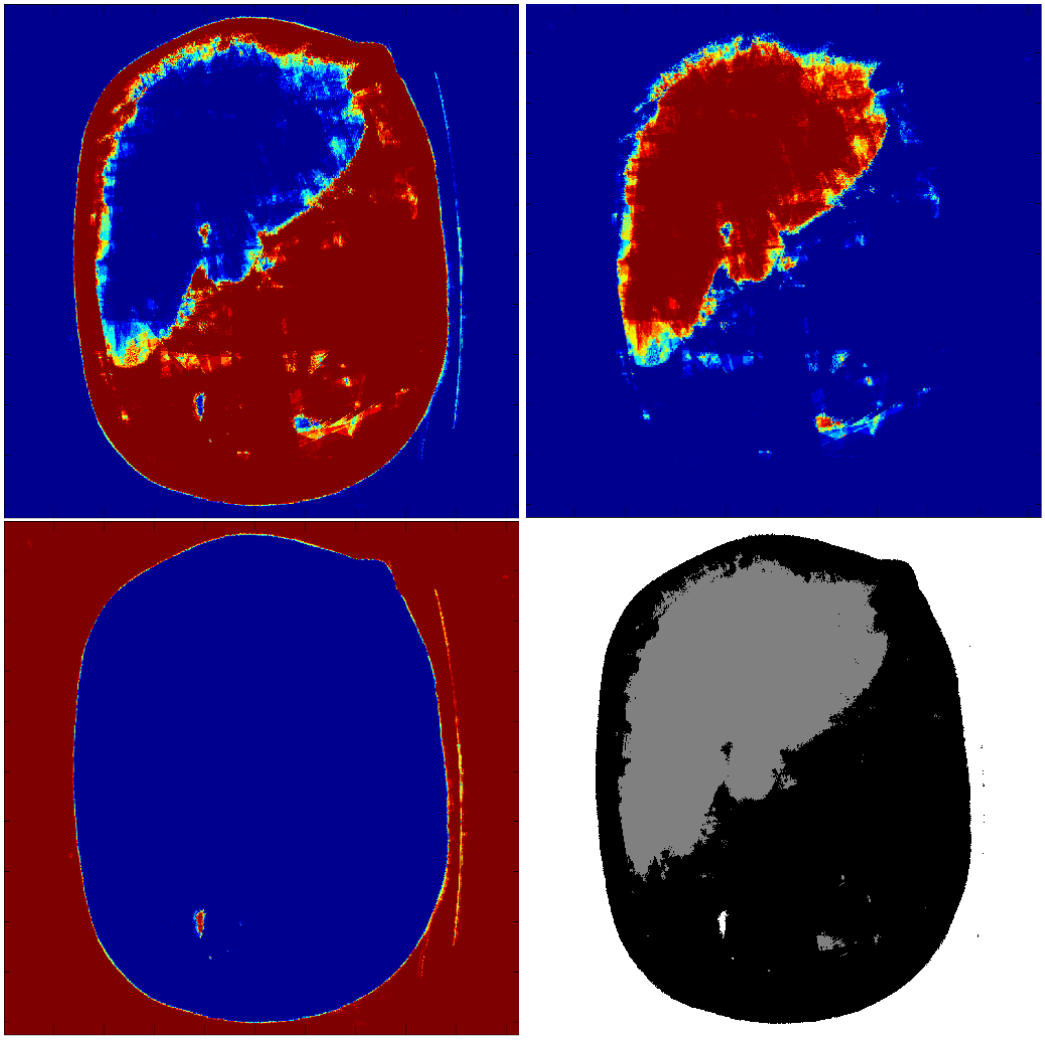


Figure 6.3: Probability distribution estimated on the test image, blues correspond to low probabilities and reds to high probability, the bottom right image is the resulting classification result.

one node at a time, a pairwise interaction.

The discrete MRF model as described in [Komodakis 2008] is formulated as a labeling problem in which each node is assigned a label $\ell \in \mathcal{L}$. We then want to find the optimal labeling that minimizes the energy:

$$\mathbf{E}(\boldsymbol{\ell}) = \sum_{p \in \mathcal{V}} u_p(\boldsymbol{\ell}_p) + \sum_{\{p,q\} \in \mathcal{E}} v_{p,q}(\boldsymbol{\ell}_p, \boldsymbol{\ell}_q) \tag{6.12}$$

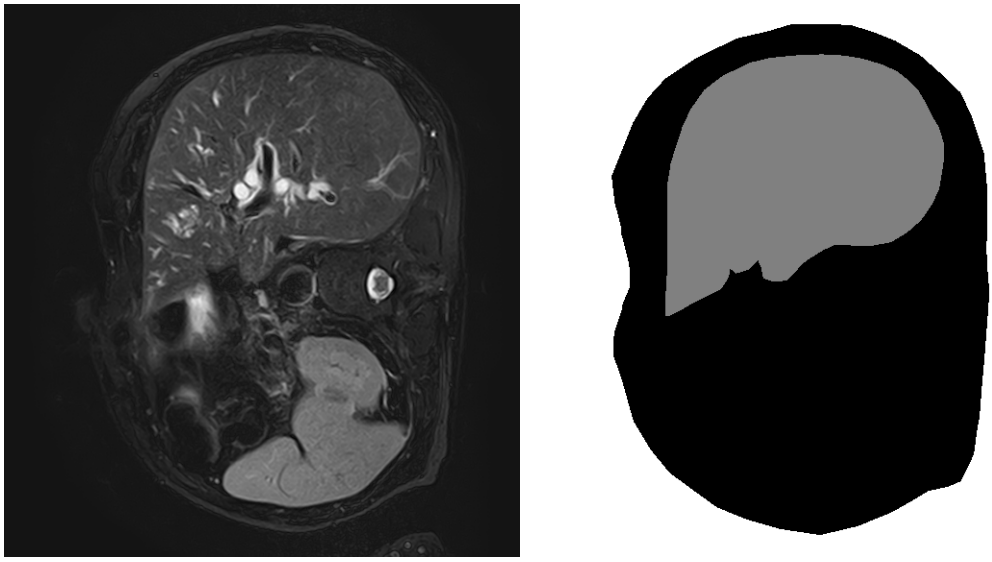


Figure 6.4: Exemplar image extracted from the T2-MRI image training data set and the companion segmentation

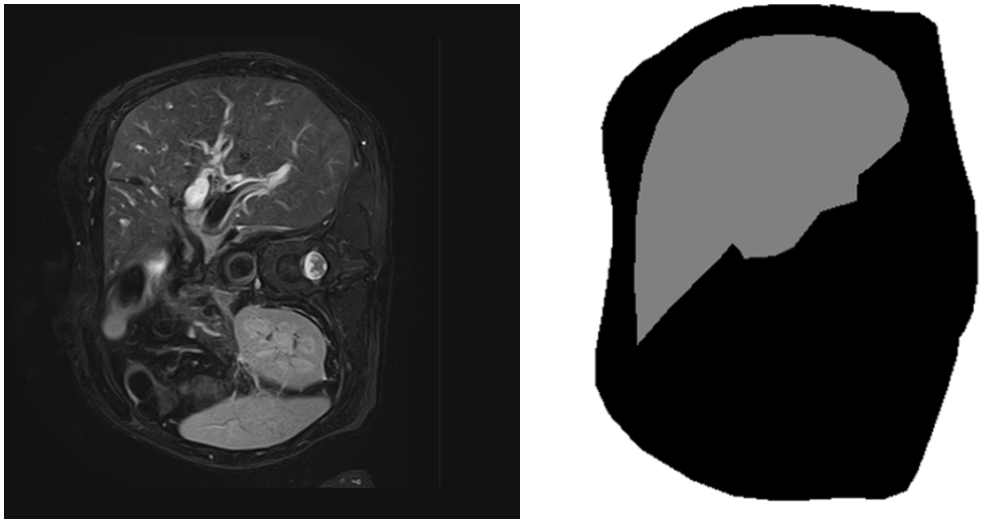


Figure 6.5: T2-MRI testing data-set along side a manual segmentation of it

In our case, the unary cost u_p will contain the information given by the GentleBoost classifier while the paiwise term v will be a term that penalizes changes in classes forcing a local smoothness constraint. The unary cost is modelled through the GentleBoost probability:

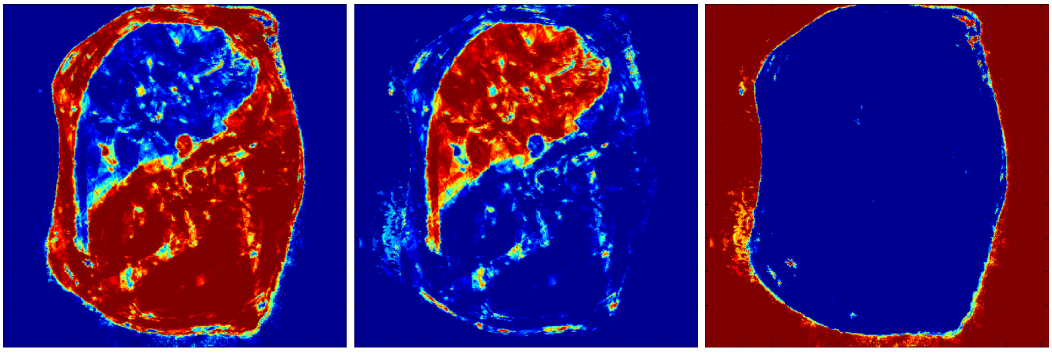


Figure 6.6: Probability distribution estimated on the T2-MRI test image, blues correspond to low probabilities and reds to high probability.

$$u_p(\boldsymbol{\ell}_p) = -\log\left(P\left(\pi(\mathbf{x}_p) \in C_{\boldsymbol{\ell}_p}\right)\right) \tag{6.13}$$

The pairwise cost that we will use throughout the remainder of this chapter follows a pairwise cost given in [Boykov 2006] and perfectly suited for feature driven segmentation:

$$v_{p,q}(\boldsymbol{\ell}_p, \boldsymbol{\ell}_q) = \lambda \exp\left(-\frac{\|\pi(\mathbf{x}_p) - \pi(\mathbf{x}_q)\|^2}{\sigma^2}\right)\delta_{p\neq q} \tag{6.14}$$

Where $\pi(\mathbf{x}_p)$ is the feature vector extracted in the pixel position of $\mathbf{x}_p$, δ the discrete Dirac distribution, σ and λ are parameters. Even though cross validation could be performed on σ to get the best possible value, we found in our experiments that using the standard deviation of the feature space gave good results. It is obvious that if λ is zero then the minimum of the energy coincides with the the gentleBoost classification. The pairwise term is equal to λ when the feature vectors are equal and the labels are not matching, and close to zero when the labels are not matching but the feature vectors are dissemblant. It is ineffective when the labels are matching. The idea is to penalize mismatching labeling when we have matching features but allow label mismatch when the feature distance shows a high likelihood of presenting a border. This term is known to be edge preserving as opposed to the δ function used on its own.

6.2.1 Neighborhood Paradigm

In our experiments we tested two kinds of Neighborhoods (figure 6.7). The first Neighborhood is a simple 4-Neighborhood as represented in figure 6.7(a) each node is paired with its immediate horizontal and vertical neighbor. Since this is done for each node in the graph, each node in the center of the graph ends up being connected to 4 nodes, hence the

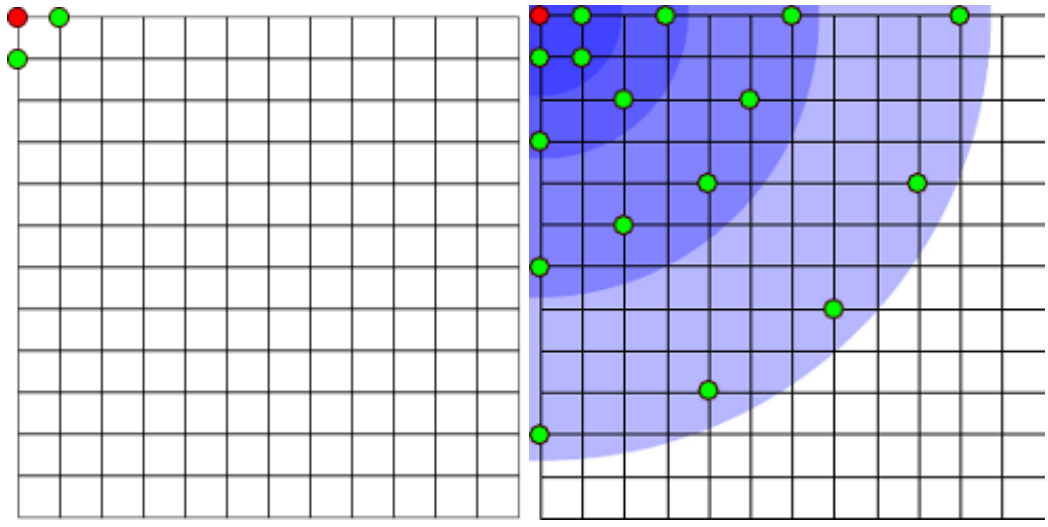
name. This Neighborhood system is widely used in the literature and usually yields good segmentation results. Segmentation results using this neighborhood are given in figure 6.8. It has to be noted that the results displayed here are the best results obtained with respect to the mean squared error computed with the ground truth segmentation when we change the parameter λ. There is obviously a trade off between an over smoothing of the segmentation that would get rid of the gray and white artifacts that appear in the segmentation but at the same time would result in an over smoothed liver segmentation.

We conducted a second set of experiments with a much more complex Neighborhood, which we name here *Circular Neighborhood*. Each node is connected with 16 nodes distributed on circles all originating from the central node as shown in figure 6.7(b). In the same fashion as with the 4-Neighborhood this means that a central node in the graph is connected to 48 nodes. The complexity of this Neighborhood will be fully understood in the next section, for now it is obvious that such a Neighborhood tends to over-smooth the segmentation and in figure 6.9 we show the best result obtained using the Neighborhood alongside what happens when we use a λ parameter slightly too high. We can see that this neighborhood gets rid of the artifacts we had with the 4-Neighborhood, but at the same time tends to over-smooth the segmentation. We would like to be able to retain the best of both worlds, by selectively activating the pairs that smooth the segmentation in the right places. Next section will attempt to solve this problem using state-of-the-art MRF-Learning techniques.

6.3 Markov Random Field Training

One of the drawbacks of Markov Random Field segmentation in the formulation that we used in the previous section is that each label gets the same kind of smoothing and even each label pairs. However we could set a coefficient in front of the pairwise term that is itself label dependent. This coefficient would help us favor some label pairs and not others. In the case of the circular Neighborhood we could also fancy having a coefficient for each circle and label pairs for instance, this would lead to vary how large the neighborhood (and in turn the smoothing) would be as a function of the label pairs. Unfortunately this kind of parametrization is intractable by hand we have 3 classes, it makes 9 pairs of labels with 4 circles we end up with 36 parameters to set. With cross-validation alone this process would be very long with no guarantee of optimality.

In [Komodakis 2011], Komodakis proposes an optimization strategy to infer the weighting parameters of a new Markov Random Field from a database of previously labeled MRFs. Let us consider two families of weights $\mathbf{w}_1$ and $\mathbf{w}_2$ that are weights to the unary and the pairwise cost respectively in the MRF formulation:



(a) 4-Neighborhood (b) Circular Neighborhood

Figure 6.7: Neighborhood Paradigm, the red circle symbolizes the central node, the green circles represent the nodes that are paired with the red node, (a) Simple 4-Neighborhood, (b) Circular Neighborhood where 16 pairs of nodes are distributed 4 circles depicted in blue.

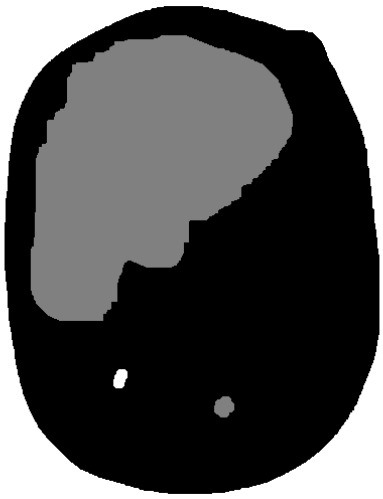


Figure 6.8: Best Segmentation result obtained with a 4-Neighborhood paradigm

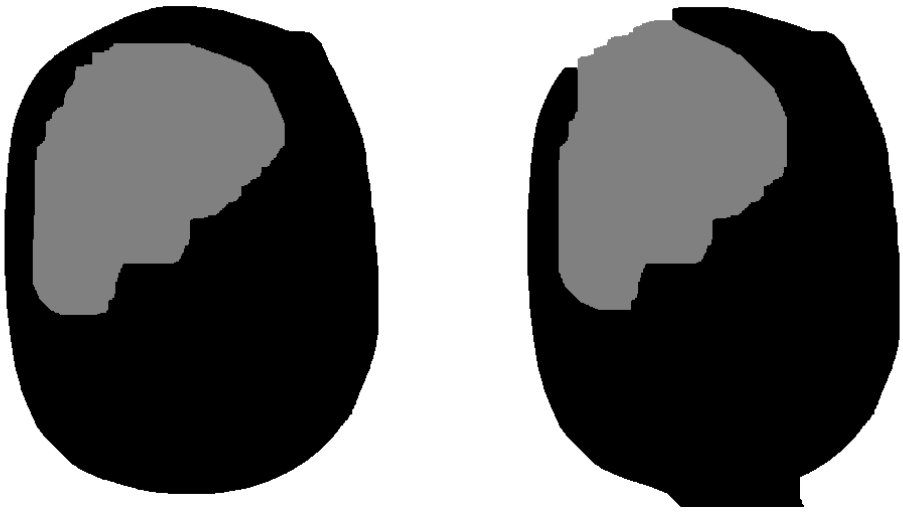


Figure 6.9: Best Segmentation result obtained with a 3 Circles Neighborhood paradigm and to the right Illustration of the segmentation degradation when the smoothness parameter is too large

$$\mathbf{E}(\boldsymbol{\ell}) = \sum_{p \in \mathcal{V}} \mathbf{w}_1 \cdot u_p(\boldsymbol{\ell}_p) + \sum_{\{p,q\} \in \mathcal{E}} \mathbf{w}_2 \cdot v_{p,q}(\boldsymbol{\ell}_p, \boldsymbol{\ell}_q) \tag{6.15}$$

there is no restriction on the action of the weights in this modelization and we could actually have different weights for each and every node for each and every label combinations. In this work we will restrict ourselves to have one $\mathbf{w}_1(\boldsymbol{\ell}_p)$ for each label and one $\mathbf{w}_2\left(\boldsymbol{\ell}_p, \boldsymbol{\ell}_q, circle(q)\right)$ for each label pair and each circle in the circular Neighborhood, denoted as $circle(q)$.

The MRF formulation then becomes:

$$\mathbf{E}^{\mathbf{w}}(\boldsymbol{\ell}) = \sum_{p \in \mathcal{V}} \mathbf{w}_1(\boldsymbol{\ell}_p) \cdot u_p(\boldsymbol{\ell}_p) + \sum_{\{p,q\} \in \mathcal{E}} \mathbf{w}_2\left(\boldsymbol{\ell}_p, \boldsymbol{\ell}_q, circle(q)\right) \cdot v_{p,q}(\boldsymbol{\ell}_p, \boldsymbol{\ell}_q) \tag{6.16}$$

Now let us assume that we have access to a collection of K graphs G_k, for which we know the unary and pairwise potentials for every label configuration and the ground-truth labeling that we will denote $\bar{\mathcal{L}}_k$. In a maximum margin Markov network we seek the parameters $\mathbf{w}_1$ and $\mathbf{w}_2$ such that the MRF energy of the desired solution $\bar{\mathcal{L}}_k$ is smaller than the MRF energy of any other solution $\mathcal{L}_k$.
The minimization problem is expressed as:

$$\min_{\mathbf{w}} \left\{ \mu R(\mathbf{w}) + \sum_{k=1}^{K} \left[\mathbf{E}^{\mathbf{w}}_{G_k} \left(\bar{\mathcal{L}_k} \right) - \min_{\mathcal{L}_k} \left\{ \mathbf{E}^{\mathbf{w}}_{G_k} \left(\mathcal{L}_k \right) \right\} \right] \right\} \tag{6.17}$$

where $R(\mathbf{w})$ is a regularization term and was set equal to $\frac{1}{2} \|\mathbf{w}\|^2$ in our case. Equation 6.17 is not solvable in polynomial time since even the estimation of the second term is not tractable. As was the case with FastPD [Komodakis 2008], [Komodakis 2011] uses a primal dual strategy to solve equation 6.17. Each graph G_k is decomposed in sub-hypergraphs G_k^i and the minimization of the Dual approximation to the equation is done by a projected sub-gradient algorithm. Details of the algorithm can be found in [Komodakis 2011].

6.3.1 Experiments

For our experiments we use the same potentials that were used in the previous section, however the training set is not formed similarly to section 6.1.6. The images that were used as testing images are now part of the training data set, the training data-set is composed of 35 CT slices of one patient along with their manual segmentation and 30 CT slices of a second patient also segmented. Of those 65 slices, 45 randomly drawn slices were used for the training of the boosting classifier and the remaining 20 slices were used for the MRF training. 1000 iterations of the projected sub-gradient algorithm were necessary which amounts to 6 hours of computation on a regular desktop computer at the time of writing. The parameter μ was taken equal to 1 in all experiments. We assess the convergence of the algorithm by looking at the error between the estimated labels and the actual ground-truth labels. In the remainder of this chapter only the circular neighborhood will be taken into consideration.

In figure 6.10 is shown the image on which we show the effectiveness of this new approach. This is a new slice extracted on the CT-scan of a third patient. Training the MRF gives us access to the familly of optimal parameters $\mathbf{w}$, using them we solve problem 6.16 in the same way as in the previous section. The resulting segmentation is provided in figure 6.11. As a mean of comparison we provide in figure 6.12 the segmentation results obtained only GentleBoost classification and a MRF segmentation using the circular Neighborhood (best result shown).

6.4 Multi-Modal Image Registration with MRF Training

Let us assume that we have access to a data-base multi-modal images that have been manually segmented. At least the organs of interest have been dutifully segmented in

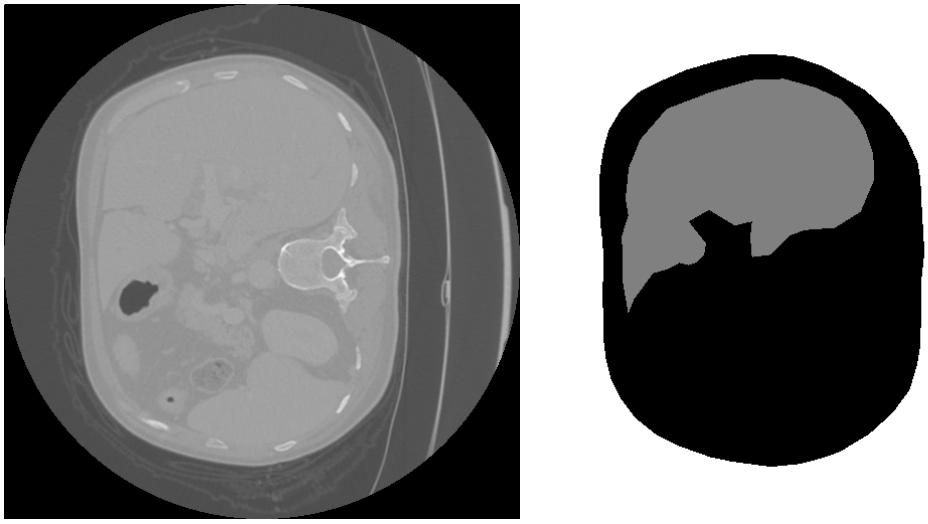


Figure 6.10: New testing CT image extracted from the CT scan of a third patient.

Figure 6.11: Result obtained with MRF training.

both modalities. Note that here we do not require that the images in the data base are co-registered, we merely require that all images are at least rigidly registered to better focus the learning process on deformed features instead of large translations and rotations.

For each modality, using the data-set we can learn the probability of a pixel in the image to belong to a given class using GentleBoost (see section 6.1). More over, for the source modality we can learn the MRF parameters that yield the best segmentation results.

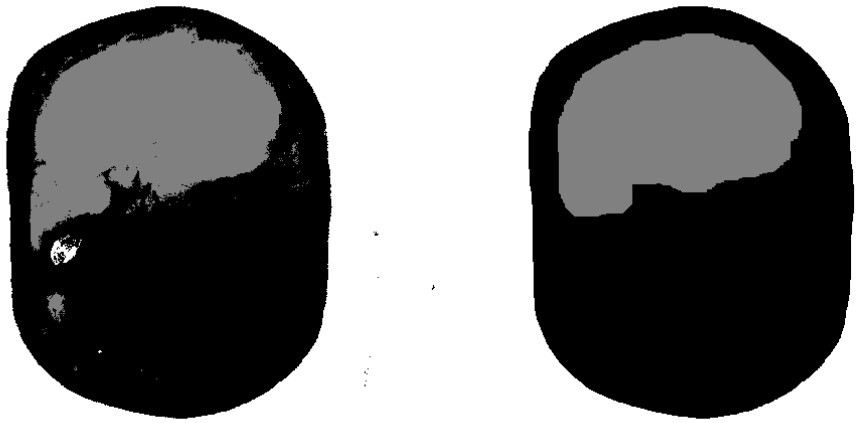


Figure 6.12: Boosting segmentation results (left) and MRF segmentation(right) results obtained on the testing set of figure 6.10, provided as a mean of comparison with figure 6.11

Let us denote by $MRFT(J, \mathbf{x})$ the class that was given to pixel position $\mathbf{x}$ in source image J by the MRF segmentation scheme weighted by MRF training. Then we can use the probability of a feature vector of the target image to belong to this class as a similarity measure:

$$\mathcal{C}(J(\mathbf{x}), I(\mathbf{y})) = -\log(P_I(\pi(\mathbf{y}) \in MRFT(J, \mathbf{x}))) \tag{6.18}$$

The idea behind this similarity measure for registration is for each iteration to compute the segmentation of the source image J using MRF segmentation and the coefficients learned by MRF training. Then using the probability computed using the GentleBoost coefficients learned on a set of images similar to the target image, we drive the registration by matching each sample from the source image to the sample in the target image it has the most probability to belong to.

MRF segmentation is a really fast process using FastPD [Komodakis 2008], and the estimation of the probabilities for the source image to feed the MRF segmentation is a matter of a matrix product and can be made in a really efficient fashion. Since the target image is not moving, the probability map on the target image is computed beforehand.

Plugging in this similarity criterion in the algorithm of [Glocker 2009] explained in section 2.3.2, we get registration results for a 2D image of size 512×512 in about 20 minutes. The real limitation of this algorithm is the training stage of the MRF where several images have to be taken into consideration with quite a large Neighborhood system,

if extended to 3D this Neighborhood system would become even larger and memory becomes the limiting factor (more than 200GB of memory needed). This is why in all our experiments we use 2D images as a proof of concept of our method.

6.4.1 Experiments

Essentially the same data sets were used in the registration experiments and all the previous experiments. Only the size of the training database has grown since 3 patients were used for training and one patient was used for testing and the testing patient was alternatively changed between the testing and the training data set in a leave one out cross validation fashion. As it can be seen, only the number of iterations on the GentleBoost algorithm is a required parameter of the registration training algorithm, we used a number of iterations equal to 1000. The number of iterations of the MRF training algorithm could be set automatically by detecting when convergence is reached in the projected sub-gradient algorithm and as such is not discussed here.

For our experiments we used 120 image slices for training the source CT image that were randomly separated in 80 images for training the boosting algorithm and 40 for training the MRF. 80 image slices from the T2-MRI images were extracted for the training of the probability distribution. Registration tests were done on 5 images in each patient resulting in 20 possible registration.

For the evaluation of the quality of registration we used manual segmentations of the liver in both source and target images and look at the evolution of the Dice coefficient before and after registration, the bigger the increase, the better the registration.

We expect that a criterion that was based on the learning of specific liver segmentations will yield better results on the Dice coefficient computed on the liver than a criterion that makes no distinction between organs.

As with previous sections, and since registration are only equal up to a smoothing parameter, which is not comparable across similarity measure, we chose to use the invariant measure of the Harmonic Energy, on 20 different settings of the smoothing parameter. This makes for a total of 400 registration experiments. Comparison was made against Mutual Information criterion which was the only one to give decent results on this data set, all other commonly criteria, including Normalized Mutual information failed at registration and gave negative dice coefficient increase. We deem this data set as extremely in this regard.
Experimentation results are given in figure 6.13.

We can see that our method does slightly better on the whole than mutual information however, the dice coefficient increase is not very significant, and since all other similarity measure used failed, we have reasons to believe that this is in part due to the extreme

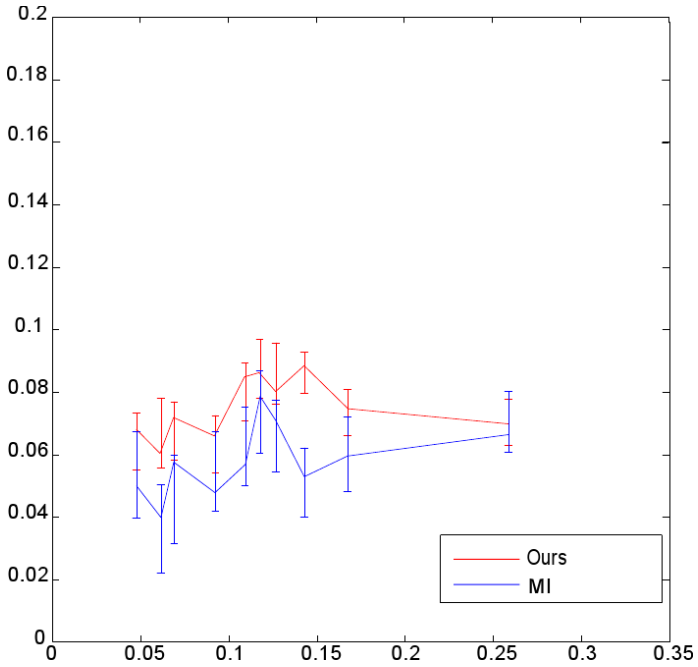


Figure 6.13: Evolution of the Dice coefficient increase as a function the harmonic energy, 400 registration experiments were necessary for this graph. The solid line represents the average case while each end of the whiskers represent the minimum and the maximum value.

challenge posed by this data-set.

6.5 Conclusion and future work

The results presented here are promising and we believe that further studies will prove that this method can potentially be a good alternative to state of the art similarity measures, provided that a training set of source and target manual segmentation is accessible.

An interesting extension to this work could be made by investigating the effects of reweighting a commonly used similarity cost such as Mutual Information with the probability used in equation 6.18. Indeed our similarity criterion is basically acting on the boundaries of the organs and lets the regularization term act on the inside of the organ. We believe that using such a reweighted criterion could help get the best of both worlds and drive registration even inside the organs.

Chapter 7

Conclusion

In this thesis we made an attempt to show how the recent advances in optimization and machine learning can allow us to make efficient and robust algorithms for image registration. Let us have a look at the major contributions of this work.

7.1 Contributions

The main contribution of this thesis is to show the possibility of effectively learning a similarity measure in a multi-modal case. A subject that has attracted a lot of interest along the years is the subject of metric learning. To the best of our knowledge, Metric Learning was not adapted to the case where similar data was not originating from the same space. We have shown that it is possible to modify some state of the art metric learning algorithms to apply them to the multi-modal case with impressive results on sometimes very challenging data sets such as the PET-CT data set.

In the same way we provided a very simple method targeted at utilizing an off the shelf maximum margin metric learning algorithm in the multi modal case. The resulting similarity measure is differentiable and usages of this technique go way beyond the framework of this thesis

Regression of medical images for registration was considered as a first attempt at multi-modal image registration. While this approach only uses the information of one image modality in the learning stage, local image regression (the regression function does not depend on the local position) can be still applied in image restoration and many computer vision applications.

Last but not least, we presented a method for multi-modal metric learning that is based on concurrent segmentation and registration of the images. The major advantage of this

method over the methods presented previously is that it allows to learn a similarity criterion on a data-set of images that are not aligned, since no point to point correspondence is necessary to perform the learning. We believe that this type of methods paves the way to a broad type of methods where heavy learning is used to create per instance similarity criterions allowing much better alignment.

7.2 Future Work

In this work we have provided promising solutions to the very challenging task of multi-modal image registration, yet immediate extensions of this work can be envisioned:

- Metric Learning algorithms presented here all share the same feature, they are all based on classification algorithms, and provide a confidence of the classification between similar and not similar as a measure in the common space. However we have access in the learning data set to a much richer information, that is a full fledged metric (taking for instance the maximum of two intra-modality distances for the same samples, could be used as an inter-modality distance for learning). The distance could then be learned by regression and not classification leading to a much more appropriate measure.

- The combination of the algorithms in chapter 5 and 6 could lead to very promising results. Using the segmentation learning to find the organs in each image and then appliying a per organ distance criterion, that was learned on similar organs would dramatically improve the results of both methods.

- Probably the most promising direction for metric learning in multi-modal image registration will be online metric learning. Indeed a similarity criterion could be refined as the registration process is undertaken using the correspondences found on the fly to make a better similarity criterion that would in turn make a better registration.

Publications by the Author

- F. Michel and N. Paragios. *Image transport regression using mixture of experts and discrete Markov random fields*. In Biomedical Imaging: From Nano to Macro, 2010 IEEE International Symposium on, pages 1229–1232. IEEE, 2010.

- C. Wang, O. Teboul, F. Michel, S. Essafi and N. Paragios. *3D knowledge-based segmentation using pose-invariant higher-order graphs*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2010, pages 189–196, 2010.

- M.M. Bronstein, A.M. Bronstein, F. Michel and N. Paragios. *Data fusion through cross-modality metric learning using similarity-sensitive hashing*. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 3594–3601. IEEE, 2010.

- F. Michel, M. Bronstein, A. Bronstein and N. Paragios. *Boosted metric learning for 3d multi-modal deformable registration*. In Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on, pages 1209–1214. IEEE, 2011.

Bibliography

[Ashburner 1997] J Ashburner and K Friston. *Multimodal image coregistration and partitioning - a unified framework*. Neuroimage, vol. 6, no. 3, pages 209–217, 1997. 113

[Bar-Hillel 2003] A. Bar-Hillel, T. Hertz, N. Shental and D. Weinshall. *Learning distance functions using equivalence relations*. In MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-, volume 20, page 11, 2003. 84

[Bar-Hillel 2006] A. Bar-Hillel, T. Hertz, N. Shental and D. Weinshall. *Learning a mahalanobis metric from equivalence constraints*. Journal of Machine Learning Research, vol. 6, no. 1, page 937, 2006. 84

[Bardera 2006] A. Bardera, M. Feixas, I. Boada and M. Sbert. *High-dimensional normalized mutual information for image registration using random lines*. Biomedical Image Registration, pages 264–271, 2006. 22

[Belkin 2003] M. Belkin and P. Niyogi. *Laplacian eigenmaps for dimensionality reduction and data representation*. Neural computation, vol. 15, no. 6, pages 1373–1396, 2003. 81

[Bengio 2004] Y. Bengio, J.F. Paiement, P. Vincent, O. Delalleau, N. Le Roux and M. Ouimet. *Out-of-sample extensions for lle, isomap, mds, eigenmaps, and spectral clustering*. Advances in neural information processing systems, vol. 16, pages 177–184, 2004. 82

[Bernardino 2006] A. Bernardino and J. Santos-Victor. *Fast IIR isotropic 2-D complex Gabor filters with boundary initialization*. Image Processing, IEEE Transactions on, vol. 15, no. 11, pages 3338–3348, 2006. 38

[Besbes 2009] A. Besbes, N. Komodakis, G. Langs and N. Paragios. *Shape priors and discrete mrfs for knowledge-based segmentation*. In Computer Vision and Pattern

Recognition, 2009. CVPR 2009. IEEE Conference on, pages 1295–1302. IEEE, 2009. 121

[Bishop 2006] C.M. Bishop and SpringerLink (Service en ligne). Pattern recognition and machine learning, volume 4. springer New York, 2006. 48, 56

[Blackall 2000] J. Blackall, D. Rueckert, C. Maurer, G. Penney, D. Hill and D. Hawkes. *An image registration approach to automated calibration for freehand 3D ultrasound*. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2000. Springer, 2000. 20

[Bookstein 1989] F.L. Bookstein. *Principal warps: Thin-plate splines and the decomposition of deformations*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 11, no. 6, pages 567–585, 1989. 103

[Boykov 2001] Y. Boykov, O. Veksler and R. Zabih. *Fast approximate energy minimization via graph cuts*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 23, no. 11, pages 1222–1239, 2001. 121

[Boykov 2006] Y. Boykov and G. Funka-Lea. *Graph cuts and efficient ND image segmentation*. International Journal of Computer Vision, vol. 70, no. 2, pages 109–131, 2006. 121, 124

[Bronstein 2010] M.M. Bronstein, A.M. Bronstein, F. Michel and N. Paragios. *Data fusion through cross-modality metric learning using similarity-sensitive hashing*. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 3594–3601. IEEE, 2010. 93

[Cachier 2000] Pascal Cachier and Xavier Pennec. *3D non-rigid registration by gradient descent on a gaussian-windowed similarity measure using convolutions*. In Mathematical Methods in Biomedical Image Analysis, 2000. Proceedings. IEEE Workshop on, pages 182–189. IEEE, 2000. 18

[Cachier 2003] P. Cachier, E. Bardinet, D. Dormont, X. Pennec and N. Ayache. *Iconic feature based nonrigid registration: the PASHA algorithm*. Computer Vision and Image Understanding, vol. 89, no. 2, pages 272–298, 2003. 11

[Castellano-Smith 2001] A. Castellano-Smith, T. Hartkens, J. Schnabel, D. Hose, H. Liu, W. Hall, C. Truwit, D. Hawkes and D. Hill. *Constructing patient specific models for correcting intraoperative brain deformation*. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2001, pages 1091–1098. Springer, 2001. 20

[Chechik 2009] G. Chechik, V. Sharma, U. Shalit and S. Bengio. *An online algorithm for large scale image similarity learning*. In Proc. NIPS, volume 1. Citeseer, 2009. 91

[Chechik 2010] G. Chechik, V. Sharma, U. Shalit and S. Bengio. *Large scale online learning of image similarity through ranking*. The Journal of Machine Learning Research, vol. 11, pages 1109–1135, 2010. 91

[Choi 2000] Y. Choi and S. Lee. *Injectivity conditions of 2D and 3D uniform cubic B-spline functions*. Graphical models, vol. 62, no. 6, pages 411–427, 2000. 12

[Christensen 1994] G.E. Christensen. *Deformable shape models for anatomy*. PhD thesis, Washington University, 1994. 16

[Chung 2002] A. Chung, W. Wells, A. Norbash and W. Grimson. *Multi-modal image registration by minimising kullback-leibler distance*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2002, pages 525–532, 2002. 14, 22

[Collignon 1995] A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens and G. Marchal. *Automated multi-modality image registration based on information theory*. In Information processing in medical imaging, volume 3, pages 264–274, 1995. 19

[Collins 2002] M. Collins, R.E. Schapire and Y. Singer. *Logistic regression, AdaBoost and Bregman distances*. Machine Learning, vol. 48, no. 1, pages 253–285, 2002. 87

[Cox 2001] T.F. Cox and M.A.A. Cox. Multidimensional scaling, volume 1. CRC Press, 2001. 78, 79

[Davis 2007] J.V. Davis, B. Kulis, P. Jain, S. Sra and I.S. Dhillon. *Information-theoretic metric learning*. In Proceedings of the 24th international conference on Machine learning, pages 209–216. ACM, 2007. 87, 91

[Delon 2004] J. Delon. *Midway image equalization*. Journal of Mathematical Imaging and Vision, vol. 21, no. 2, pages 119–134, 2004. 102

[Deriche 1993] R. Deriche. *Recursively implementating the Gaussian and its derivatives*. 1993. 39

[Dietterich 2000] T. Dietterich. *Ensemble methods in machine learning*. Multiple classifier systems, pages 1–15, 2000. 97, 119

[Felsberg 2001] M. Felsberg and G. Sommer. *The monogenic signal*. Signal Processing, IEEE Transactions on, vol. 49, no. 12, pages 3136–3144, 2001. 32

[Freeman 1991] W.T. Freeman, E.H. Adelson, Massachusetts Institute of Technology. Media Laboratory. Vision and Modeling Group. *The design and use of steerable filters*. IEEE Transactions on Pattern analysis and machine intelligence, vol. 13, no. 9, pages 891–906, 1991. 32

[Freund 1995] Y. Freund and R. Schapire. *A desicion-theoretic generalization of on-line learning and an application to boosting*. In Computational learning theory, pages 23–37. Springer, 1995. 95, 114, 116, 118

[Freund 1996] Y. Freund and R.E. Schapire. *Experiments with a new boosting algorithm*. In MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-, pages 148–156. MORGAN KAUFMANN PUBLISHERS, INC., 1996. 118

[Freund 1999] Y. Freund, R. Schapire and N. Abe. *A short introduction to boosting*. Journal-Japanese Society For Artificial Intelligence, vol. 14, no. 771-780, page 1612, 1999. 114, 116

[Friedman 2000] J. Friedman, T. Hastie and R. Tibshirani. *Additive logistic regression: a statistical view of boosting (With discussion and a rejoinder by the authors)*. The annals of statistics, vol. 28, no. 2, pages 337–407, 2000. 116

[Geman 1984] S. Geman and D. Geman. *Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, no. 6, pages 721–741, 1984. 121

[Geusebroek 2003] J.M. Geusebroek, A.W.M. Smeulders and J. Van De Weijer. *Fast anisotropic gauss filtering*. Image Processing, IEEE Transactions on, vol. 12, no. 8, pages 938–943, 2003. 39

[Globerson 2006] A. Globerson and S. Roweis. *Metric learning by collapsing classes*. Advances in neural information processing systems, vol. 18, page 451, 2006. 85

[Glocker 2008] B. Glocker, N. Komodakis, G. Tziritas, N. Navab and N. Paragios. *Dense image registration through MRFs and efficient linear programming*. Medical Image Analysis, vol. 12, no. 6, pages 731–741, 2008. 15, 16, 101

[Glocker 2009] B. Glocker, N. Komodakis, N. Navab, G. Tziritas and N. Paragios. *Dense registration with deformation priors*. In Information Processing in Medical Imaging, pages 540–551. Springer, 2009. 16, 130

[Goldberger 2004] J. Goldberger, S. Roweis, G. Hinton and R. Salakhutdinov. *Neighbourhood components analysis*. 2004. 85

[Gooya 2011] A. Gooya, K. Pohl, M. Bilello, G. Biros and C. Davatzikos. *Joint segmentation and deformable registration of brain scans guided by a tumor growth model*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2011, pages 532–540, 2011. 113

[Grau 2007] V. Grau, H. Becher and J.A. Noble. *Registration of multiview real-time 3-D echocardiographic sequences*. Medical Imaging, IEEE Transactions on, vol. 26, no. 9, pages 1154–1165, 2007. 34

[Guetter 2005] C. Guetter, C. Xu, F. Sauer and J. Hornegger. *Learning based non-rigid multi-modal image registration using Kullback-Leibler divergence*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2005, pages 255–262, 2005. 22

[Hamza 2003] A. Hamza and H. Krim. *Image registration and segmentation by maximizing the jensen-rényi divergence*. In Energy Minimization Methods in Computer Vision and Pattern Recognition, pages 147–163. Springer, 2003. 21

[Han 2010] X. Han. *Feature-constrained nonlinear registration of lung CT images*. Medical Image Analysis for the Clinic: A Grand Challenge, pages 63–72, 2010. 34

[Haralick 1973] R.M. Haralick, K. Shanmugam and I.H. Dinstein. *Textural features for image classification*. Systems, Man and Cybernetics, IEEE Transactions on, vol. 3, no. 6, pages 610–621, 1973. 29, 30

[He 2003] Y. He, A.B. Hamza and H. Krim. *A generalized divergence measure for robust image registration*. Signal Processing, IEEE Transactions on, vol. 51, no. 5, pages 1211–1220, 2003. 21

[Hermosillo 2002] Gerardo Hermosillo, Christophe Chefd'Hotel and Olivier Faugeras. *Variational methods for multimodal image matching*. International Journal of Computer Vision, vol. 50, no. 3, pages 329–343, 2002. 18

[Hofmann 2008] M. Hofmann, F. Steinke, V. Scheel, G. Charpiat, J. Farquhar, P. Aschoff, M. Brady, B. Schölkopf and B. J. Pichler. *MRI-Based Attenuation Correction for PET/MRI: A Novel Approach Combining Pattern Recognition and Atlas Registration*. Journal of Nuclear Medicine, 2008. xiv, 47, 48, 60

[Honnorat 2010] N. Honnorat, R. Vaillant and N. Paragios. *Guide-wire extraction through perceptual organization of local segments in fluoroscopic images*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2010, pages 440–448, 2010. 32

[Jacob 2004] M. Jacob and M. Unser. *Design of steerable filters for feature detection using Canny-like criteria*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 26, no. 8, pages 1007–1019, 2004. 32

[Jacobs 1991] R.A. Jacobs, M.I. Jordan, S.J. Nowlan and G.E. Hinton. *Adaptive mixtures of local experts*. Neural computation, vol. 3, no. 1, pages 79–87, 1991. 52

[Jain 2008] P. Jain, B. Kulis, I.S. Dhillon and K. Grauman. *Online Metric Learning and Fast Similarity Search*. Advances in Neural Information Processing Systems 21, pages 761–768, 2008. 91

[Jenkinson 2001] M. Jenkinson and S. Smith. *A global optimisation method for robust affine registration of brain images*. Medical image analysis, vol. 5, no. 2, pages 143–156, 2001. 9

[Juan 2009] L. Juan and O. Gwun. *A comparison of sift, pca-sift and surf*. International Journal of Image Processing, vol. 3, no. 4, pages 143–152, 2009. 34

[Karaçali 2007] B. Karaçali. *Information theoretic deformable registration using local image information*. International journal of computer vision, vol. 72, no. 3, pages 219–237, 2007. 22

[Keller 2006] P.W. Keller, S. Mannor and D. Precup. *Automatic basis function construction for approximate dynamic programming and reinforcement learning*. In Proceedings of the 23rd international conference on Machine learning, pages 449–456. ACM, 2006. 85

[Kim 2004] J. Kim and J.A. Fessler. *Intensity-based image registration using robust correlation coefficients*. Medical Imaging, IEEE Transactions on, vol. 23, no. 11, pages 1430–1444, 2004. 18

[Klein 2007] S. Klein, M. Staring and J.P.W. Pluim. *Evaluation of optimization methods for nonrigid medical image registration using mutual information and B-splines*. Image Processing, IEEE Transactions on, vol. 16, no. 12, pages 2879–2890, 2007. 13

[Kokkinos 2008] I. Kokkinos and A. Yuille. *Scale invariance without scale selection*. In Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pages 1–8. IEEE, 2008. xiii, 33, 39

[Kolmogorov 2006] V. Kolmogorov. *Convergent tree-reweighted message passing for energy minimization*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 28, no. 10, pages 1568–1583, 2006. 16, 121

[Koltchinskii 2002] Vladimir Koltchinskii and Dmitry Panchenko. *Empirical margin distributions and bounding the generalization error of combined classifiers*. The Annals of Statistics, vol. 30, no. 1, pages 1–50, 2002. 119

[Komodakis 2007] N. Komodakis and G. Tziritas. *Approximate labeling via graph cuts based on linear programming*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 29, no. 8, pages 1436–1453, 2007. 16, 63, 101

[Komodakis 2008] N. Komodakis, G. Tziritas and N. Paragios. *Performance vs computational efficiency for optimizing single and dynamic MRFs: Setting the state of the art with primal-dual strategies*. Computer Vision and Image Understanding, vol. 112, no. 1, pages 14–29, 2008. 16, 63, 101, 121, 122, 128, 130

[Komodakis 2011] N. Komodakis. *Efficient training for pairwise or higher order crfs via dual decomposition*. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 1841–1848. IEEE, 2011. 125, 128

[Kwok 2003] J.T. Kwok and I.W. Tsang. *Learning with idealized kernels*. In MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-, volume 20, page 400, 2003. 85

[Lee 2008] C.H. Lee, S. Wang, A. Murtha, M. Brown and R. Greiner. *Segmenting brain tumors using pseudo–conditional random fields*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2008, pages 359–366, 2008. 121

[Lee 2009] D. Lee, M. Hofmann, F. Steinke, Y. Altun, N.D. Cahill and B. Schlkopf. *Learning the similarity measure for multi-modal 3d image registration*. In IEEE-CVPR, 2009. 25, 100

[Liao 2006] R. Liao, C. Guetter, C. Xu, Y. Sun, A. Khamene and F. Sauer. *Learning-based 2D/3D rigid registration using Jensen-Shannon divergence for image-guided surgery*. Medical Imaging and Augmented Reality, pages 228–235, 2006. 22

[Loeckx 2010] D. Loeckx, P. Slagmolen, F. Maes, D. Vandermeulen and P. Suetens. *Nonrigid image registration using conditional mutual information*. Medical Imaging, IEEE Transactions on, vol. 29, no. 1, pages 19–29, 2010. 22

[Lorenzi 2013] Marco Lorenzi, Nicholas Ayache, Giovanni B Frisoni and Xavier Pennec. *LCC-Demons: a robust and accurate diffeomorphic registration algorithm*. NeuroImage, 2013. 18

[Lowe 2004] D.G. Lowe. *Distinctive image features from scale-invariant keypoints*. International journal of computer vision, vol. 60, no. 2, pages 91–110, 2004. 34

[Lu 2011] C. Lu, S. Chelikani, X. Papademetris, J.P. Knisely, M.F. Milosevic, Z. Chen, D.A. Jaffray, L.H. Staib and J.S. Duncan. *An integrated approach to segmentation and nonrigid registration for application in image-guided pelvic radiotherapy*. Medical Image Analysis, vol. 15, no. 5, pages 772–785, 2011. 113

[Luenberger 2008] D.G. Luenberger and Y. Ye. Linear and nonlinear programming, volume 116. Springer Verlag, 2008. 14

[Maes 1997] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal and P. Suetens. *Multimodality image registration by maximization of mutual information*. Medical Imaging, IEEE Transactions on, vol. 16, no. 2, pages 187–198, 1997. 9, 14, 19

[Maintz 2001] JBA Maintz, PA Van den Elsen and MA Viergever. *3D multimodality medical image registration using morphological tools*. Image and vision computing, vol. 19, no. 1-2, pages 53–62, 2001. xiii, 23, 24

[Mallat 1999] S.G. Mallat. A wavelet tour of signal processing. Academic Pr, 1999. 34

[Manjunath 1996] B.S. Manjunath and W.Y. Ma. *Texture features for browsing and retrieval of image data*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 18, no. 8, pages 837–842, 1996. xiii, 35, 36

[Masotti 2006] M. Masotti. *A ranklet-based image representation for mass classification in digital mammograms*. Medical physics, vol. 33, page 3951, 2006. 31

[Masotti 2008] M. Masotti and R. Campanini. *Texture classification using invariant ranklet features*. Pattern Recognition Letters, vol. 29, no. 14, pages 1980–1986, 2008. 31

[Mellor 2004] M. Mellor and M. Brady. *Non-rigid multimodal image registration using local phase*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2004, pages 789–796, 2004. 34

[Michel 2010] F. Michel and N. Paragios. *Image transport regression using mixture of experts and discrete Markov random fields*. In Biomedical Imaging: From Nano to Macro, 2010 IEEE International Symposium on, pages 1229–1232. IEEE, 2010. 63, 65

[Michel 2011] F. Michel, M. Bronstein, A. Bronstein and N. Paragios. *Boosted metric learning for 3d multi-modal deformable registration*. In Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on, pages 1209–1214. IEEE, 2011. 93

[Mikolajczyk 2005] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir and L.V. Gool. *A comparison of affine region detectors*. International journal of computer vision, vol. 65, no. 1, pages 43–72, 2005. 29

[Neemuchwala 2002] H. Neemuchwala, A. Hero and P. Carson. *Image registration using entropic graph-matching criteria*. In Signals, Systems and Computers, 2002. Conference Record of the Thirty-Sixth Asilomar Conference on, volume 1, pages 134–138. IEEE, 2002. 21

[Ojala 2002] T. Ojala, M. Pietikainen and T. Maenpaa. *Multiresolution gray-scale and rotation invariant texture classification with local binary patterns*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 24, no. 7, pages 971–987, 2002. 30

[Onimaru 2003] R. Onimaru, H. Shirato, S. Shimizu, K. Kitamura, B. Xu, S. Fukumoto, T.C. Chang, K. Fujita, M. Oita, K. Miyasaka*et al.* *Tolerance of organs at risk in small-volume, hypofractionated, image-guided radiotherapy for primary and metastatic lung cancers*. International Journal of Radiation Oncology* Biology* Physics, vol. 56, no. 1, pages 126–135, 2003. xiii, 4

[Ou 2009] Y. Ou and C. Davatzikos. *DRAMMS: deformable registration via attribute matching and mutual-saliency weighting*. In Information Processing in Medical Imaging, pages 50–62. Springer, 2009. 16, 37

[Ou 2011] Y. Ou, A. Sotiras, N. Paragios and C. Davatzikos. *DRAMMS: Deformable registration via attribute matching and mutual-saliency weighting*. Medical Image Analysis, vol. 15, no. 4, pages 622–639, 2011. 37

[Pan 2006] X.B. Pan, M. Brady, R. Highnam and J. Declerck. *The use of multi-scale monogenic signal on structure orientation identification and segmentation*. Digital Mammography, pages 601–608, 2006. 34

[Parisot 2012a] S. Parisot, H. Duffau, S. Chemouny and N. Paragios. *Graph-based Detection, Segmentation & Characterization of Brain Tumors*. In CVPR-25th IEEE Conference on Computer Vision and Pattern Recognition 2012, 2012. 37

[Parisot 2012b] S. Parisot, H. Duffau, S. Chemouny and N. Paragios. *Joint Tumor Segmentation and Dense Deformable Registration of Brain MR Images*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2012, pages 651–658, 2012. 113

[Pennec 1999] X. Pennec, P. Cachier and N. Ayache. *Understanding the "demon's algorithm": 3D non-rigid registration by gradient descent*. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 1999, pages 597–605. Springer, 1999. 10

[Pescia 2008] D. Pescia, N. Paragios and S. Chemouny. *Automatic detection of liver tumors*. In Biomedical Imaging: From Nano to Macro, 2008. ISBI 2008. 5th IEEE International Symposium on, pages 672–675. IEEE, 2008. 30

[Pluim 2000] J. Pluim, J. Maintz and M. Viergever. *Image registration by maximization of combined mutual information and gradient information*. In Medical Image Computing and Computer-Assisted Intervention–MICCAI 2000, pages 103–129. Springer, 2000. 14, 20

[Pluim 2003] J.P.W. Pluim, J.B.A. Maintz and M.A. Viergever. *Mutual-information-based registration of medical images: a survey*. Medical Imaging, IEEE Transactions on, vol. 22, no. 8, pages 986–1004, 2003. 19

[Pluim 2004] J.P.W. Pluim, J.B.A. Maintz and M.A. Viergever. *f-Information measures in medical image registration*. Medical Imaging, IEEE Transactions on, vol. 23, no. 12, pages 1508–1516, 2004. 20

[Press 1986] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling*et al.* Numerical recipes, volume 547. Cambridge Univ Press, 1986. 13

[Randen 1999] T. Randen and J.H. Husoy. *Filtering for texture classification: A comparative study*. Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 21, no. 4, pages 291–310, 1999. 31

[Ren 2005] L. Ren, G. Shakhnarovich, J.K. Hodgins, H. Pfister and P. Viola. *Learning silhouette features for control of human motion*. ACM Transactions on Graphics (TOG), vol. 24, no. 4, pages 1303–1331, 2005. 86, 93

[Roche 1998] A. Roche, G. Malandain, X. Pennec and N. Ayache. *The correlation ratio as a new similarity measure for multimodal image registration*. Medical Image Computing and Computer-Assisted Interventation–MICCAI 1998, pages 1115–1124, 1998. 14, 20

[Roche 2001] A. Roche, X. Pennec, G. Malandain and N. Ayache. *Rigid registration of 3-D ultrasound with MR images: a new approach combining intensity and gradient information*. Medical Imaging, IEEE Transactions on, vol. 20, no. 10, pages 1038–1049, 2001. 9, 43

[Roweis 2000] S.T. Roweis and L.K. Saul. *Nonlinear dimensionality reduction by locally linear embedding*. Science, vol. 290, no. 5500, pages 2323–2326, 2000. 80

[Rueckert 1999] D. Rueckert, L.I. Sonoda, C. Hayes, D.L.G. Hill, M.O. Leach and D.J. Hawkes. *Nonrigid registration using free-form deformations: application to breast MR images*. Medical Imaging, IEEE Transactions on, vol. 18, no. 8, pages 712–721, 1999. 9, 11, 12, 13

[Rueckert 2000] D. Rueckert, MJ Clarkson, DLG Hill and DJ Hawkes. *Non-rigid registration using higher-order mutual information*. In Proceedings of SPIE, volume 3979, page 438, 2000. 21, 22

[Rueckert 2006] D. Rueckert, P. Aljabar, R. Heckemann, J. Hajnal and A. Hammers. *Diffeomorphic registration using B-splines*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2006, pages 702–709, 2006. 12

[Russakoff 2004] D. Russakoff, C. Tomasi, T. Rohlfing and C. Maurer. *Image similarity using mutual information of regions*. Computer Vision-ECCV 2004, pages 596–607, 2004. 22

[Schapire 1999a] R.E. Schapire. *A brief introduction to boosting*. In International Joint Conference on Artificial Intelligence, volume 16, pages 1401–1406. LAWRENCE ERLBAUM ASSOCIATES LTD, 1999. 87

[Schapire 1999b] R.E. Schapire and Y. Singer. *Improved boosting algorithms using confidence-rated predictions*. Machine learning, vol. 37, no. 3, pages 297–336, 1999. 118

[Schmid 2000] C. Schmid, R. Mohr and C. Bauckhage. *Evaluation of interest point detectors*. International Journal of computer vision, vol. 37, no. 2, pages 151–172, 2000. 29

[Sederberg 1986] T.W. Sederberg and S.R. Parry. *Free-form deformation of solid geometric models*. ACM Siggraph Computer Graphics, vol. 20, no. 4, pages 151–160, 1986. 11

[Setia 2006] L. Setia, A. Teynor, A. Halawani and H. Burkhardt. *Image classification using cluster cooccurrence matrices of local relational features*. In Proceedings of the 8th ACM international workshop on Multimedia information retrieval, pages 173–182. ACM, 2006. 30

[Shakhnarovich 2005] G. Shakhnarovich. *Learning task-specific similarity*. PhD thesis, Massachusetts Institute of Technology, 2005. 86, 93

[Shalev-Shwartz 2004] S. Shalev-Shwartz, Y. Singer and A.Y. Ng. *Online and batch learning of pseudo-metrics*. In Proceedings of the twenty-first international conference on Machine learning, page 94. ACM, 2004. 90

[Shekhovtsov 2008] A. Shekhovtsov, I. Kovtun and V. Hlavác. *Efficient MRF deformation model for non-rigid image matching*. Computer Vision and Image Understanding, vol. 112, no. 1, pages 91–99, 2008. 16

[Shen 2002] D. Shen and C. Davatzikos. *HAMMER: hierarchical attribute matching mechanism for elastic registration*. Medical Imaging, IEEE Transactions on, vol. 21, no. 11, pages 1421–1439, 2002. 11, 18

[Shen 2008] C. Shen, A. Welsh and L. Wang. *PSDBoost: Matrix-generation linear programming for positive semidefinite matrices learning*. Proc. Adv. Neural Inf. Process. Syst, pages 1473–1480, 2008. 89

[Shen 2009] C. Shen, J. Kim, L. Wang and A. Hengel. *Positive semidefinite metric learning with boosting*. Advances in neural information processing systems, 2009. 89

[Shen 2012] C. Shen, J. Kim, L. Wang and A. Hengel. *Positive Semidefinite Metric Learning Using Boosting-like Algorithms*. Journal of Machine Learning Research, Accepted in 2012. 89, 99

[Shental 2006] N. Shental, T. Hertz, D. Weinshall and M. Pavel. *Adjustment learning and relevant component analysis*. ECCV 2002, pages 181–185, 2006. xv, 82, 83

[Smeraldi 2002] F. Smeraldi. *Ranklets: orientation selective non-parametric features applied to face detection*. In Pattern Recognition, 2002. Proceedings. 16th International Conference on, volume 3, pages 379–382. IEEE, 2002. xiii, 30, 31

[Smeraldi 2003] F. Smeraldi. *Ranklets: a complete family of multiscale, orientation selective rank features*. Research Report RR0309-01, Dept. of Computer Science, Queen Mary, Univ. of London, 2003. 30

[Smola 1998] A.J. Smola and B. Schölkopf. Learning with kernels. Citeseer, 1998. 76

[Sotiras 2010] A. Sotiras, Y. Ou, B. Glocker, C. Davatzikos and N. Paragios. *Simultaneous geometric-iconic registration*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2010, pages 676–683, 2010. 37

[Sotiras 2011] A. Sotiras. *Discrete Image Registration: a Hybrid Paradigm*. PhD thesis, ECP, 2011. 10, 13, 18

[Studholme 1999] C. Studholme, D.L.G. Hill, D.J. Hawkes*et al.* *An overlap invariant entropy measure of 3D medical image alignment*. Pattern recognition, vol. 32, no. 1, pages 71–86, 1999. 20

[Studholme 2000] C. Studholme, R.T. Constable and J.S. Duncan. *Accurate alignment of functional EPI data to anatomical MRI using a physics-based distortion model*. Medical Imaging, IEEE Transactions on, vol. 19, no. 11, pages 1115–1127, 2000. 20

[Studholme 2001] C. Studholme, E. Novotny, IG Zubal and JS Duncan. *Estimating tissue deformation between functional images induced by intracranial electrode implantation using anatomical MRI*. NeuroImage, vol. 13, no. 4, pages 561–576, 2001. 20

[Tenenbaum 2000] J.B. Tenenbaum, V. De Silva and J.C. Langford. *A global geometric framework for nonlinear dimensionality reduction*. Science, vol. 290, no. 5500, pages 2319–2323, 2000. xv, 79

[Thirion 1998] J.P. Thirion. *Image matching as a diffusion process: an analogy with Maxwell's demons*. Medical image analysis, vol. 2, no. 3, pages 243–260, 1998. 10

[Torresani 2007] L. Torresani and K. Lee. *Large margin component analysis*. Advances in neural information processing systems, vol. 19, page 1385, 2007. 89, 99

[Tsang 2005] I.W. Tsang, P.M. Cheung and J.T. Kwok. *Kernel relevant component analysis for distance metric learning*. In Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on, volume 2, pages 954–959. Ieee, 2005. 83

[Tuceryan 1993] M. Tuceryan and A.K. Jain. *Texture analysis*. Handbook of pattern recognition and computer vision, vol. 276, 1993. 29

[Vercauteren 2007a] T. Vercauteren, X. Pennec, E. Malis, A. Perchant and N. Ayache. *Insight into efficient image registration techniques and the demons algorithm*. In Information Processing in Medical Imaging, pages 495–506. Springer, 2007. 14

[Vercauteren 2007b] T. Vercauteren, X. Pennec, A. Perchant and N. Ayache. *Non-parametric diffeomorphic image registration with the demons algorithm*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2007, pages 319–326, 2007. 11

[Viola 1997] P. Viola and W.M. Wells III. *Alignment by maximization of mutual information*. International journal of computer vision, vol. 24, no. 2, pages 137–154, 1997. 19

[Wachinger 2011] C. Wachinger and N. Navab. *Entropy and Laplacian images: Structural representations for multi-modal registration*. Medical Image Analysis, 2011. xiii, 24, 25, 26

[Wang 2008] C. Wang, L. Zhang and H.J. Zhang. *Learning to reduce the semantic gap in web image retrieval and annotation*. In Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, pages 355–362. ACM, 2008. 85

[Wang 2010] C. Wang, O. Teboul, F. Michel, S. Essafi and N. Paragios. *3D knowledge-based segmentation using pose-invariant higher-order graphs*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2010, pages 189–196, 2010. 37

[Wein 2008] W. Wein, S. Brunke, A. Khamene, M.R. Callstrom and N. Navab. *Automatic CT-ultrasound registration for diagnostic imaging and image-guided intervention*. Medical Image Analysis, vol. 12, no. 5, pages 577–585, 2008. xiv, 43, 44

[Weinberger 2006] K.Q. Weinberger, J. Blitzer and L.K. Saul. *Distance metric learning for large margin nearest neighbor classification*. In In NIPS. Citeseer, 2006. 88

[Weinberger 2009] K.Q. Weinberger and L.K. Saul. *Distance metric learning for large margin nearest neighbor classification*. The Journal of Machine Learning Research, vol. 10, pages 207–244, 2009. 88

[Wells III 1996] W.M. Wells III, P. Viola, H. Atsumi, S. Nakajima and R. Kikinis. *Multi-modal volume registration by maximization of mutual information*. Medical image analysis, vol. 1, no. 1, pages 35–51, 1996. 19

[Xiang 2011] B. Xiang, C. Wang, J.F. Deux, A. Rahmouni and N. Paragios. *Tagged cardiac MR image segmentation using boundary & regional-support and graph-based deformable priors*. In Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on, pages 1706–1711. IEEE, 2011. 37

[Xiang 2012] B. Xiang, C. Wang, J.F. Deux, A. Rahmouni and N. Paragios. *3D cardiac segmentation with pose-invariant higer-order MRFs*. 2012. 37

[Xiaohua 2004] C. Xiaohua, M. Brady and D. Rueckert. *Simultaneous segmentation and registration for medical image*. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2004, pages 663–670, 2004. 113

[Xing 2002] E.P. Xing, A.Y. Ng, M.I. Jordan and S. Russell. *Distance metric learning, with application to clustering with side-information*. Advances in neural information processing systems, vol. 15, pages 505–512, 2002. 84, 91

[Xu 1995] L. Xu, M.I. Jordan and G.E. Hinton. *An alternative model for mixtures of experts*. Advances in neural information processing systems, pages 633–640, 1995. 54

[Xue 2004] Z. Xue, D. Shen and C. Davatzikos. *Determining correspondence in 3-D MR brain images using attribute vectors as morphological signatures of voxels*. Medical Imaging, IEEE Transactions on, vol. 23, no. 10, pages 1276–1291, 2004. 18, 34

[Yang 2006] L. Yang and R. Jin. *Distance metric learning: A comprehensive survey*. Michigan State Universiy, pages 1–51, 2006. 78

[Yeo 2009] B.T.T. Yeo, T. Vercauteren, P. Fillard, J.M. Peyrat, X. Pennec, P. Golland, N. Ayache and O. Clatz. *DT-REFinD: Diffusion tensor registration with exact finite-strain differential*. Medical Imaging, IEEE Transactions on, vol. 28, no. 12, pages 1914–1928, 2009. 65, 102

[Yuan 2007] J. Yuan, Y. Wu and M. Yang. *Discovery of collocation patterns: from visual words to visual phrases*. In Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, pages 1–8. Ieee, 2007. 85

[Zhan 2003] Y. Zhan and D. Shen. *Automated segmentation of 3D US prostate images using statistical texture-based matching method*. Medical Image Computing and Computer-Assisted Intervention-MICCAI 2003, pages 688–696, 2003. xiii, 37

[Zhou 2008] H. Zhou, R. Wang and C. Wang. *A novel extended local-binary-pattern operator for texture analysis*. Information Sciences, vol. 178, no. 22, pages 4314–4325, 2008. 30

[Zhuang 2011] X. Zhuang, S. Arridge, D.J. Hawkes and S. Ourselin. *A nonrigid registration framework using spatially encoded mutual information and free-form deformations*. IEEE Transactions on Medical Imaging, vol. 30, no. 10, page 1819, 2011. 22

[Zikic 2010] D. Zikic, B. Glocker, O. Kutter, M. Groher, N. Komodakis, A. Kamen, N. Paragios and N. Navab. *Linear intensity-based image registration by Markov random fields and discrete optimization*. Medical Image Analysis, vol. 14, no. 4, pages 550–562, 2010. 16