Abstract

Information Theoretic Integrated Segmentation and

Registration of Dual 2D Portal Images and 3D CT Images

Ravi Bansal

1999

This thesis develops an information theoretic registration framework where the segmentation and registration of dual anterior–posterior and left lateral portal images to a treatment planning three–dimensional computed tomography (CT) image is carried out simultaneously and iteratively. The proposed registration framework is termed the *minimax* entropy registration framework as it has two steps, the *max* step and the *min* step. Appropriate entropies are evaluated in each step in order to segment the portal images (the *max* step) and to estimate the registration parameters (the *min* step). The registration framework is based on the intuition that if some structure can be segmented in the portal image, the segmented structure, in addition to the gray–scale pixel intensity information, can be used to better estimate the registration parameters. On the other hand, given an estimate of the registration parameters, information from the high resolution 3D CT image dataset can be used to guide segmentation of the portal images. Performance analysis and comparisons to other registration methods demonstrates the robustness and accuracy of the proposed registration framework.

To further improve the estimated segmentation of the portal images and the accuracy of the estimated registration parameters, correlation among the image pixel intensities is modeled using a one–dimensional Markov random process. *Line processes* are incorporated in the Markov random process model which estimate the edges between the segmented regions. As a future research direction, we propose to incorporate the estimated edges in the *min* step to further improve the registration. The proposed framework is independent of the image dataset and hence, in general, can be straightforwardly extended to register any low resolution, low contrast image to a high resolution, high contrast image.

Information Theoretic Integrated Segmentation and Registration of Dual 2D Portal Images and 3D CT Images

A Dissertation

Presented to the Faculty of the Graduate School

of

Yale University

in Candidacy for the Degree of

Doctor of Philosophy

by

Ravi Bansal

Dissertation Director: James Scott Duncan

December 1999

© Copyright by Ravi Bansal 2000

All Rights Reserved

.

To:

My brothers Sanjay Bansal and Sandeep Bansal

and my wife Nilakshi Bansal

Acknowledgments

To begin with, I am grateful to my advisor, James Duncan, for the invaluable guidance and encouragement throughout the thesis project. I am especially grateful to Lawrence Staib without whose guidance and infinite discussions I would have been lost. I am thankful to Ravinder Nath and Roman Kuc for serving on my committee. I am thankful to Zhe Chen for all the discussions we had and Jonathan Knisely for the guidance in the project, especially from the radiotherapist point of view and making real patient data available. I am thankful to Vladimir Neklesa for his herculean efforts in making sure that I finally did get the digitized data. I must also thank Anand Rangarajan for his insights into the problem, especially the mathematics. Thanks to Charles Pelizzari for serving as my external reader.

I must thank all the current members and the alumni of the Image Processing and Analysis Group at Yale: Sudhakar Chelikani, Haili Chui, Melissa Koudelka, Francois Meyer, Xenios Papademetris, Suguna Pappu, K. Purosottam, Pengcheng Shi, N. Rajeevan, Hong Shi, Rik Stokking, Colin Studholme, Hemant Tagare, Yongmei Wang, Lawrence Win, Xiaolan Zeng and George Zubal. I will like to thank Reshma Munbodh for correcting so many grammatical errors in the thesis. I am particularly thankful to Carolyn Meloling for the numerous times she helped me.

I would like to thank my parents, Babu Lal Bansal and Kalawati Bansal, and brothers, Sanjay Bansal and Sandeep Bansal, for their support and encouragement. I am especially thankful to my wife Nilakshi Bansal for her love and understanding and for making the whole experience so pleasant.

List of Figures

3.1 Attenuation of Photons 32

3.2 Setup for 3D CT and DRR 37

3.3 Mixture of two Gaussian 41

3.4 MRF Neighborhood Systems 49

5.1 MIME Algorithm Flow Chart 85

5.2 Treatment Planning CT and Real Portal Image 97

5.3 Single Portal Image Setup 98

5.4 Dual AP and LL Portal Image Setup 98

5.5 2D Analysis Using Blurred Portal Image 100

5.6 3D Analysis Using Real AP Portal Image 101

5.7 Comparing Different Segmentation Methods 102

5.8 Error in Estimated Parameters Using Single AP Portal Image 104

5.9 3D Analysis Using Dual Simulated Portal Images 105

5.10 3D Analysis Using Dual Blurred Portal Images 107

5.11 3D Analysis Using Dual Noisy Portal Images 108

5.12 Error in Estimated 3D Parameters for Increasing Noise and Blur 109

5.13 Error in Estimated 3D Parameters for Increasing Noise and Blur with Femurs Edited Out 112
5.14 Performance on Real Dual Portal Images 113

6.1 Simulated Portal Image with Different Noise and Blur 123
6.2 Surface Plots of Energy Fuctions for 2D Analysis Using Blurred Portal Image 124
6.3 Surface Plots of Energy Fuctions for 2D Analysis Using Noisy Portal Image 125
6.4 Simulated Portal Images and Estimated Segmentation 126
6.5 Graphs of Estimated 2D Parameters for Increasing Noise and Blur in Simulated Portal Image 127
6.6 Surface Plots of Energy Function Using Estimated Segmentation 128
6.7 Parameters Estimated by the Ridge–Based Method with Increasing Noise and Blur 130
6.8 Parameters Estimated by MI and MIME with Increasing Noise and Blur . 131
6.9 Estimated Segmentation and Ridged Enhanced Image of Noisy Portal Image 132
6.10 Surface Plots of Energy Function of GL, MI and MIME algorithms 132
6.11 AP and LL DRRs and Corresponding Ridge–Enhanced Images 135
6.12 Simulated Noisy AP and LL Portal Images and Corresponding Ridge–Enhanced Images 136
6.13 Simulated Blurred AP and LL Portal Images and Corresponding Ridge–Enhanced Images 137
6.14 Parameters Estimated by GL Algorithm under Increasing Noise and Blur 138
6.15 Estimated Segmentation of Noisy AL and LL Portal Images 139
6.16 Parameters Estimated by GL, MI, MIME Algorithms under Increasing Noise and Blur 140

6.17 Surface Plots of Energy Functions of GL, MI, MIME Algorithms for Noisy AP and LL Portal Images 141

8.1 2D Analysis using MI and MIME for Noisy Portal Image Using True Distributions 168

8.2 2D Analysis using MI and MIME for Noisier Portal Image 169

8.3 2D Analysis using MI and MIME for Blurred Portal Image 170

8.4 Surface Plots of MI and MIME Energy Functions for Noisy Portal Image 171

8.5 Surface Plots of MI and MIME Energy Functions for Blurred Portal Image 172

8.6 2D Analysis using Noisy Portal Image and Estimated Distributions 173

8.7 2D Analysis using Blurred Portal Image and Estimated Distributions . . . 174

8.8 Surface Plots of Energy Functions Using Estimated Distributions of Noisy Portal Image 176

8.9 Surface Plots of Energy Functions Using Estimated Distributions of Blurred Portal Image 177

8.10 Error in Estimated Parameters with Increasing Noise and Blur in Portal Image 178

8.11 3D Analysis Using Dual Noisy Portal Images and Estimated Distributions 180

8.12 3D Analysis Using Dual Blurred Portal Images and Estimated Distributions 182

8.13 3D Analysis Using Dual Noisy Portal Images and Estimated Distributions 183

8.14 3D Analysis Using Dual Blurred Portal Images and Estimated Distributions 184

8.15 Surface Plots of Energy Functions for Noisy Dual Portal Images 185

8.16 Surface Plots of Energy Functions for Blurred Dual Portal Images 186

8.17 Surface Plots of Energy Functions for Noisy Dual Portal Images 187

8.18 Surface Plots of Energy Functions for Blurred Dual Portal Images 188

8.19 Error in Estimated 3D Parameters under Increasing Noise and Blur . . . 189

8.20 Error in Estimated 3D Parameters under Increasing Noise and Blur . . . 190

8.21 Performance of MIME algorithm Using Real Dual Portal Images 192

8.22 Performance of MI algorithm Using Real Dual Portal Images 193

8.23 195

8.24 197

A.1 Graph Depicting Relation Between MI and Correlation Coefficient 215

Contents

Dedication **iii**

Acknowledgements **iv**

List of Figures **v**

1 Introduction **1**

1.1 Introduction to the Problem 1

1.1.1 Patient Setup Verification in EBRT 3

1.2 Main Contributions 4

1.3 Overview of the Thesis 5

2 Related Work **8**

2.1 Two–Dimensional Analysis 9

2.1.1 Feature Based 9

2.1.2 Dense Field–Based 12

2.2 Three–Dimensional Analysis 14

2.2.1 Feature–Based 15

2.2.2 Dense Field–Based 17

2.2.3 Interactive Methods 18
2.3 General Registration Methods 19
2.3.1 Feature–Based or Sparse–Field Methods 19
2.3.2 Dense–Field Methods 23
2.3.3 Non–Rigid Image Registration 27
2.4 Summary 28

3 Image formation models and background 30

3.1 Introduction 30
3.2 Formation of Images 31
3.2.1 Computer Tomography Images 31
3.2.2 Portal Images 34
3.2.3 Projection Model 37
3.3 The EM Algorithm 38
3.3.1 Problem Definition 39
3.3.2 Intuition 40
3.3.3 General Statement 43
3.3.4 Mixture Model 44
3.3.5 Alternate Views 44
3.3.6 Numerical Advantages 46
3.4 Markov Random Fields 46
3.4.1 Ising Model 50
3.4.2 Mean Field Approximations 51
3.5 Density Estimation 53
3.5.1 Parametric Methods 53

3.5.2 Non–Parametric Methods 54
3.6 Mutual Information 57
3.6.1 Entropy 57
3.6.2 Relation to Correlation Function 59
3.6.3 Empirical Estimation of Entropy 62
3.6.4 Maximum Entropy Principle 63
3.7 Summary 67

4 Segmentation–Integrated Registration 68

4.1 Introduction 68
4.2 Problem Definition 71
4.3 Maximum A–Posteriori Formulation 73
4.3.1 Drawbacks 77
4.4 Maximum Likelihood Formulation 78
4.4.1 EM Formulation 79
4.4.2 Observations 80
4.5 Summary 82

5 Minimax Entropy Algorithm 83

5.1 Introduction 83
5.2 Minimax Entropy Formulation 83
5.2.1 Formulating the MAX step 84
5.2.2 Formulating the MIN step 89
5.2.3 Stochastic Gradient Descent Based Optimization 91
5.2.4 Estimation of Covariance Matrix 93
5.3 Coordinate Descent Interpretation 94

5.4 Utilizing Dual Portal Images 95
5.5 Extension to 2D/2D Image Registration 96
5.6 Results 97
5.6.1 Registration of 2D/2D Images 102
5.6.2 2D/3D Registration 103
5.6.3 Performance with Femur Edited Out 111
5.6.4 Performance on Real Data 112
5.7 Summary 114

6 Performance Analysis and Comparisons 116

6.1 Introduction 116
6.2 Performance Evaluation Strategy 117
6.2.1 Ridge–Based Method 117
6.2.2 Mutual Information 120
6.3 Comparisons for 2D/2D Image Registration 121
6.4 Comparisons for 2D/3D Image Registration 129
6.4.1 3D Analysis Using Single Portal Image 133
6.4.2 3D Analysis Using Dual Portal Images 134
6.5 Summary 142

7 Incorporating Pixel Intensity Correlations 144

7.1 Introduction 144
7.1.1 Outline of the Approach 145
7.2 Mathematical Formulation of the MAX step 147
7.2.1 Integrating out Edge Information 150
7.2.2 Maximum Entropy Estimation 152

7.2.3 Estimating Correlations 160
7.3 Formulation of the MIN step 160
7.4 Summarizing the Algorithm 162
7.5 Convergence of the Minimax Algorithm 163
7.6 Reduction of Formulation Under Pixel Independence 164
7.7 Summary 165

8 Performance Analysis of the Algorithm Incorporating Pixel Correlation 166

8.1 Introduction 166
8.2 2D Simulator to 2D Portal Image Registration 167
8.2.1 Using True Segmentation of the Portal Images 167
8.2.2 Using Estimated Segmentation of the Portal Images 175
8.2.3 Performance Under Increasing Noise and Blur 178
8.3 2D Portal to 3DCT Registration 179
8.3.1 3D Analysis Using Dual Simulated Portal Images 179
8.3.2 Performance Under Varying Noise and Blur 188
8.4 Performance on Real Patient Data Set 191
8.4.1 Performance Using Phantom Data 194
8.5 Summary 199

9 Future Work 200

9.1 Introduction 200
9.2 Formulation Incorporating Edge Information 201
9.2.1 Incorporation of Whole Boundary Information 204
9.2.2 Using higher order MRFs 205
9.3 Organ Motion and Using Ultrasound Data 205

9.4 Summary 206

10 Summary **207**

A Relation between Correlation Coefficient and Mutual Information **210**

A.0.1 **Evaluation of $\frac{\partial}{\partial T}H(y)$:** 212

A.0.2 **Evaluation of $\frac{\partial}{\partial T}H(x,y)$:** 212

A.1 Extensions to Mixture of Gaussian Density Functions 214

B Derivation of the EM steps **216**

B.0.1 Development of the E–step 217

B.0.2 Development of the M–step 219

C Developing the Minimax Entropy Registration Framework **221**

C.1 Why Evaluate Expected Values 221

C.2 Relation Between the EM and the Min Step 224

C.3 Deriving the Min Step from Joint Conditional Entropy 228

C.4 Gradient Evaluation 230

C.5 Coordinate Descent Interpretation of Minimax 234

C.6 Convergence Analysis of Stochastic Approximation Algorithms 240

D Incorporating Correlation in Registration Framework **243**

D.1 Maximum Entropy Estimation 243

D.1.1 Evaluation of the term $\frac{\partial}{\partial \varepsilon_{vi}}F_I(\beta)$ 244

D.1.2 Evaluation of the term $\frac{\partial}{\partial \varepsilon_{vi}} < E_I(\mathbf{Z}) >$ 244

D.1.3 Evaluating the term $\sum_j \sum_{\alpha \in A} \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} z_{\alpha j+1} >$ 248

D.1.4 Evaluating the term $\frac{1}{\beta}\sum_j \frac{\partial}{\partial \varepsilon_{vi}} < \ln l_j >$ 250

D.2 Evaluating Correlation Between Pixels 254

D.3 Approximated Expected Value 259

D.4 Conditional Probability of the ith edge 260

D.5 Evaluation of various entropy terms 264

D.5.1 Evaluation of the term $H(X, Y|M)$ 264

D.5.2 Evaluation of the term $H(Y)$ 267

D.5.3 Evaluation of the term $H(\mathcal{E}, G_e, T)$ 268

D.6 Parzen Window Stochastic Estimates of the Entropies 272

D.6.1 Evaluating $H_\alpha(x, y)$ 272

D.6.2 Evaluating $H(y)$ 275

D.6.3 Evaluating $H_1(s')$ 277

D.6.4 Evaluating $H_{\alpha\beta}(x_2, y_2, x_1, y_1)$ 281

Chapter 1

Introduction

1.1 Introduction to the Problem

Medical images using different modalities, for example computed tomography (CT), single photon emission computed tomography (SPECT), positron emission tomography (PET), magnetic resonance imaging (MRI), ultrasound and *functional* MRI (fMRI) are usually obtained for clinical diagnosis, planning and evaluation of a therapy. Images from different modalities often depict different aspects of the same anatomical region. For example, the CT image dataset is useful for highlighting the bony anatomy of a region but it captures very little soft tissue information. On the other hand, the soft tissue information is best captured using MRI.

Integration or fusion of the information obtained from different modalities can greatly assist the physician in diagnosis, planning and evaluation of a treatment. For example, in external beam radiotherapy (EBRT) of cancer, a CT scan is needed for dose distribution calculations, while MRI is best for contouring the target lesion. Similarly, the integration of functional information from PET, SPECT and fMRI and anatomical

information from MRI can be used to improve the reconstruction of PET or SPECT and also to localize the anatomy. While fusion or registration of information from different modalities can greatly assist clinicians, the presence of different information in the images makes automated registration a difficult task. Recently, mutual information based registration method [117, 28] has been successfully used for the registration of multi–modality images.

While registering images from different modalities, often one of the images to be registered is of low resolution and contrast (for example, PET, SPECT, fMRI) and the other image is of high resolution and contrast (for example, MRI, CT). A knowledge of the segmentation of the low resolution image can greatly assist in robust and accurate registration of the images. On the other hand, if a low contrast image is accurately registered to a high resolution image, information from the high resolution image can be used to help segment the low resolution image. However, a segmentation of the low resolution image is almost never available, nor do we have an accurate registration of the images.

The segmentation of a low resolution, low contrast image and its registration to a high resolution, high contrast image are thus dual problems in the sense that a segmentation of the low resolution image can be used to estimate accurate registration of the images and an accurate registration can be used to estimate segmentation of the low resolution image. Thus, while registering a low resolution image to a high resolution image, there is a need for a registration framework where the segmentation of an image and its registration is carried out simultaneously and iteratively. The intuition behind such a framework is that a rough estimate of the registration can help estimate a segmentation of the low resolution image which in–turn can be used to better register the images and

so on.

1.1.1 Patient Setup Verification in EBRT

For this thesis, we are interested in the patient setup verification during external beam radiotherapy (EBRT) of prostate cancer. Many treatment centers around the country are moving towards providing full 3D conformal EBRT [109, 99] with the overall goal of reducing the irradiated volume to a minimum while still covering the target volume. Such conformal schemes may reduce treatment–related morbidity while providing the same tumor cure rate. However, in general, as the irradiated volume becomes smaller and more complex in shape and as the doses are escalated, the sensitivity of the treatment to uncertainties, especially those due to patient positioning and organ motion, increases. Also, currently there is no robust and effective methodology to automatically co–register information, primarily image–based, derived from the different setup verification sources.

Treatment planning 3D CT image data sets, obtained using diagnostic energy X–rays (40 KeV to 100 KeV), are used for segmenting the prostate, treatment planning and dosimetric calculations. High quality two–dimensional (2D) simulator images, in the *anterior–posterior* (AP) and the *left–lateral* (LL) directions are usually obtained, using diagnostic energy X–rays. When the patient is moved to the treatment room, low quality portal images, again in the AP and LL directions, are obtained using treatment energy X–rays (6 MeV–20 MeV) before the full fraction of the dose is given to the patient. The registration of the portal images to the simulator or the 3D CT image dataset is used to verify the position of the patient on the treatment table.

In many radiotherapy centers, the state–of–the–art method of registering portal images to simulator images involves manual registration of the images by a clinician. In addition to the errors introduced by the human operator, the presence of out–of–plane

rotations in the portal images further increases the registration errors. To account for out–of–plane rotations in the images, it is necessary to register the portal images to the 3D CT image. As the 3D CT data set is obtained using diagnostic energy X–rays, the CT images are usually of high contrast and high spatial resolution. On the other hand, the portal images, obtained using treatment energy X–rays, are of low contrast and low spatial resolution. Thus, automated registration of the portal images to the 3D CT dataset becomes a very difficult problem.

1.2 Main Contributions

This dissertation develops an information–theoretic integrated registration framework where the segmentation of the portal images and their registration to the 3D CT image dataset is carried out simultaneously and iteratively. This framework leads to an accurate and robust registration algorithm with sub–pixel accuracy in the estimated parameters. The major contributions of the dissertation can be listed to be

- The problem of estimating a segmentation of the portal image and the registration parameters is posed for the first time as a maximum a–posteriori (MAP) problem where both the segmentation labels and the registration parameters are being explicitly estimated.

- The problem is then setup in the expectation–maximization (EM) [32] framework where the segmentation labels are treated as hidden variables.

- Noting some restrictions of the EM algorithm for the registration of images from different modalities, we setup a unique information–theoretic minimax entropy registration framework for the estimation of the registration and the distribution on

the segmentation labels.

- A coordinate descent interpretation of the minimax entropy registration framework is presented which provides an intuition about the convergence of the algorithm to a local optimum.

- To better segment the portal images, the portal images are modeled as 1D Markov random fields (MRFs). *Line processes* [43] are introduced into the MRF model to improve the segmentation of the portal image.

- As future work, this dissertation develops a framework for the incorporation of edges into the registration framework for further improving the accuracy of the estimated registration parameters.

1.3 Overview of the Thesis

The next chapter, Chapter 2, reviews relevant background literature on various methods used for image registration, especially the methods used for multi–modal image registration and the methods used for registering dual portal images to 3D CT datasets.

Chapter 3 discusses, in brief, the mathematical details of image formation models, the MRF based image segmentation method, the EM algorithm, different methods used for estimating probability density functions from a given sample of data and various relevant concepts from information theory.

Chapter 4 first formulates the problem of estimating the portal image segmentation and its registration to 3D CT dataset as a MAP problem where both the segmentation labels and the transformation parameters are being estimated explicitly. Due to some of the problems noted for such an approach, the problem is then formulated in an ML

framework where only the transformation parameters are estimated explicitly, with only the probability distribution on the segmentation labels being evaluated in each step. The ML transformation parameters are proposed to be estimated by using the EM algorithm.

Chapter 5 notes some of the restrictions of the ML based formulation of the problem and proposes the *minimax* entropy based registration approach where appropriate entropies are being manipulated to estimate both the registration parameters and the distribution of the segmentation labels. The performance of the proposed framework is then evaluated using single and dual simulated portal images. The framework is also extended to register 2D simulator images to 2D portal images. Finally, the performance of the proposed framework is evaluated using real dual portal images.

Chapter 6 evaluates and compares the performance of the proposed minimax entropy framework with the performance of the mutual information based registration method [117, 28] and the ridge–based [44] registration framework.

Chapter 7 incorporates the correlation among the neighboring pixel intensities in the portal image by using 1D MRF model. Line processes are incorporated to better segment the portal images. As the portal image is to be segmented into two classes and also to reduce the computational load, a simplified 1D Ising chain model is used to formulate pixel correlations. The joint conditional entropy of the 1D Markov random processes is then minimized to estimate the transformation parameters.

Chapter 8 evaluates and compares the performance of the *minimax entropy* (MIME) registration framework with the performance of the mutual information based registration method. The algorithm is again extended to register 2D simulator images to 2D portal images. First, the results are presented for registering 2D images. Then the performance of the two algorithms is compared using simulated dual portal images. Finally,

the performance of the proposed framework is evaluated on real dual portal images.

Chapter 9 points out some of the future directions of the thesis and also points out that for the purpose of external beam radiotherapy (EBRT), localization of the soft tissue, the prostate, is of critical importance. The registration of portal images to the 3D CT dataset is driven, in the proposed framework, mostly by the bony anatomy. However, due to the rectal filling, the prostate can move by as much as 2 cm with respect to the bony anatomy. Thus, for conformal EBRT of the prostate cancer, it is necessary to account for the movement of the prostate.

Finally, chapter 10 summarizes the contributions of the dissertation.

Chapter 2

Related Work

Many radiotherapy centers are planning to move towards providing full conformal radiotherapy for prostate cancer treatment. This technique aims at delivering a specific amount of dose, with high precision, to the cancerous tissue, avoiding exposure to the radio-sensitive healthy tissue that may exist near the tumor. However, to be effective, such a strategy requires accurate positioning of the patient during the external beam radiotherapy. A lack of treatment precision can lead to adverse outcomes [123, 36]. Traditionally, alignment of the patient is often achieved by using crosses and lines drawn on the skin and aligning them with lasers in order to determine the irradiation coordinate system. However, the accuracy of this method is susceptible to numerous sources of errors. To ensure complete immobilization of the patient, stereo-tactic frames have been proposed [47, 98]. In addition to being invasive, the fixation cannot be reproduced from day to day within the fractionated radiotherapy procedure [99] and also patient fixation is difficult for parts of the body other than head and neck.

A more convenient, and possibly more accurate, approach is to use multi-modal images, such as, a treatment planning 3D CT dataset and high energy intra-operative 2D

portal images, to determine patient setup errors using software registration techniques. Various semi–automated and automated methods have been proposed in the literature, which assess patient setup errors with varying degrees of accuracy. These techniques can be broadly classified into two categories. First, techniques which utilize two–dimensional image information: the simulator and the portal images. These methods can estimate only the in–plane movements of the patient. However, in the presence of out–of–plane movement, the accuracy of the estimated patient setup error, using these methods, can be compromised. Second, techniques that evaluate the patient setup in the three dimensions, utilizing 3D CT dataset and the 2D portal images. These techniques are essential to accurately estimate out–of–plane rotations and thus for the introduction of complex conformal treatment plans.

2.1 Two–Dimensional Analysis

In this section we describe techniques, proposed in the literature, that evaluate the patient setup in two dimensions utilizing the simulator images and the portal images. These techniques are further classified as the *feature based* (or the sparse field) methods and the *dense field* methods depending upon the information utilized to bring the images into alignment.

2.1.1 Feature Based

Feature based (or sparse field) methods aim at extracting homologous features from the images to be registered. The extracted features are then utilized to estimate the alignment between the two images. These methods tend to be fast. However, the accuracy of these methods is limited by the accuracy of the preprocessing step, that is, the extraction of

the features. Since most feature extraction methods require taking derivatives of the images, the accuracy of these methods is limited by the accuracy of the feature detection and localization step. These methods also assume that corresponding features have been extracted from the images to be registered, an assumption especially problematic with multi–modal images.

2.1.1.1 Core–Based

Core, or the multi-scale medial axis, based registration of the simulator and the portal image has been proposed in [40, 87, 41]. Cores aim at capturing the middle and width properties of an object at multiple spatial scales. They are skeleton like structures that run down center of the objects, with an additional dimension, scale, which is proportional to the width of the object at each spatial position on the core. A core point represents the spatial position and the size of the medialness kernel that is a best fit locally, both in position and scale. Due to the representation of the cores on different scales, cores for large–scale objects can be robust to noise and blur.

The evaluation of all cores in the images is computationally expensive and hence Fritsch *et al.* [40] proposed a method in which a few cores are selected from the *reference image*. These selected cores are then utilized for the calculation of cores of similar scales and position in the acquired image. Once the cores in the two image sets have has been selected, the authors propose two methods to estimate the transformation parameters. First, the authors use Balter's curve matching algorithm [3]. Second, they utilize the chamfer matching algorithm.

2.1.1.2 Chamfer Distance Based

Chamfer based registration has been proposed [70, 46, 75], where the authors propose to extract anatomical feature and the field edges automatically from the portal and simulator images. Registration of images using the chamfer distance method proceeds in the following two steps. First, one of the feature images is distance transformed, the most popular distance transform being the chamfer distance transform [18]. Leszczynski *et al.* [70], utilize the *5–7–11* 5×5 chamfer distance proposed in [18]. The *5–7–11* distance kernel for propagating local neighborhood distance is defined as

$$\begin{matrix} 14 & 11 & 10 & 11 & 14 \\ 11 & 7 & 5 & 7 & 11 \\ 10 & 5 & 0 & 5 & 10 \\ 11 & 7 & 5 & 7 & 11 \\ 14 & 11 & 10 & 11 & 14 \end{matrix}$$

Second, the features from the other feature image are mapped onto the distance transformed image, at the current estimates of the transformation parameters. Optimizing cost, as a function of the pixel values in the transformed image under the mapped features, determines the optimal transformation. The advantage of the chamfer matching method lies in the fact that this method can be easily automated, does not require identification of the corresponding features in the images and can be extended to the registration of multi–model images. However, again this method assumes that the automatic feature extraction method works well. Gilhuijs *et al.* [46] utilize an arithmetic mean of the pixel values of the distance transformed image as a cost function whereas Leszczynski *et al.* [70] utilize both the arithmetic mean and the root of mean square distance.

2.1.1.3 Open Curves–Based

Cai *et al.* [23] proposed to register the simulator and the portal images by matching open curves [3] extracted from the images. The algorithm samples corresponding points along the curves and a global transformation matrix is obtained by matching the point set. The proposed algorithm does not require the evaluation of the curvature of the curves.

2.1.2 Dense Field–Based

Dense field methods directly manipulate image pixel intensities and tend to be robust to noise and blur. However, these methods in general tend to be computationally expensive.

2.1.2.1 Least Squares Template Matching

Berger *et al.* [11] utilize a least squares template matching (LSM) algorithm for the registration of digitally reconstructed radiographs (DRRs) to portal image. The proposed LSM method is an area based method which does not require extraction of features from the images to be registered. In addition to the estimation of transformation parameters, the proposed algorithm tries to estimate and correct a linear relationship between pixel intensities in the two images. In this approach, authors assume the following relation between the images, $f(u)$ and $g(u)$ to be registered:

$$f(u) + e(u) = \overline{g}(x)$$

where, $\overline{g}(u) = \alpha + \beta g(u)$ define the linear relation which adjusts pixel intensities radiometrically, and $x = \psi(\eta, u)$ defines the geometric transformation. Square error $e^T Pe$, where P is an optional weight matrix, is minimized to estimate the geometric and inten-

sity transformation parameters. A multiple template approach, instead of using one large template, can possibly compensate for a global bias field. To compensate for out–of–plane rotations in portal images, the authors propose using an affine transformation. However, note that this strategy can account only for the in–plane transformation parameters.

2.1.2.2 Correlation–Based Methods

Hristov *et al.* [54] propose a correlation–based registration method to bring portal and simulator images into alignment. This method uses a fast–Fourier–transform based cross–correlation operator to find the optimal in–plane rotation and translations. The authors compare the performance of two cross–correlation operators: the Pearson's linear correlation coefficient (PCC) and the normalized correlation coefficient (NCC). Let $g(x, y)$ and $f(x, y)$ denote the two images to be registered. Let $w(x, y)$ denote a search window within the image $g(x, y)$, enclosing a feature to be matched in the image $f(x, y)$. Let $f_\alpha(x, y)$ denote the image obtained after transforming $f(x, y)$ by the set of transformation parameters α. Then, for each point (m, n) in the image $f_\alpha(x, y)$, the similarity between $w(x, y)$ and the region of $f_\alpha(x, y)$ can be evaluated as, the negative of the Pearson's linear correlation coefficient (PCC),

$$L_P^{m,n}(f_\alpha, w) = -r_\alpha(m, n) = \frac{\sum_x \sum_y [f_\alpha(x, y) - \overline{f}_\alpha(m, n)][w(x - m, y - n) - \overline{w}]}{\sqrt{\sum_x \sum_y [w(x - m, y - n) - \overline{w}]^2}\sqrt{\sum_x \sum_y [f_\alpha(x, y) - \overline{f}_\alpha(m, n)]^2}}$$

where $\overline{w}$ and $\overline{f}_\alpha$ denote the average intensity of the mask and the average value of $f_\alpha(x, y)$ in the region coincident with $w(x, y)$.

The negative normalized cross–correlation coefficient (NCC) is evaluated as

$$L_N^{m,n}(f_\alpha, w) = -c_\alpha(m,n) = \frac{\sum_x \sum_y f_\alpha(x,y) w(x-m, y-n)}{\sqrt{\sum_x \sum_y w^2(x-m, y-n)} \sqrt{\sum_x \sum_y f_\alpha^2(x,y)}}$$

Note that NCC is invariant under scaling of the image intensities, i.e. under the transformation $g(x,y) \to C_1 \cdot g(x,y)$ and PCC is invariant under the transformation $g(x,y) \to C_1 \cdot g(x,y) + C_2$, for some constants C_1, C_2.

Dong *et al.* [35] register portal images to megavolt digitally reconstructed radiographs (DRRs) by optimizing the cross–correlation coefficient in the regions of interest (ROIs). The megavolt DRRs are obtained by first mapping the CT attenuation coefficients from the therapeutic energy X–rays to the coefficient at the treatment energy and then obtaining a projection through the transformed CT in the prescribed direction. The resulting megavolt DRR pixel intensities are then transformed by histogram–matching to the histogram of the portal image pixel intensities before estimating the registration parameters.

2.2 Three–Dimensional Analysis

The previous section summarizes various strategies proposed in literature for the estimation of patient setup error utilizing two dimensional portal and simulator images. An inherent problem with the two dimensional strategies is that they cannot account for out–of–plane patient rotations. The out–of–plane rotations will lead to distortion of the anatomy between the portal and the simulator images, thus limiting the accuracy of the algorithms. These mis-registration errors will become significant while providing conformal radiotherapy. Hanley *et al.* [49] study the effect of out–of–plane rotations on

two–dimensional image registration and conclude that dosimetrically, errors introduced in the patient position by two–dimensional image registration for out–of–plane rotations of 2° or more have significant consequences. Thus, it is necessary that out–of–plane rotation errors be accurately estimated.

This section briefly reviews various strategies proposed in the literature which estimate patient setup error by three–dimensional image registration, that is, by registering a treatment planning 3D CT dataset to 2D portal images. These methods can again be classified as either feature based methods or dense field methods.

2.2.1 Feature–Based

2.2.1.1 Bony Ridge–Based

Gilhuijs *et al.* [44] describe an automated registration method for the verification of patient setup in three–dimension using a 3D CT data set and two orthogonal (the anterior–posterior (AP) and the left–lateral (LL)) portal images. In this method, bony ridge features are detected either automatically using a morphological top–hat operator or manually. This method is based on the intuition that the location of the bony ridges in the portal images concurs with the local maximum in the trajectory of the X–rays through the bone. The optimization procedure thus adjusts the pose of the 3D CT to maximize the distances of the X–rays through the bone between the irradiation source and the bony ridges in the portal images. The authors utilize Powell's minimization [89] and the downhill Simplex method [80] to optimize the cost function. We implemented this method for comparisions (See chapter 6).

2.2.1.2 Two–Dimensional Silhouette Based

Lavallee *et al.* [66] propose a general method for registering a 2D image to a 3D dataset. The algorithm assumes that the 2D image has been pre–processed and a silhouette has been extracted from it. The 3D image data set is also assumed to have been pre–processed to extract a 3D surface. The algorithm assumes that when the 3D surface is properly aligned, the radiation lines from the irradiation source to the 2D silhouette are tangential to the 3D surface. A signed distance metric is proposed which measures the sum of square distances of the rays from the 3D surface. This distance is minimized to estimate the optimal transformation parameters. To reduce the computational complexity of the algorithm, the authors utilize fast distance computation using octree–splines.

This approach is further extended by Hamadeh *et al.* [48] who exploit the duality between registration and segmentation in a model–based vision system. This method associates a likelihood value to each pixel in the projection images that corresponds to the probability that the pixel belongs to the silhouette of the 3D object on interest. Such a likelihood–based scheme is necessary as an automated edge detection method may lead to the detection of many spurious edges. The likelihood of a pixel belonging to a contour depends on three features: the gray scale gradient at the pixel, the length of the connected component to which the pixel belongs and the distance between the pixel and the contour projected from the 3D surface at its current estimated position. The cost function is then appropriately modified to utilize the additional information, leading to an automated and accurate registration method.

2.2.2 Dense Field–Based

2.2.2.1 Correlation Based

Lemieux *et al.* [69] present an automated pixel intensity correlation based technique for registering 3D CT image data to two portal images. In this strategy, digitally reconstructed radiographs (DRRs) derived from the CT dataset, at different scales and positions, are aligned iteratively with portal images to estimate the transformation parameters. Note that in each iteration, complete projections of the CT dataset in two different directions are obtained, thus increasing the computational complexity of the algorithm. The authors assume that the initial pose of the CT data set is close to the true pose and utilize Powell's multidimensional directions set method to search the optimal parameters.

2.2.2.2 Pattern Intensity

Weese *et al.* [120] propose pattern intensity as a similarity measure for the voxel based registration of 3D CT images and the intra–operatively acquired X–ray fluoroscopies. This approach requires segmentation of the vertebra in the pre–operative CT images and pseudo projections are computed using only a small part of the CT image, covering only the outlined vertebra. Thus, the pseudo projections show intensity variations only due to the vertebra. The pseudo projections are subtracted from the x–ray images to generate difference images. If the 3D CT is properly aligned to the x–ray images, the vertebra will not show in the difference image; otherwise some *structure* will show. The authors propose the following measure of *structuredness*, called the pattern intensity $P_{r,\sigma}(I_{diff})$,

in the difference image:

$$P_{r,\sigma}(I_{diff}) = \sum_{(x,y)} \sum_{\{(v,w)|(x-v)^2+(y-w)^2 \leq r^2\}} \frac{\sigma^2}{\sigma^2 + (I_{diff}(x,y) - I_{diff}(v,w))^2}$$

where I_{diff} denotes the difference image, r defines the size of the neighborhood in which gray–value variations are considered and σ acts as a threshold, estimated empirically, defining whether a gray–value variation is to be considered a structure or not. Pattern intensity is a function of the transformation parameters and is optimized to estimate the parameters. However, the pattern intensity remains an ad–hoc method at–best and requires pre–processing of the images.

2.2.3 Interactive Methods

The image registration methods described above can be broadly classified as being automated or semi–automated; that is, they require little or no interaction from the user.

Gilhuijs *et al.* [45] have proposed an interactive image registration technique for three dimensional patient setup verification using 3D CT data set and dual portal images. Three different methods for interactive patient setup verification are suggested. In the first method, the DRRs and the portal images are compared side by side. In the second method, anatomical structures extracted from the portal images are overlaid onto the DRRs. In the third strategy, DRRs and the corresponding portal images are rapidly alternated on the screen. A translation mismatch is perceived as a shifting motion and a rotational mismatch is perceived as a rocking motion. The authors found the third approach to be most sensitive to mismatch between the corresponding images.

Mubata *et al.* [78] have also proposed an interactive technique to register the portal images to the corresponding simulator images. Note that since two–dimensional

images are being used, this technique can estimate only the in–plane patient setup deviations. Neva *et al.* [82] propose to visually assess corresponding portal images and simulator images to correct for patient setup deviations. They found this strategy to be sufficient for detecting errors in the lateral direction, but not for errors in the longitudinal direction.

2.3 General Registration Methods

The problem of registering image datasets between the same and different modalities has been studied in the medical image analysis community for a variety of applications. Excellent reviews of these methods, with classification, have been published [113, 65, 21, 73]. Similar to the classification of methods in the previous section, these methods can also be classified as either dense field or sparse field based methods.

2.3.1 Feature–Based or Sparse–Field Methods

2.3.1.1 Point–Based Matching

Point based registration methods assume that if corresponding landmark points can be identified in the two images to be registered, then bringing these points into alignment will register the two images. Let $X = \{\mathbf{x_i}\}$ and $Y = \{\mathbf{y_i}\}$ for $i = 1, \ldots, N$ denote the two sets of points to be aligned. If we assume a transformation T that will align these points, then we can define a mean square cost function to be,

$$D(T) = \sum_{i=1}^{N} \|\mathbf{x_i} - T(\mathbf{y_i})\|^2$$

The cost function $D(T)$ is then minimized to estimate the transformation parameters. A singular value matrix decomposition method, proposed by Arun *et al.* [2], can be utilized to estimate the parameters if a rigid transformation is assumed between the set of points. However, note that this procedure assumes that the correspondence between the points in the two point sets is known and the two sets to be registered have same number of points. However, this is not true in general. *Iterative closest point* (ICP), proposed by Besl *et al.* [15], only requires a procedure to find the closest point on a geometric entity to a given point. The ICP algorithm has been shown to always converge to the nearest local minimum monotonically. Two other interesting approaches, discussed below, estimate both the correspondence and the transformation parameters between two sets of points.

Statistical Point Matching. Wells *et al.* [121] present a statistical approach to object recognition. The authors propose a probabilistic model for the correspondence between image features, edge points, and features belonging to either the object or to the background. In the Maximum A–posteriori (MAP) formulation, the authors explicitly solve for the correspondence between the feature and the pose parameters of the 3D model. Then the authors propose a maximum–likelihood (ML) formulation where only the pose of the 3D model is evaluated, which is solved using the expectation–maximization (EM) algorithm [32]. Correspondence between the feature points falls out as a by–product of this method. Note that this algorithm can handle a different number of features in the two feature sets. However, multiple image features can be mapped to a single feature belonging to the object or the background.

Robust Point Matching. Kosowsky *et al.* [63] propose an approach which solves for one–to–one correspondence between two sets of feature points. Using statistical physics,

the authors construct an effective energy and use steepest descent to derive an algorithm to solve the correspondence problem. However, note that this method, though solving for one–to–one correspondence, cannot handle different number of features in the two points set to be registered. Rangarajan *et al.* [91] extend this work in the following ways. Rangarajan *et al.* [91] develop an algorithm, within an optimization framework, that can handle a different number of points in the point sets to be registered and yet solve for one–to–one correspondence and the transformation parameters which will bring the two feature sets into alignment. Since the two feature sets can have a different number of features which cannot be mapped onto features into the other image some of the features are marked as outliers, automatically. This method provides a robust method of handling two sets of features and also, with a deterministic annealing schedule, the algorithm can converge to a global optimum. The algorithm has been further extended to utilize Procrustes distance [92] and also to the problem of articulated motion [84].

2.3.1.2 Surface–Based

Surface–based techniques aim at extracting surfaces of corresponding anatomical regions from the images which are then utilized to estimate the registration between the images. After preprocessing images, these methods can also be easily extended to register images from different modalities. If the extracted surfaces are represented as sets of points, then an ICP [15] based strategy can bring them into alignment.

Various surface matching algorithms for registering multi-modal images have been proposed [86, 74, 110]. Pelizzari *et al.* [86] align multi-modality images by matching corresponding surfaces extracted from the images. For registering MR and PET brain images, skin surfaces are semi–automatedly delineated in each slice of the MR and PET images to form 3D surfaces. A simple threshold boundary following algorithm is used to

extract surface contours in each slice. Manual editing of the surfaces is done as needed.

A very interesting feature based approach is described by Zubal *et al.* and Tagare *et al.* [125, 107] which supports complex landmark structures, including points, curves, planes and volumes. In this approach, the user can also specify whether the two structures correspond fully or partially.

2.3.1.3 Ridge Operator–Based

Van den Elsen *et al.* [111, 112] proposed using differential operators in scale–space for matching images from different modalities. Florack *et al.* and Romeny *et al.* [39, 96] proposed multi-scale differential operators to study the differential structure of the images. These operators are derived from the Gaussian function and its partial derivatives. The width of the Gaussian function, σ, determines the level in the scale space. An image is embedded into scale space by simply convolving it with a Gaussian function. Let L_0 denote the original image. The scale space image $L(x, y, \sigma)$ at scale σ is obtained as

$$L(x, y, \sigma) = L_0 * G(x, y, \sigma)$$

where $G(x, y, \sigma)$ is a zero mean Gaussian kernel of standard deviation σ and $*$ denotes convolution between the images. The differential image in scale space is obtained by convolution between the image and appropriate derivative of the Gaussian kernel, as follows:

$$L_x(x, y, \sigma) = L_0 * G_x(x, y, \sigma)$$

where $L_x(x, y, \sigma)$ denotes the derivative image along the x–axis, at scale σ, and $G_x(x, y, \sigma)$ denotes the derivative of the Gaussian function along the x–axis.

Van den Elsen *et al.* [111] define a differential *ridgeness* operator, denoted as L_{vv}, to enhance the ridges in the images to be registered. The differential operator L_{vv} denotes the second order derivative of the image in the direction v, where v is perpendicular to the direction of the local gradient, denoted by w. Using local Cartesian derivatives, L_{vv} can be computed as follows:

$$L_{vv} = \frac{1}{\|v\|^2}(v \cdot \nabla)^2 L = \frac{L_y^2 L_{xx} - 2L_x L_y L_{xy} + L_x^2 L_{yy}}{L_x^2 + L_y^2}$$

For registering two images, both images are transformed by convolving the images with the ridge operator L_{vv}. Of the two resulting binary images, one image is distance–transformed using chamfer distance transform. The two images are then registered by transforming the binary image. The sum of square distance is utilized as a match metric to evaluate the optimal spatial transformation parameters. Van den Elsen *et al.* [111] used this strategy to register a 2D CT image slice to a corresponding 2D MRI slice. The extension of the 2D ridge operator to three–dimensions is not straightforward and some early work has been proposed [38].

2.3.2 Dense–Field Methods

2.3.2.1 Correlating Gradient Images

Brown *et al.* [20] propose registering 2D real and simulated radiographs by first intensity–correcting the real radiograph. Intensity–corrected radiographs are then correlated with gradient operator to obtain gradient images. The resulting gradient images are then

correlated to estimate rigid in–plane transformation parameters.

2.3.2.2 Variance of Ratio Intensities

Woods *et al.* [124] proposed a method for the registering 3D images based on the intuition that when the two images are registered, the variance of the ratio of intensities in the overlapping regions of the images will be minimized. Note that this algorithm directly manipulates the pixel intensities from the anatomical regions in the image and hence can be used retrospectively and is robust to noise. Let a_i denote a voxel value in the first image and b_i denote the corresponding voxel value in the second image, let $r_i = a_i/b_i$ denote the ratio of the corresponding voxel values in the two images. The proposed algorithm evaluates $r_i, \forall i$ in the overlapping regions of the two images. Let σ_r and r_{mean} denote the variance and mean values, respectively, of the random variable r_i. The algorithm seeks to minimize the ratio

$$\nu = \frac{\sigma_r}{r_{mean}}$$

This is an interesting measure as it can be seen as a precursor to other popular methods based on mutual information. However, this measure remains largely heuristic.

2.3.2.3 Mutual Information Based

Viola *et al.* [116, 117] and Collignon *et al.* [27] proposed an information theoretic based match method which aligns two images by optimizing mutual information between the images. Let $\mathbf{x}$ and $\mathbf{y}$ denote two random variables. If $p(x)$ and $p(y)$ denote the probability density functions of the random variables $\mathbf{x}$ and $\mathbf{y}$ respectively, then the mutual

information between the random variables is defined as [83, 103]:

$$I(x, y) = H(x) + H(y) - H(x, y)$$

where $H(x)$ denotes the entropy of the random variable $\mathbf{x}$ defined as $H(x) = -\int p(x)\ \ln p(x)\ dx$ and $H(x, y)$ denotes the joint entropy of the random variables defined to be $H(x, y) = -\int\int p(x, y)\ \ln p(x, y)\ dx\ dy$. The probability density function $p(x, y)$ denotes the joint density function of $\mathbf{x}$ and $\mathbf{y}$.

As opposed to cross–correlation based methods, which assume a linear intensity relation between the images to be registered, mutual information based methods can successfully register images with a non–linear relation between the pixel intensities. This allows successful registration of multi-modal images.

Viola *et al.* [117] proposed a Parzen window–based [37] stochastic gradient descent implementation for estimating the optimal parameters, whereas Collignon *et al.* [27] used a histogram based approach.

2.3.2.4 Normalized Mutual Information

The mutual information–based methods described above assume that the images being registered contain similar portions of the structure. However, when registering a small image to a large image, say registering a brain MR image to a whole body MR image, it might be necessary to normalize the entropies in the overlapped region to obtain an accurate registration. Studholme *et al.* [105] proposed an overlap–normalized entropy measure which can handle these situations correctly. The cost function to be optimized

is defined as

$$C(x, y) = \frac{H(x) + H(y)}{H(x, y)}$$

which is a ratio of the sum of marginal entropies to the joint entropy of the random variables.

In this work, to search for the optimal transformation parameters, a multi-scale strategy is employed, both to reduce the computational complexity and to increase robustness to noise. At each scale, or resolution of the images, optimal parameters are estimated by sampling the parameter space. At low resolution, the parameter space is coarsely sampled over a wide range. At high resolution, the parameter space is finely sampled over a small range of parameters. Optimal parameters estimated at one scale are passed to the next scale as the initial estimate of parameters.

2.3.2.5 Incorporation of Segmentation Information

While registering multi-modal images, regions of the same anatomical structures may not be connected in one modality whereas they might appear connected in the other modality, especially when registering MR and PET images. Studholme *et al.* [104] utilize a region labeling information channel, in addition to the pixel intensity channels usually utilized in mutual information based methods. These labels are used to specify the regional connectivity which can possibly be derived from the images to be registered. However, the images are manually segmented and labeled [104].

2.3.2.6 Using Joint Density Function

Leventon *et al.* [71] proposes a maximum–likelihood method for registration of multi-modal images. Before registering a set of multi-modal images, a parametric joint density function of the pixel intensities is estimated by using a pre–registered training set of images from the same modality. This supervised method of learning determines the parameters of the joint density function. Once the parameters of the joint density function are estimated, the log of the joint density function is optimized for the images to be registered, to estimate the rigid transformation parameters using Powell's [89] method of optimization.

2.3.3 Non–Rigid Image Registration

The image registration methods described above aim at estimating a rigid transformation between a set of images to be registered. However, for some applications, it is necessary to assume a non–rigid transformation between the images to be registered. For example, in this project, the prostate is a soft tissue which deforms and so the registration of prostate in the ultrasound images and the 3D CT image would require non–rigid image registration strategy. There is a rich literature of methods that estimate a non–rigid transformation between a set of images [100, 115, 81, 108, 26] and we discuss only a few methods here. Gaens *et al.* [42] extended the mutual information–based match method to non–rigid multi-modal image registration.

2.3.3.1 Optical Flow Based Methods

Vemuri *et al.* [115] propose an interesting image registration algorithm based on optical flow computation, where the registration between two same modality images is formulated

as a motion–estimation problem. The flow field is represented using B–splines basis functions, thus implicitly incorporating smoothness in the estimated field. Further, a hierarchical optical flow motion model is employed to capture both the global and local flow fields. Utilizing the technique proposed in [34, 22], Vemuri *et al.* [115] use a modified Newton's method of optimization using Hessian of the energy function pre-computed at the optimal point.

2.3.3.2 Elastic Model Based

The non–rigid, fluid flow–based, registration methods proposed in the literature suffer from the under–constrained, ill–posed [12] nature of the problem. Wang *et al.* [118, 119] propose incorporating statistical shape model, proposed by Cootes *et al.* [29], into elastic model–based registration to help constrain the problem, thus hopefully leading to an accurate image registration algorithm. The statistical shape models are used to model the physical deformations of elastic solids. Using these shape models, the object of interest is segmented from the images. As a point–based shape model is used to segment the objects, this step also establishes correspondence between a set of points in the images to be registered. As the corresponding points in the two images should match, these points are further used to constrain the elastic model based registration of the images.

2.4 Summary

This chapter summarized a large number of techniques proposed in the literature to register 3D–CT image datasets to dual portal images and also described methods popularly used in the medical image analysis field. The methods have been broadly classified into two categories, first the sparse–field or the feature–based methods and second the dense–

field methods. The sparse–field methods are, in general, computationally less expensive as compared to the dense–field based methods. However, the accuracy of the sparse–field methods is limited by the accuracy of the feature extraction algorithms. Dense–field methods are robust to noise and blur but might be poor in accuracy. All the strategies have some merits and some down-sides. Our proposed registration framework simultaneously performs registration and segmentation, integrates segmentation results into the registration process, and uses *both* intensity–based and feature–based information as part of the registration process.

Chapter 3

Image formation models and background

3.1 Introduction

The previous chapter reviewed various registration methods proposed in the literature. This chapter briefly discusses the mathematical details of the methods used in the registration framework proposed in this thesis. The first section of this chapter details the physics of image formation. This section, in particular, discusses the formation of diagnostic energy 3D CT images, megavolt portal images, the rendering method used to obtain digitally reconstructed radiographs (DRRs) and the projection model. The second section describes the expectation–maximization (EM) algorithm [32], discusses its strengths and its weaknesses, highlights the problem it solves and works out a simple example. The third section discusses the mathematics of the Markov random fields (MRF) and their application to region based image segmentation. This section also discusses the *Ising model*, which is a special case of MRFs. Stochastic relaxation [43] is a popular

approach to estimate the MRF based image segmentation of image. We present a mean field based approximation to the Ising model to estimate image segmentation. The last two sections discuss the mathematics behind the mutual information based method and some popular methods used to estimate probability density functions.

3.2 Formation of Images

This section will briefly describe how different modality images, which are the main focus of this thesis, are formed. Christensen *et al.* [25] provides an excellent overview of the subject. In particular, we are interested in diagnostic quality 3D CT datasets and the treatment energy 2D portal images. From 3D CT datasets, 2D digitally reconstructed radiographs (DRRs) are rendered, after the 3D CT voxel values have been mapped from diagnostic energy to those at treatment energy using conversion tables [56]. For rendering DRRs, a perspective projection model is assumed and will be briefly described.

3.2.1 Computer Tomography Images

Computed tomography (CT) reconstructs internal structures of an object from multiple projections of the object. In X–ray CT, the projections are formed using the X–ray projections in many different directions. Thin cross–sections of a body are reconstructed, one at a time, by taking projections from different angles with a narrow X–ray beam. The numerical data from the projections for each cross–section is then computer processed to reconstruct an image.

Each cross–sectional layer is divided into tiny blocks, called *voxels*. Each voxel is assigned a number proportional to the degree it attenuates the X–ray beam. A linear attenuation coefficient, μ, is used to measure the amount of attenuation caused by each

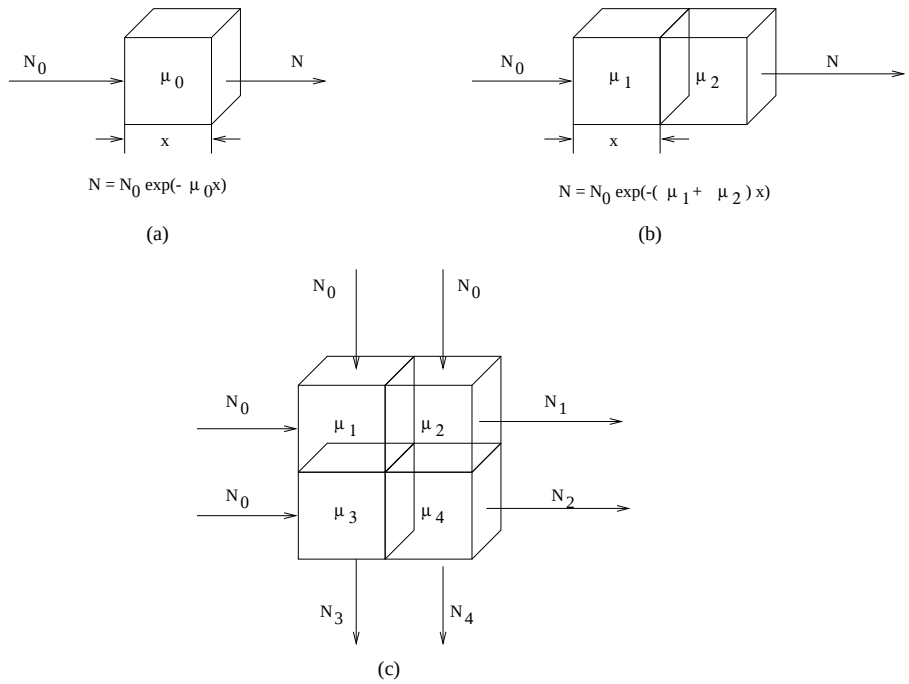


Figure 3.1: Attenuation of Photons

voxel. A monochromatic X–ray source is assumed which has an attenuation coefficient μ_0 for the tissue in a voxel (Note that the attenuation coefficient is a function of the tissue type and energy of the X–rays). If x is the thickness of the voxel, N_0 are the number of X–ray photons entering the voxel, then the number of photons, N, leaving the voxel is given as:

$$N = N_0 \ \exp(-\mu_0 \ x)$$

If there are two voxels in the path of the beam, each of width x, see figure 3.1 (b), then

the equation becomes:

$$N = N_0 \ \exp(-(\mu_1 + \mu_2)\ x)$$

Given N and N_0, this equation cannot be solved to estimate μ_1, μ_2. One more equation is required to uniquely estimate the attenuation coefficients. Consider figure3.1 (c). Projections for a 4×4 array of voxels are obtained from two different directions which allows us to construct four different equations:

$$\begin{aligned}
N_1 &= N_0 \ \exp(-(\mu_1 + \mu_2)\ x) \\
N_2 &= N_0 \ \exp(-(\mu_3 + \mu_4)\ x) \\
N_3 &= N_0 \ \exp(-(\mu_1 + \mu_3)\ x) \\
N_4 &= N_0 \ \exp(-(\mu_2 + \mu_4)\ x)
\end{aligned}$$

Note that these equations can now be uniquely solved to estimate the linear attenuation coefficients, μ_1, μ_2, μ_3 and μ_4 and hence reconstruct the 4×4 array.

If there are n voxels with attenuation coefficients $\mu_1, \mu_2, \ldots, \mu_n$ in the path of the X–ray, then the number of photons leaving at the end of the nth voxel is given by the relation:

$$N = N_0 \ \exp(-(\mu_1 + \mu_2 + \cdots + \mu_n)\ x)$$

To solve for the linear attenuation coefficients, at-least n more simultaneous equations are required. Thus, this shows the necessity of obtaining projections from many different

directions for the reconstruction of a single slice.

Filtered Back–Projection [97] is the most popular method of reconstructing a slice of data from a set of projections. Filtered back–projection, as the name suggests, first filters the projection data and then back-projects it to reconstruct a slice. Filtering is carried out to counterbalance blurring within a slice due to sudden changes in the intensities.

3.2.1.1 CT Numbers

For each pixel in a slice, the calculated linear attenuation coefficient is normalized to the attenuation coefficient of water (μ_w), multiplied by a scaling constant (K), usually 1000, converted to next the higher integer, and reported as a new number called the *CT number*. Thus, the CT numbers are evaluated as follows:

$$\text{CT Number} = K\,\frac{\mu_p - \mu_w}{\mu_w}$$

where μ_p is the calculated linear coefficient of a pixel.

3.2.2 Portal Images

Portal images are X–ray projection images obtained with treatment energy photons (2 MeV – 25 MeV). As noted in the previous section, the linear attenuation coefficients of tissues are functions of the energy of the photon energy. For higher energy photons, differences in the attenuation coefficients of different tissues are reduced, leading to low contrast X–ray images. This section derives a mathematical relation between the CT voxel values and the pixel intensity value of the portal image. Usually the portal images are obtained on an X–ray film (as opposed digital detector), further affecting the quality

of the image. Brown *et al.* [20] also present a similar relationship.

Assume that the linear attenuation coefficients associated with the CT voxel values have already been mapped from diagnostic energies to those at treatment energies. Further assume that the X–rays are of monochromatic energy. Let r_0 be an X–ray from radiation source to a pixel in the portal image, v be a small volume along the X–ray through the CT volume and $\beta(v)$ denote the linear attenuation coefficient of the volume v. If N_{in} denotes the number of photons entering the 3DCT volume, then the number of photons exiting, N_{out}, is given as:

$$N_{out} = N_{in} \ \exp\left\{ - \int_{r_0} \beta(v) \ dv \right\}$$

or

$$\ln\left(\frac{N_{out}}{N_{in}}\right) = - \int_{r_0} \beta(v) \ dv$$

The information content of the X–ray image must be transferred on an X–ray film to be visible. Portal images are usually obtained on a film. The blackening of a film is directly related to the intensity of the radiation reaching the film, which is measured as *photographic density* and usually called the *density* of the film. If, I_0 is the light intensity incident on a film, I_t is the light intensity transmitted, then the density, D, of the film is defined to be:

$$D = \ln \frac{I_0}{I_t}$$

In the linear part of the *characteristic curve* of a film, the density of the film is directly

proportional to the logarithm of the exposure. Thus, in the linear part of the characteristic curve:

$$\ln\left(\frac{N_{out}}{N_{in}}\right) \propto \ln\frac{I_0}{I_t}$$

Thus, we have:

$$\ln\frac{I_0}{I_t} \propto -\int_{r_0} \beta(v)\, dv$$

When the X–ray film is digitized, the intensity of a pixel, I_r, is proportional to I_t, the intensity of the light transmitted by the film. Thus, we can write,

$$I_r \propto \left(\frac{I_t}{I_0}\right)$$

Assuming that the digitizer performs a linear transformation of the transmitted intensity, we have:

$$a_1 I_r + b_1 = \left(\frac{I_t}{I_0}\right)$$

and hence,

$$\ln\left(\frac{1}{a_1 I_r + b_1}\right) \propto -\int_{r_0} \beta(v)\, dv$$

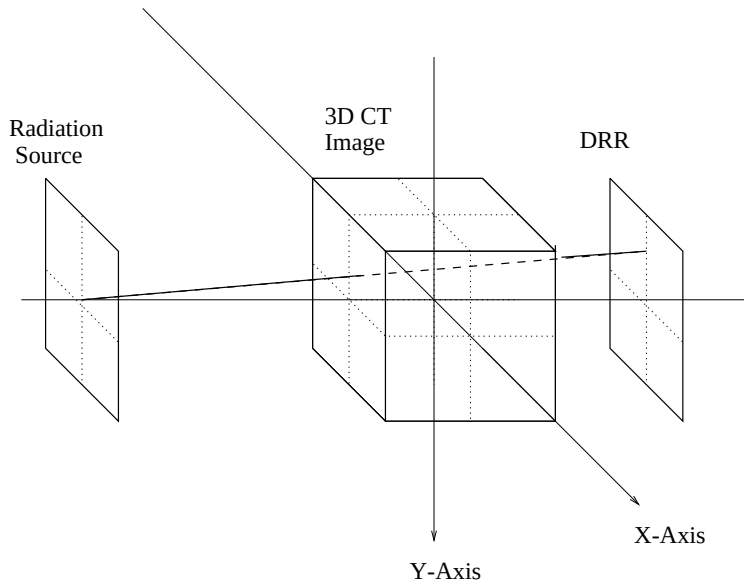


Figure 3.2: Setup for 3D CT and DRR

which can further be reduced to

$$\ln(I_r + k) \propto \int_{r_0} \beta(v)\, dv \tag{3.1}$$

The equation (3.1) thus specifies the relation between pixel intensity of a digitized portal image to the voxel values along the projection X–ray.

3.2.3 Projection Model

For obtaining digitally reconstructed radiographs (DRRs) from the 3DCT dataset, a perspective projection model is assumed. The radiation source is assumed to be a point source and the various projection parameters are also assumed to be known. Figure 3.2 shows how the 3DCT, radiation source and the DRR plane are setup in three space.

The pixel intensity in the DRR is obtained by summing the 3DCT voxel values along the projection ray from the pixel to the radiation source. The CT values have been mapped to the values at treatment energy X–rays using the procedure as follows. First assuming a monochromatic X–ray beam, the tissue composition of each voxel in the CT volume is estimated by knowing the energies of the diagnostic energy X–rays and the linear attenuation coefficient of each voxel. Second, since we know the tissue type in each voxel and the energies of the X–rays at the treatment energy X–rays, the attenuation coefficient for each voxel is calculated using the attenuation tables in [56]. Note that these tables take care of the Compton scattering and the photo–electric effects at various X–ray energies. Also, equation (3.1) describes the relation between the pixel intensity in the projection image and the sum of the CT voxel values along a projection ray. To estimate the CT value at the point where a projection ray intersects a slice of 3DCT, bilinear interpolation of the four neighboring voxel values is used. The various perspective parameters, the distance between the radiation source and the origin (of the 3D coordinate system) and the distance between the DRR and the origin are assumed to be known.

3.3 The EM Algorithm

The *Expectation–Maximization* (EM) [32] algorithm is an iterative computation of maximum–likelihood (ML) estimates of parameters of a distribution. Each iteration of the algorithm consists of two steps, the expectation step, followed by the maximization step, and hence the name, the EM algorithm. This algorithm is especially useful for problems where the observed data can be viewed as incomplete data, that is, when there is a many–to–one mapping from an underlying distribution to the distribution governing the observations. Some of the situations where this might arise are when outcomes are clumped before they

are observed, e.g. in histogram operations, or situations where the information regarding the class to which a data belongs to has been lost. Mixture models are the most popular models used in the last situation. Meng *et al.* [76] give an excellent historical account of the application of the algorithm with a summary of extensions. Moon [77] gives an excellent introduction of the algorithm to the signal processing community.

3.3.1 Problem Definition

In this section, we will follow the introductory flavor of Tagare [106]. Consider a maximum–likelihood (ML) problem of parameter estimation. Let

$$\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^T$$

be a set of observations from n experiments, a random vector with a specified probability density function $p_{\mathbf{x}}(x|\theta)$, which is a function of parameters θ:

$$\theta = (\theta_1, \ldots, \theta_m)^T$$

Given the observation vector $\mathbf{x}$, the problem is to estimate the parameter vector θ which best explains the observation in the maximum–likelihood (ML) sense. If it is known that θ is a random vector and if the probability density function for θ is also specified, then this can be used as prior information in a maximum a–posteriori (MAP) formulation of the parameter estimation. Mathematically, if $\hat{\theta}$ is the optimal parameter, then the ML principle states that it should be chosen as

$$\hat{\theta} = \arg\max_{\theta} \; p(x|\theta)$$

and $\hat{\theta}$ is then called the *maximum–likelihood estimate* (MLE) of the parameter vector θ. For mathematical convenience, this is converted into the log–likelihood estimation problem by taking *log* of the likelihood probability density function:

$$\hat{\theta} = \arg\max_{\theta} \ \ln \ p(x|\theta) = \arg\max_{\theta} l(\theta|x)$$

If $l(\theta|x)$ is a continuous function of θ, standard gradient descent–based optimization methods can be utilized to estimate the optimal parameters $\hat{\theta}$.

3.3.2 Intuition

Why do we need anything more? In some of the applications, $\mathbf{x}$ can be best viewed as *incomplete data*, which can be obtained from *complete data* $\mathbf{y}$ through a many–to–one mapping. However, the complete data $\mathbf{y}$ is not known but ironically the joint density function $p_{\mathbf{y}}(y|\theta)$ might turn out to be in an analytically much simpler form than the density function $p_{\mathbf{x}}(x|\theta)$. Thus, we would like to use $p_{\mathbf{y}}(y|\theta)$ in the parameter estimation problem, but without having to worry about additional *missing data*. The EM algorithm provides a solution to this problem.

Mathematically, there are two sample spaces, $\mathcal{X}$ and $\mathcal{Y}$, and there is a many–to–one mapping from $\mathcal{Y}$ to $\mathcal{X}$. The observed data $\mathbf{x}$, called *incomplete data*, is a realization of $\mathcal{X}$. The corresponding $\mathbf{y}$, called *complete data* in $\mathcal{Y}$ is not observed. Let $\mathcal{Y}(\mathbf{x})$ denote the subset in $\mathcal{Y}$ from which the observed data $\mathbf{x}$ can be obtained. If $p_{\mathbf{x}}(x|\theta)$ is the probability density function of the observed data over the sample space $\mathcal{X}$, and $p_{\mathbf{y}}(y|\theta)$ is the corresponding probability density function of the complete data over the sample

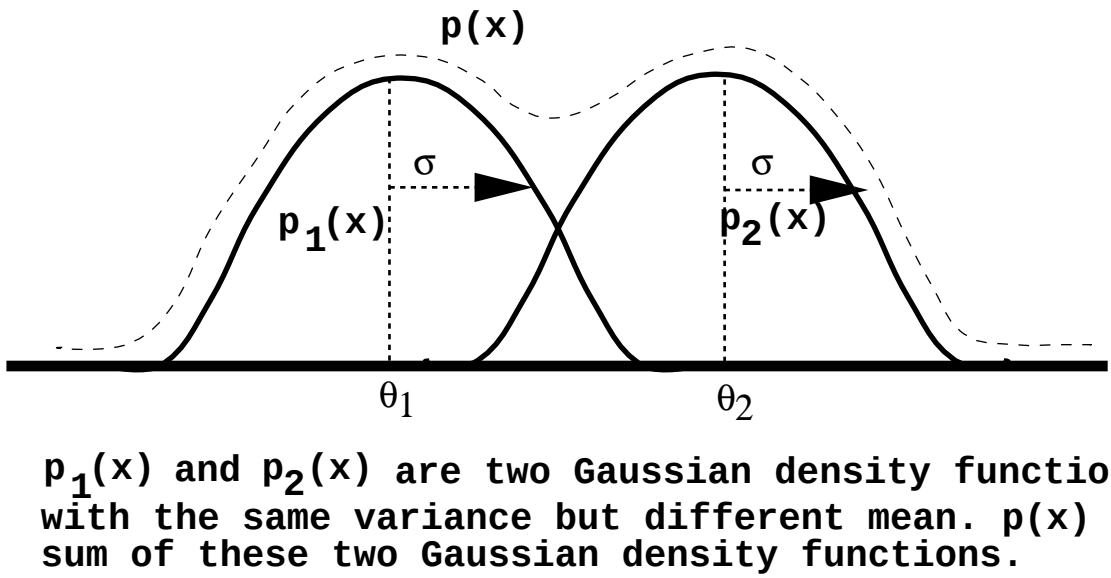


Figure 3.3: Mixture of two Gaussian

space $\mathcal{Y}$, the two density functions are related as follows:

$$p_{\mathbf{x}}(x|\theta) = \int_{\mathcal{Y}(\mathbf{x})} p_{\mathbf{y}}(y|\theta)\, dy$$

Notice that for a given incomplete data specification $p_{\mathbf{x}}(x|\theta)$, there are many possible complete data statistics $p_{\mathbf{y}}(y|\theta)$ that will generate $p_{\mathbf{x}}(x|\theta)$. The EM algorithm aims at estimating the value of θ which maximizes $p_{\mathbf{x}}(x|\theta)$ given the observed data, but in doing so it uses the associated family $p_{\mathbf{y}}(y|\theta)$. From these comments, it is obvious that the EM approach is useful only for the problems where, first, the statistics $p_{\mathbf{y}}(y|\theta)$ are obvious from the given $p_{\mathbf{x}}(x|\theta)$ and, second, where using $p_{\mathbf{y}}(y|\theta)$ is analytically more tractable than $p_{\mathbf{x}}(x|\theta)$.

Consider a probability density function labeled $p(x)$ in Figure 3.3, which is a function of three parameters, $\theta_1, \theta_2, \sigma$. Further, assume that $p_1(x)$ and $p_2(x)$ are two Gaussian density functions with parameters θ_1, σ and θ_2, σ respectively. The parameter estimation problem is stated as follows: *Given a set of observations* $\{x_1, x_2, \ldots, x_n\}$ *distributed according to the probability density function* $p(x)$*, estimate the set of parameters* $\theta_1, \theta_2, \sigma$. Note that even for this simple problem, analytically it is difficult to write the density function $p(x)$. It is easier to consider $p(x)$ to be a mixture of densities, $p_1(x), p_2(x)$, and write:

$$p(x) = p(c = 1)\ p_1(x) + p(c = 2)\ p_2(x)$$

where c denotes the class from which an observation x_i is drawn. If an observation belongs to class 1, it is distributed according to the density function $p_1(x)$ and if it belongs to class 2, it is distributed according to $p_2(x)$. Dependence on the set of parameters $\theta = \{\theta_1, \theta_2, \sigma\}$ can be explicitly shown as

$$p(x|\theta) = p(c = 1)\ p_1(x|\theta) + p(c = 2)\ p_2(x|\theta)$$

However, for the estimation of the parameters θ, only the set of observations $\{x_1, x_2, \ldots, x_n\}$ are given, the information about which class each observation belongs to is lost. This becomes a typical example where the EM strategy can be utilized to estimate θ, by assuming the class information as the missing data, thus simplifying the problem analytically.

As a closing note for this section, it should be noted that the EM algorithm leads to only a maximum–likelihood estimation of the parameters of the density function.

3.3.3 General Statement

As discussed above, random variables $\mathbf{x}$ and $\mathbf{y}$ are defined over the sample spaces, $\mathcal{X}$ and $\mathcal{Y}$ respectively, and there is a many–to–one mapping from $\mathcal{Y}$ to $\mathcal{X}$. $\mathcal{Y}(\mathbf{x})$ denotes the subset in $\mathcal{Y}$ from which the observed data $\mathbf{x}$ can be obtained. The relation between the density functions $p_{\mathbf{x}}(x|\theta)$ and $p_{\mathbf{y}}(y|\theta)$ is described in the previous section. Then, the two steps in the EM algorithm to compute the parameters θ are derived to be:

E Step: Compute

$$Q(\theta|\theta^k) = E[\ln p(y|\theta)|x, \theta^k]$$

where the expected value is evaluated with respect to the density function $p(y|x, \theta^k)$.

M step: Estimate

$$\theta^{k+1} = \arg\max_\theta Q(\theta|\theta^k)$$

The E–step requires evaluation of the expected value of $\ln p(y|\theta)$, the reason being to *integrate out* the missing data, and thus, to write the objective function only in terms of observed data and the parameters to be estimated. Taking the expectation with respect to the density function $p(y|x, \theta^k)$ is of core importance to the iterative nature of the algorithm. Note that as the expectation is taken with respect to the density function with parameters estimated from the previous iteration, θ^k, this helps give an iterative nature to the algorithm. The other advantage to using this density function is that it allows using Jensen's inequality [51] to simplify the E–step. The EM algorithm consists of choosing an initial estimate of the parameters θ^0 and iterating the two steps until convergence. The EM algorithm has been shown to monotonically converge to a local minima.

The M–step, in all its simplicity, is a classical optimization problem. Except for

some simple cases, it is not possible to form a closed form solution for the optimal set of parameters θ^{k+1}. Thus, one must resort to standard numerical techniques to estimate θ^{k+1} which *increases* $Q(\theta|\theta^k)$, but may not maximize it. Estimation of such a sequence of parameters is also shown to converge to a local optimum and such a procedure is called the *generalized EM* (GEM) algorithm [32].

3.3.4 Mixture Model

Redner *et al.* [93] provide an in–depth analysis of the mixture density problem with maximum likelihood and EM based solutions. Duda *et al.* [37] provide a concise overview of the problem.

The problem of interest is a parametric family of finite mixture densities of the form

$$p(x|\theta) = \sum_{i=1}^{m} \alpha_i \, p_i(x|\theta_i), \quad x \in \Re^m$$

where each $\alpha_i \geq 0$, also called the *mixing proportions* which satisfy the constraint $\sum_{i=1}^{m} \alpha_i = 1$. Each p_i is a valid probability density function parameterized by θ_i, and is also called the *component density* function.

3.3.5 Alternate Views

Neal *et al.* [79] and Hathaway [52] provide a coordinate descent interpretation of the EM algorithm for mixture distributions, which is used to study the convergence properties and justify some of the incremental variants of the EM algorithm.

Let $\mathbf{x}$ be the observed random variable and $\mathbf{y}$ be the missing data. The parameters of density function θ are to be estimated. The joint density function is given as $p(x, y|\theta)$

and the marginal density function of $\mathbf{x}$ is then

$$p(x|\theta) = \int p(x, y|\theta)\ dy$$

The EM algorithm starts with an initial estimate of the parameters θ^0 and proceeds iteratively, until it converges to a local optimum, by applying the the E and M steps:

E Step:

Compute a density function $\tilde{p}^t$ over the range of $\mathbf{y}$ such that $\tilde{p}^t(y) = p(y|x, \theta^{t-1})$

M step:

$$\theta^t = \arg\max_\theta\ E_{\tilde{p}^t}[\ln p(x, y|\theta)]$$

Neal *et al.* [79] show that the two steps of the EM algorithm can be seen as increasing the same function $F(\tilde{p}, \theta)$ defined as:

$$F(\tilde{p}, \theta) = E_{\tilde{p}}[\ln p(x, y|\theta) + H(\tilde{p})$$

where $H(\tilde{p}) = -\int \tilde{p}(y)\ \ln \tilde{p}(y)\ dy$ is the entropy of the distribution $\tilde{p}$. $F(\tilde{p}, \theta)$ can also be written in terms of the *Kullback–Leibler* divergence (see definition 3.2) between the distributions $\tilde{p}(y)$ and $p(y|x, \theta)$ as:

$$F(\tilde{p}, \theta) = -D(\tilde{p}(y)\|p(y|x, \theta)) + l(\theta|x)$$

Now the EM steps are equivalent to the following two steps:

E Step:

$$\tilde{p}^t = \arg\max_{\tilde{p}} F(\tilde{p}, \theta^{t-1})$$

M step:

$$\theta^t = \arg\max_{\theta} \; F(\tilde{p}^t, \theta)$$

3.3.6 Numerical Advantages

In addition to the above stated reasons, there are many numerical reasons for the widespread popularity of the EM algorithm. In principle, the EM algorithm is simple. Every iteration of the algorithm increases the likelihood function, until convergence. Thus, the convergence to a local optimum is guaranteed. Also, unlike most other optimization techniques, the EM algorithm does not require computation of gradients or Hessians of the cost function, nor is it necessary to worry about setting the step size parameters, as required in gradient based techniques.

3.4 Markov Random Fields

Region–based segmentation techniques aim at assigning each pixel in the image to a class depending on the local neighborhood characteristics of the image. A number of region–based segmentation techniques have been proposed, of which statistical techniques have become recently popular. Statistical techniques model an image as a random field. Assuming an image to be a homogeneous and ergodic random field allows maximum likelihood (ML) and maximum a–posteriori (MAP) formulations of the image segmentation and reconstruction problems. Dynamic programming [33, 50], stochastic relaxation [43], simulated annealing [62], and deterministic relaxation [13, 14, 67] techniques have been proposed to optimize the cost function to estimate a segmentation of the image.

Markov random field (MRF) models have been successfully used for image analysis and texture segmentation, especially for noisy images [43]. Leahy *et al.* [67] utilized an MRF model for medical image analysis. Image segmentation and analysis is in general an ill–posed [88] problem. Prior information is usually used to have a regularizing effect on the problem so that more likely images are favored. In texture segmentation and image analysis problems, images usually consist of different regions with slowly varying pixel intensities, except at region boundaries. Prior information which captures the local characteristics of the image will be a powerful tool for solving many image processing problems. MRFs provide a systematic way to incorporate the prior information, in terms of local smoothness of the image regions, to regularize the problems. Further, region boundary information can also be incorporated into the problem by using *line processes* [43].

Let $Z = \{(i,j) : 1 \leq i \leq M,\ 1 \leq j \leq M\}$ denote the $M \times M$ integer lattice over which the image pixel intensities are defined. Let the random variable $\mathbf{x}_{i,j}$ denote the pixel intensity at the site $(i,j) \in Z$. An image is represented as a random field, $\mathbf{X}$, which is a collection of the random variables $\mathbf{x}_{i,j}$ and can be written as:

$$\mathbf{X} = \{\mathbf{x}_{i,j},\ 1 \leq i \leq M,\ 1 \leq j \leq M\} \in \Re^{M \times M}$$

Let $\mathbf{X}_{Z/(i,j)}$ denote the set of all random variables except for $\mathbf{x}_{i,j}$, and let $N_{i,j}$ be the sets of sites *neighboring* the site (i,j). Thus, the random variables $\mathbf{X}_{Z/(i,j)}$, $\mathbf{X}_{N_{i,j}}$ are the set of random variables, on the lattice Z:

$$\mathbf{X}_{Z/(i,j)} = \{\mathbf{x}_{u,v} : \forall (u,v) \in Z : (u,v) \neq (i,j)\}$$

$$\mathbf{X}_{N_{i,j}} = \{\mathbf{x}_{u,v} : \forall (u,v) \in N_{i,j}\}$$

Let Ω define all possible configurations of $\mathbf{X}$:

$$\Omega = \{\omega = (x_{1,1}, x_{1,2}, \ldots, x_{M,M}) : x_{i,j} \in \Re\}$$

and let $p_{\mathbf{X}}(\omega)$ be a valid probability density function over Ω. Then, the random field $\mathbf{X}$ is said to be a Markov random field if the conditional probability density function satisfies the following relation:

$$p(x_{i,j}|X_{Z/(i,j)}) = p(x_{i,j}|X_{N_{i,j}})$$

that is, the conditional density function of $\mathbf{x}_{i,j}$ depends only on its neighbors on the lattice Z. Neighborhoods in MRF models can be defined with great flexibility, which can be tuned to the problem to be solved. Figure 3.4 shows example neighborhood systems. In the 0th order neighborhood system, each site is a neighbor of itself only. The 1st order neighborhood system consists of the four nearest sites, a 2nd order neighborhood systems consists of the eight nearest sites and so on. Once a neighborhood system has been decided for the random field, $\mathbf{X}$, *cliques* are defined. A clique is a set of one or more sites such that each site is neighbor to all the sites in the set, for that neighborhood system. Figure 3.4 shows that for higher order neighborhood systems, the number of cliques can increase rapidly.

Conditional probability density functions capture the correlation between neighboring sites, which allows us to build priors on the possible set of images with neighborhoods that are small enough to ensure feasible computational loads and yet still rich

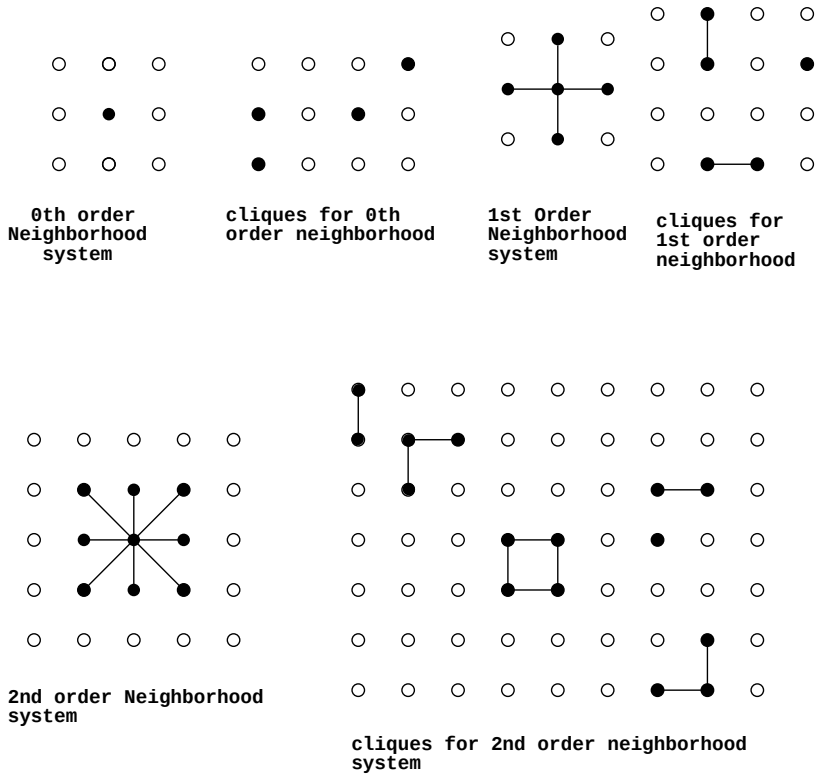


Figure 3.4: Neighborhood Systems and Cliques

This figure shows 0th, 1st and 2nd order neighborhood systems and some of the possible cliques for the corresponding neighborhoods.

enough to model and restore interesting classes of images and textures.

However, by itself, there are two problems with the MRF formulation so far:

- Even though specifying the conditional probability density functions should uniquely determine the joint density function, it is not apparent what is the joint density function.
- It is very difficult to know when a given set of functions are conditional density

functions for some probability density function on Ω.

The problem is overcome by using the *Hammersley–Clifford* theorem [13]. The advantage of the Hammersley–Clifford theorem is that one needs to specify only the potential functions $V_c(\mathbf{X})$ for the cliques of a random fields, where the potential functions are usually chosen to reflect the desired properties of the image. The only condition one needs to satisfy is that the potential functions be summable. The images with these desired properties will have low energy and hence will be more probable. This prior information is used, along–with a likelihood term, for MAP estimation and segmentation of images.

3.4.0.1 Stochastic Relaxation

Geman *et al.* [43] proposed a stochastic relaxation optimization technique for estimating a segmentation of an image. The stochastic relaxation based approach has been shown to converge to a global optimum. Note that stochastic relaxation (SR) is a very different approach to solving the image segmentation and analysis algorithm than the relaxation labeling (RL) algorithm [57, 97]. SR and RL have shared features and shared goals, locality and parallelism, but RL remains ad hoc and heuristic. RL is a *non–stochastic* process, both in the interaction model between the sites and in the updating algorithms, and there is no joint probability law over the configurations.

3.4.1 Ising Model

This section describes the 1D Ising model which has been studied extensively in the statistical physics literature [85, 24] to study the phase transition properties of the materials. The Ising model can be seen as a simple MRF.

Consider a discrete 1D lattice with a spin s_i at the ith site on the lattice, where $s_i \in \{+1, -1\}$. In the presence of an external, site dependent, field h_i, the energy of the system is written as

$$H_I = -\frac{J}{2} \sum_{(i,j)} s_i s_j - \mu \sum_i h_i s_i$$

where the sum over i and k is taken over the nearest neighbor and J is called the coupling constant. $J > 0$ favors a configuration where the neighbors are aligned in the same direction. The simplicity of the Ising model is deceptive. Even after the first introduction of the Ising models, exact solutions for the model are available for only in one–dimension with uniform field and in two–dimensions at zero field. Mean field approximations have remained as an attractive method to evaluate approximate solutions.

3.4.2 Mean Field Approximations

This section briefly discusses the mean field approximation solution to the 1D Ising model. Chandler [24] discusses the approximate solution for the site independent external field case and obtains a closed form solution for the problem. For the site dependent external field see Parisi *et al.* [85]. For the mean field development followed in this thesis, see Hofmann *et al.* [55].

Consider the problem with a site–independent external field. The Ising model problem can be defined to be the problem of evaluating the Boltzmann weighted sum:

$$Q = \sum_{s_1, s_2, \ldots, s_N} \exp[-\beta E(s_1, s_2, \ldots, s_N)]$$

where

$$E(s_1, s_2, \ldots, s_N) = -\frac{J}{2}\sum_{(i,j)} s_i s_j - \mu \sum_i h s_i$$

In the variational analysis, a mean field approximation of the energy function is of the form:

$$E_{MF}(s_1, s_2, \ldots, s_N) = -\mu(h + \Delta h)\sum_i s_i$$

The above energy function models a system with independent spins, where the spins are being influenced by a static site–dependent field and a mean field due to its neighbors. Note that this analysis is called the mean field approximation because the effects of the field due to the neighboring sites are reduced to a mean value of the field. Fluctuations away from the mean value of the field are being ignored, which leads to a model of a system where all the spins are independent. The partition function for the system with this energy function can simply be written as:

$$\begin{aligned} Q_{MF} &= \sum_{s_1, s_2, \ldots, s_N} \prod_{i=1}^{N} \exp[-\beta\mu(h + \Delta h)s_i] \\ &= \{2\cosh[\beta\mu(h + \Delta h)]\}^N \end{aligned}$$

The average spin, $< s_i >_{MF}$, at the ith site is then given as:

$$< s_i >_{MF} = \tanh[\beta\mu(h + \Delta h)]$$

which is site independent. The variational treatment comes into the picture when one has to estimate the optimal Δh, which can be seen as the parameter of that model. Following the treatment in Chandler [24], it can be easily shown that the optimal Δh is:

$$\Delta h = 2J < s_1 >_{MF} / \mu$$

The mean field approximations will be utilized in section 7.2.2.1, chapter 7, for the incorporation of 1D MRFs into our proposed minimax entropy registration framework.

3.5 Density Estimation

Estimating the joint and marginal density functions of pixel intensities from a set of sample values is key to the image registration framework developed in this thesis. Probability density estimation methods can be classified into two categories, *parametric* and *non–parametric* density estimation methods. Duda *et al.* [37] provides an overview of various density estimation methods.

3.5.1 Parametric Methods

Parametric density estimation methods assume that the form (Gaussian, exponential, Rayleigh, etc.) of the density function to be estimated is known, with the only variables to be estimated being the parameters of the functions. Suppose that it can be reasonably assumed that $p_{\mathbf{x}}(x)$ is a normal density with mean μ and covariance matrix Ψ, this assumption simplifies the problem of estimating the function $p_{\mathbf{x}}(x)$ to that of estimating the parameters μ and Ψ. Although these density estimation methods seem restrictive, they become important for cases where the dimensionality of the feature vector $\mathbf{x}$ is large

and the available samples are too small. *Maximum likelihood* and *Bayesian* estimation methods are two popular methods for estimating the parameters. Maximum likelihood estimation methods assume that the parameters to be estimated are fixed unknowns whereas in the Bayesian estimation methods the parameters are viewed as random variables with known prior distributions.

3.5.2 Non–Parametric Methods

The parametric methods described in the previous section assume that the form of the underlying density function is known. All of the classical parametric densities are unimodal, but practical problems involve multi-modal density functions. One way to solve the problem is to estimate the density function $p_{\mathbf{x}}(x)$ using non–parametric density estimation methods.

3.5.2.1 Parzen Window Method

Duda *et al.* [37] provide an excellent overview of the Parzen window based density estimation. This method approximates a density function at a point by a linear combination of *window functions*, $\varphi(u)$, centered at the sample values. In general, the window function can be any valid density function, which satisfies:

$$\varphi(u) \geq 0, \qquad \forall u$$

and

$$\int \varphi(u)\, du = 1$$

The probability density, at a point x, can then be estimated using n samples:

$$p_n(x) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{V_n} \varphi\left(\frac{x - x_i}{h_n}\right)$$

where h_n is the length of a hypercube with volume V_n. A few points which are obvious from the above formulation is that the estimated density at point x, $p_n(x)$, is a function of the window function $\varphi(u)$ and its width. Since a window function is placed at each sample and the density estimate is a linear combination of these window functions, the Parzen window estimate of the density computes a neighborhood average of the samples, where the neighborhood of a point x is determined by the size of the window function.

The Parzen window method can be used to estimate any smooth density function; the smoothness requirement is obvious from the previous section. The construction of the density function is also simple: a window function is placed at each sample. However, evaluation of the density involves time complexity proportional to the size of the data. This method also requires computation of the parameters of the window function used, for example, the variance of a Gaussian window function.

The most common window functions are unimodal functions which decay quickly to zero away from the mean value. The zero mean Gaussian function, $g_\psi(u)$, with variance ψ, is the most popular window function. Using this window function, the Parzen window estimate of density function $p(x)$, using a set of samples a of size n, can be written as:

$$p_n(x, a) = \frac{1}{n} \sum_{x_a \in a} g_\psi(x - x_a)$$

The only parameter that needs to be estimated is the variance ψ. Maximum likelihood estimation can be used to estimate the variance ψ. However, a procedure which minimizes

the entropy of the density function is utilized. To estimate empirical entropy of the density $p_n(x, a)$ two sets of samples a and b, of sizes N_a and N_b respectively, are selected at random. If only a single sample of data is used to estimate both the density function and the entropy of the distribution, then the variance ψ will tend to zero. This will be so because a density function with delta functions at each sample has minimum entropy. Thus, the Parzen window estimate, with Gaussian window of variance becoming zero, will effectively put a delta function at each sample to estimate the density function. Using two sets of samples avoids this degenerate case. The reason for this can be seen as follows. One set of samples are used to estimate the density function. The other set of samples are then used to estimate the entropy of this distribution. If the variance of the Gaussian window functions tends to zero, then the empirical entropy will increase towards positive infinity. However, we are estimating the variance which minimizes the entropy, hence avoiding the degenerate case of zero variance.

An important technical note. *A very important technical point to note in non–parametric estimation of density function using Parzen window method is that the estimated density function, in the limit, is independent of the variance of the Gaussian kernel and for that matter, is even independent of the kernel used in estimating the density function [37]. Thus, if a density function is estimated from a sample of pixels its entropy is independent of the variance of the Gaussian kernels used to estimate the density function. However, for a finite sample size, the entropy of the estimated density function will vary with the variance of the Gaussian kernel. Thus, the variance of the Gaussian kernel which minimizes the entropy of the estimated density function is chosen to be the optimal variance.*

3.6 Mutual Information

Papoulis [83] provides an excellent overview of probability and stochastic processes with some introduction to information theory. Cover and Thomas [30] and Reza [94] are excellent books on the subject. This section gives a brief overview some of the concepts related to the work in this thesis.

3.6.1 Entropy

Entropy as a measure of information was first proposed by Shannon [103]. Assume that a random variable $\mathbf{x}$ is distributed according to the probability mass function $P_{\mathbf{x}}(x)$. If an experiment is carried out, where the output is governed according to $P_{\mathbf{x}}(x)$, then the information generated by the experiment, for each output, is determined by the entropy of the distribution $P_{\mathbf{x}}(x)$. Said another way, entropy measures the uncertainty in the output value generated by the experiment. Starting with three basic axioms, Shannon [103], derived a mathematical measure of the entropy, H(x), of a probability distribution to be:

$$H(x) = -\sum_{x} P_{\mathbf{x}}(x) \ \ln P_{\mathbf{x}}(x)$$

It should be noted that even though entropy is conventionally written as $H(x)$, it is a functional, a function of a probability mass function. Many other definitions of entropy have been proposed but the above one remains the most popular and useful [58, 59]. Following are some of the definitions used in the thesis.

Definition 3.1: *The joint entropy, H(x,y), of random variables* $\mathbf{x}$ *and* $\mathbf{y}$ *distributed according to the joint probability mass function* $P(x, y)$ *is defined as:*

$$H(x, y) = -\sum_x \sum_y P(x, y) \ \ln P(x, y)$$

Definition 3.2: *The Kullback–Leibler distance,* $D(P\|Q)$*, between two probability mass functions* $P(x)$ *and* $Q(x)$ *is defined as:*

$$D(P\|Q) = \sum_x P(x) \ \ln \frac{P(x)}{Q(x)}$$

It can be easily shown that $D(P\|Q) \geq 0$, with equality if and only if $P(x) = Q(x)$, $\forall x$. However, $D(P\|Q) \neq D(Q\|P)$ in general and it does not satisfy the triangle inequality of distance metrics. Thus, Kullback–Leibler distance is not a distance between probability distributions and to make this distinction, it is also called, *Kullback–Leibler divergence.*

Definition 3.3: *Mutual Information between random variables* $\mathbf{x}$ *and* $\mathbf{y}$*, distributed according to the joint probability mass function* $P(x, y)$ *and marginal probability mass functions* $P(x)$ *and* $P(y)$*, is defined as:*

$$I(x, y) = D(P(x, y)\|P(x)P(y)) = \sum_x \sum_y P(x, y) \ \ln \frac{P(x, y)}{P(x)P(y)}$$

Positivity of mutual information follows from the positivity of the Kullback–Leibler divergence. Usually, mutual information is written in terms of entropies as:

$$I(x, y) \quad = \quad H(x) - H(x|y)$$

$$= \quad H(x) + H(y) - H(x, y)$$

3.6.1.1 Differential Entropy

While most of the definitions and results for continuous random variables follow those for the discrete random variables, there remain some important differences which warrant careful observation. To stress differences between the continuous and the discrete random variables, the entropy of the continuous random variable is also called *differential entropy.*

Definition 3.4: *The differential entropy $h(x)$ of a continuous random variable* $\mathbf{x}$ *with a probability density function $p(x)$ is defined as:*

$$h(x) = -\int p(x)\ \ln p(x)\ dx$$

Differential entropy provides only a relative measure of randomness, that is if $h(x) > h(y)$, then the random variable x is more unpredictable than the random variable y. However, for some density functions, the differential entropy can even be negative.

Example 3.1: *If x is uniformly distributed on an interval [0,$\frac{1}{8}$], then the differential entropy of the random variable, $h(x)$, is -3.*

The differential entropy of a discrete random variable can be considered to be negative infinity. Definitions and properties of the mutual information and Kullback–Leibler divergence are the same as those for discrete random variables.

3.6.2 Relation to Correlation Function

To see the relation between correlation coefficient and mutual information, consider two images I_1 and I_2 which are to be registered. Image I_2 is transformed, according to the

transformation parameters T, yielding image I_T. Assume that the random variable $\mathbf{x}$ denotes the distribution of the pixel intensities of image I_1 and the random variable $\mathbf{y}$ denotes the distribution of the pixel intensities of the transformed image, I_T. Further assume that the random variables $\mathbf{x}$ and $\mathbf{y}$ are jointly Gaussian distributed, that is,

$$p_{\mathbf{x}}(x) = \frac{1}{\sigma_x\sqrt{2\pi}} \exp\left(\frac{-1}{2}\frac{(x-\mu_x)^2}{\sigma_x^2}\right)$$

$$p_{\mathbf{y}}(y) = \frac{1}{\sigma_T\sqrt{2\pi}} \exp\left(\frac{-1}{2}\frac{(y-\mu_T)^2}{\sigma_T^2}\right)$$

Note that the mean, μ_T, and variance, σ_T, of $\mathbf{y}$ are a function of the transformation parameters T. Let

$$r_T = \frac{E\{(x-\mu_x)(y-\mu_T)\}}{\sigma_x\sigma_T}$$

be the correlation coefficient of $\mathbf{x}$ and $\mathbf{y}$. The joint Gaussian density function can now be written as:

$$p_{\mathbf{x},\mathbf{y}}(x,y) = \frac{1}{2\pi|\Psi|^{1/2}} \exp\left[-\frac{1}{2}(x-\mu_x \quad y-\mu_T)\Psi^{-1}\begin{pmatrix} x-\mu_x \\ y-\mu_T \end{pmatrix}\right]$$

where Ψ denotes the covariance matrix of the joint density function, given as,

$$\Psi = \begin{pmatrix} \sigma_x^2 & r_T\sigma_x\sigma_T \\ r_T\sigma_x\sigma_T & \sigma_T^2 \end{pmatrix}$$

and

$$|\Psi|^{1/2} = \sqrt{\mathrm{Det}\Psi} = \sigma_x \sigma_T \sqrt{1 - r_T^2}$$

If, $I(x, y)$ denotes the mutual information between $\mathbf{x}$ and $\mathbf{y}$, then after some arithmetic, see appendix A, it can be shown that:

$$I(x, y) = -\ln \sqrt{1 - r_T^2} \tag{3.2}$$

Thus, for this simple example, maximization of mutual information, $I(x, y)$ to bring I_1 and I_2 into registration is equivalent to maximization of r_T. However, if we define sensitivity of a registration measure with respect to transformation parameters as a small change in value for a small change in transformation parameter, then differentiating equation (3.2) by transformation parameters T, we have,

$$\frac{dI(x, y)}{dT} = \left(\frac{r_T}{1 - r_T^2} \right) \frac{dr_T}{dT}$$

This relation shows that a small change in transformation parameters, close to the true registration, leads to a large change in the mutual information measure as compared to the change in the correlation coefficient. Thus, it can be concluded that even for the registration of same modality images, the mutual information measure is more sensitive to registration as compared to the correlation coefficient.

3.6.3 Empirical Estimation of Entropy

In this thesis, non–parametric Parzen window estimates will be used to estimate both the marginal and the joint probability density functions. The analysis will closely follow the work of Viola *et al.* [117]. Let X denote the *reference image* and Y denote the transformed test image. Let $\mathbf{x}$ and $\mathbf{y}$ be the random variables denoting the distribution of pixel intensities in the reference and the transformed images respectively. Parzen window estimates are used to estimate the two density functions. Let $g_\psi(\cdot)$ be the Gaussian kernel, with zero mean and variance ψ, used in the density estimation of $\mathbf{y}$. Assume that a set a of samples, of size N_a, is selected at random from the image Y. Then the Parzen window estimate of the density of $\mathbf{y}$ is:

$$p_{\mathbf{y}}(y, a) = \frac{1}{N_a} \sum_{y_a \in a} g_\psi(y - y_a)$$

To estimate the entropy of distribution $p_{\mathbf{y}}(y, a)$ another set of samples b, of size N_b, is selected from the image Y and the entropy, H(y), is estimated as a sample mean as follows:

$$\begin{aligned} H(y) &= E_b \left\{ \ln \left(p_{\mathbf{y}}(y, a) \right) \right\} \\ &= \frac{1}{N_b} \sum_{y_b \in b} \ln p_{\mathbf{y}}(y_b, a) \end{aligned}$$

A stochastic gradient descent method is used as the optimization technique to estimate the optimal parameters. Viola [116] provides a convergence analysis of the descent algorithm for this problem.

3.6.4 Maximum Entropy Principle

Jaynes [58, 59] proposed that the maximum entropy principle forms a fundamental principle both in statistical mechanics and probability theory. Jaynes [60] showed that, starting with Shannon's definition of entropy, the maximum entropy principle can be used to derive the Bernoulli distribution law for the distributions of successes in repeated experiments.

The maximum entropy principle aims at estimating a density function of a random variable, $\mathbf{x}$, under specified constraints which has the maximum entropy. This problem is formulated using variational calculus where the function to be estimated is a density function. The following example will clarify the basics of the principle.

Example 3.2: *Suppose that density function $p(x)$ of the random variable $\mathbf{x}$ is to be estimated under the constraints*

$$\int p(x)\, dx = 1$$

and n specified values η_i, expected values of known functions $g_i(x)$:

$$E\{g_i(x)\} = \int g_i(x)\, p(x)\, dx = \eta_i; \qquad i = 1, \ldots, n$$

Thus, the Lagrange multipliers method can be used to set up the calculus of variations problem as follows. Estimate $p(x)$ which maximizes

$$\begin{aligned} &-\int p(x)\ \ln p(x)\, dx - (\lambda_0 - 1)\left(\int p(x)\, dx - 1\right) \\ &-\lambda_1\left(\int g_1(x)\, p(x)\, dx - \eta_1\right) - \cdots - -\lambda_n\left(\int g_n(x)\, p(x)\, dx - \eta_n\right) \end{aligned}$$

Using the calculus of variation, it can be shown that $p(x)$ is an exponential distribution:

$$p(x) = \exp\{-\lambda_0 - \lambda_1 g_1(x) - \cdots - \lambda_n g_n(x)\}$$

The Lagrange multipliers are estimated by solving the set of constraint equations.

Following is another example [16] which shows that the maximum entropy method (MEM) principle is a fundamental principle.

Example 3.3: *Estimate the discrete probabilities p_i for $i = 1, \ldots, N$ when it is known that these probabilities lie close to some prior knowledge r_i, which is a valid probability mass distribution. To estimate p_i, we maximize the function h as a function of G:*

$$h(p) = -\sum p_n \ \ln p_n + G(p, r)$$

To find the optimal p_n differentiate h with respect to each p_n and set it to zero.

$$\frac{\partial h}{\partial p_n} = -(1 + \log p_n) + \frac{\partial G}{\partial p_n} = 0 \qquad \textit{when } p = r$$

A non–trivial function which satisfies the above relation is

$$G(p, r) = p_n + p_n \ln r_n + G_1(p_1, \ldots, p_{n-1}, p_{n+1}, \ldots, p_N, r)$$

Repeating this for other p_i yields

$$G = \sum_{n=1}^{N} (p_n + p_n \ \ln r_n) = 1 + \sum_{n=1}^{N} p_n \ \ln r_n$$

Thus, the objective to maximize to estimate p_i's then becomes

$$h(p) = -\sum_{n=1}^{N} p_n \ \ln \frac{p_n}{r_n}$$

where we have ignored the constant 1*. Note that*

$$\frac{\partial^2 h}{\partial p_n^2} = -\frac{1}{p_n} \leq 0 \ \forall n$$

Thus, $h(p)$ has one maximum at $\frac{\partial h}{\partial p_n} = 0$, which subject to the integration constraint $\sum p_n = 1$ yields $p_n = r_n \ \ \forall n$.

In retrospect, the solution to *example 3.3* seems obvious from the Kullback–Leibler divergence between two density functions. However, this example shows that the principle of maximum entropy is more fundamental principle, where the Shannon's definition of entropy is utilized. The principle of maximum entropy leads to estimation of a probability density function $p(x)$ which satisfies the consistency axioms: (1) uniqueness of the solution, (2) invariance to choice of coordinate system, (3) system independence, and (4) subset independence.

Consider an example where only L of the N values of r_k are specified.

Example 3.4: *Suppose the density function $p(x)$ of a random variable* $\mathbf{x}$ *is to be estimated where only L of the N values of r_k are specified, with the remaining M*

unknown. From the previous example, we have,

$$G(\overline{p}, \overline{r}) = \sum_{l=1}^{L} (p_l + p_l \ \ln r_l)$$

This gives the function h' to be maximized to be

$$\begin{aligned} h' &= -\sum_{l=1}^{L} p_l \ \ln p_l + \sum_{l=1}^{L} (p_l + p_l \ \ln r_l) - \sum_{m=L+1}^{N} p_m \ \ln p_m \\ &= -\sum_{l=1}^{L} p_l \ \ln \left(\frac{p_l}{r_l} \right) + \sum_{l=1}^{L} p_l - \sum_{m=L+1}^{N} p_m \ \ln p_m \end{aligned}$$

Taking second derivatives of h' with respect to p_k yields

$$\frac{\partial^2 h'}{\partial p_k^2} = -1/p_k$$

for all l and m and so the function h' has only one maximum. The maxima is found by solving the equations $\frac{\partial h'}{\partial p_k} = 0$ for each l such that $1 \leq l \leq L$. Solving these equations yields, as in the previous example, $p_l = r_l$. To solve for the remaining values of p_m for $L < m \leq N$, equations $\frac{\partial h'}{\partial p_m} = 0$ are solved subject to the constraint

$$\begin{aligned} \sum_{l=1}^{L} p_l + \sum_{m=L+1}^{N} p_m &= \sum_{l=1}^{L} r_l + \sum_{m=L+1}^{N} p_m \\ &= 1 \end{aligned}$$

Solving these equations yields $p_m = A$, a constant, which evaluates to

$$A = \frac{1 - \sum_{l=1}^{L} r_l}{M}$$

by using the previous equations.

Example 3.4 shows that the maximum entropy principle is a general formulation to the problem of probability density estimation.

3.7 Summary

This chapter briefly described various mathematical methods that are used to develop the registration framework presented in this thesis. The interested reader is pointed to various sources which cover every topic in more detail.

Chapter 4

Segmentation–Integrated Registration

4.1 Introduction

Portal images acquired during the EBRT treatment of the prostate cancer are noisy and are of low contrast. Thus, accurate patinet setup verification using portal images is a difficult task. However, since we already have a high quality treatment planning 3D CT image dataset, acquired using diagnostic energy X–rays, we utilize this information to help segment some structure from the portal images and then use this estimated segmentation to help better estimate the registration parameters. In this chapter we outline the integrated registration framework for patient setup verification in EBRT of prostate cancer where the segmentation of the portal image is carried out simultaneously and iteratively. As mentioned in Chapter 2, the registration methods proposed in the literature can be broadly classified into two categories of registration methods, the sparse–field or feature–based and dense–field. The sparse field methods typically aim at extracting

sparse feature information from the given image data to be registered. The registration of two images is then accomplished by registering the two feature sets. Usually, the feature extraction process involves first or higher order derivatives of the intensity images. For example, edge detection might require second order derivatives of the image. Hence, the accuracy of the feature–based methods is sensitive to the noise in the images. However, if the features have been accurately detected and localized, these methods can accurately estimate the registration parameters. The dense field methods, for example gray scale correlation, directly manipulate image pixel intensities to estimate the registration parameters. As these methods do not, in general, require evaluating higher order derivatives of the images, these methods tend to be robust to noise. However, these methods are computationally more expensive as compared to the feature based methods. Feature based methods were usually preferred for multi–modality image registration because if homologous features can be extracted from the images then the two image datasets can be registered irrespective of the modalities used to acquire the image data. However, recently a mutual information [117, 28] based dense field method has been successfully used to register multi–modal images.

Both of these strategies have some merit and some problems. There are fundamental tradeoff between the two strategies which are relevant to the registration approach presented in this thesis:

- Accurate feature identification–based and/or portal object segmentation–based strategies are good for registration localization and accuracy, but are sensitive to noise and blur.

- Intensity–based strategies are typically poor in localization accuracy, but will be more robust to noise and blur.

In this thesis we propose that the segmentation and the registration of portal images to the 3D CT data should be carried out simultaneously and iteratively. The framework is based on the intuition that if we have a rough estimate of the segmentation of a portal image, then it can be used to better estimate the registration parameters. Estimated registration parameters can then, in turn, be used to better segment the portal image. The iterative registration framework is of special interest for the registration of portal images to 3D CT data set as the portal images are usually of poor quality. Usually, the 3D CT dataset is of high quality data. Thus, the key idea is to utilize the information from 3D CT to help better segment the portal images. A better segmentation of the portal images can then be used to better register portal image and 3D CT image data. However, the 3D CT information can possibly guide segmentation of the portal image only if it is in registration in the first place, making this a chicken–and–an–egg problem.

The next section mathematically defines the image registration problem. It should be noted that we are primarily interested in estimating the registration parameters. Next the problem is formulated as a maximum a–posteriori (MAP) estimation problem where both the segmentation labels and the registration parameters are being explicitly estimated. We note some of the difficulties with such an approach and propose a maximum likelihood (ML) based approach. The EM algorithm–based formulation is then presented to estimate the maximum–likelihood registration parameters. We note that this approach overcomes some of the weaknesses of the proposed MAP formulation. However, for our problem, the EM formulation has some weaknesses, which leads to the information theoretic *minimax entropy* registration framework proposed in the following chapter.

4.2 Problem Definition

Let, $\mathbf{X}$= $\{\mathbf{x}(\mathrm{i})\}$, for $i = 1, \ldots, N^2$ denote the $N \times N$ random field from which the portal images are sampled. Let, $\mathbf{G}$= $\{\mathbf{g}(\mathrm{i})\}$, for $i = 1, \ldots, N^3$ denote the random field from which 3DCT images are sampled. Let $\mathbf{Y}(T) = \{\mathbf{y}(\mathrm{i},\, T)\}$ for $i = 1, \ldots, N^2$ denote the $N \times N$ random field from which the projections from the 3DCT data set are sampled, at a given set of transformation parameters $\mathbf{T} = T$. In the following we will assume a rigid transformation and hence $\mathbf{T}$ represents a set of six parameters, three translations (t_x, t_y, t_z) and three rotations (θ_{XY}, θ_{XZ}, θ_{YZ}). The projected 3DCT images are also called the digitally reconstructed radiographs. Note that we index pixels in the images using a single index, even though the images are 2D (or 3D) images. For simplicity in the formulations below, we have assumed that the portal and the projected 3DCT are of the same dimensions. However, this need not be true. We further assume that the pixels, for all the random fields, are independently distributed (This assumption will be later relaxed in chapter 7). Thus, the probability density function of the random field $\mathbf{X}$ can be written in factored form as $p_{\mathbf{X}}(X) = \prod_i p_{\mathbf{x}_i}(x_i)$. Since the only structure visible in the portal images is the bone, each pixel is being classified as belonging to one of two classes, bone or no–bone (everything else), and the pixel intensities in each class are assumed to be identically distributed according to the probability density function of its class. Since the pixels are assumed to be independently distributed, we can determine the random field from which the portal images are sampled. Note that for notational simplicity, we shall now write $\mathbf{x}(i)$ as $\mathbf{x}_i$ and $\mathbf{y}(i, T)$ as $\mathbf{y}_i$, where the current transformation parameters for which a DRR is obtained should be clear from the context.

Segmentation information is incorporated into the problem by considering the joint density function $p(x_i, y_i)$ as a mixture density (see section 3.3.4 in the last chapter).

To set up the joint density function into a mixture density form, let $A = \{$*bone, no–bone*$\}$ $= \{$*1,2*$\}$, denote the set of classes. Note that the set of classes can also be denoted by a set of vectors, $\{(0,1),(1,0)\}$. Such a representation allows us to introduce the concept of indicator variables, where the indicator variables are primarily used to simplify the mathematics. Since X–rays pass through both bone and tissue before hitting the portal image, a clarification of the classes *bone* and *no–bone* is necessary. We seek to classify pixels in the portal image as *bone* for which the X–rays passed through at least some bone tissue. The other pixels we seek to label *no–bone*. Let, $\mathbf{M}= \{\mathbf{m}(\mathrm{i})\}$, for $i = 1, \ldots, N^2$ denote the $N \times N$ random field on the segmentation of the portal images.

Let $\mathbf{Z}$ be the $N^2 \times 2$ *classification matrix*, with each row $\mathbf{z}_i$ of the matrix defining a set of random variables $\mathbf{z}_i = (\mathbf{z}_{1i}, \mathbf{z}_{2i})$, defined to be:

$$\mathbf{z}_{1i} = \begin{cases} 1, & \text{if } \mathbf{m}_i = \textit{bone} \\ 0, & \text{if } \mathbf{m}_i = \textit{no–bone} \end{cases}, \quad \mathbf{z}_{2i} = \begin{cases} 0, & \text{if } \mathbf{m}_i = \textit{bone} \\ 1, & \text{if } \mathbf{m}_i = \textit{no–bone} \end{cases}$$

The expected values of the random variables $\mathbf{z}_{ai}, \forall a$, denoted by $<z_{ai}>= P(\mathbf{m}_i = a)$, satisfy the constraint, $\sum_{a \in A} <z_{ai}>= 1$. Note that the random variables $\mathbf{z}_{1i}, \mathbf{z}_{2i}$ are negatively correlated random variables, with the random variable $\mathbf{z}_i$ taking only two possible values, $\{(0,1),(1,0)\}$.

We first pose our problem in a maximum a–posteriori (MAP) framework where both the segmentation, $\mathbf{M}$, and the transformation parameters, $\mathbf{T}$, are being estimated explicitly. As noted earlier, we want to estimate a segmentation of the portal image in order to help the registration process. The problem of segmenting the portal image is formulated as a labeling problem in which each pixel is labeled either as bone or no–bone. To incorporate the segmentation information, the joint density function, $p_{\mathbf{x},\mathbf{y}}(x,y)$, is written

as a mixture density where the labels on the portal image pixels are treated as the missing information. Note that MAP leads to a particular estimate of the segmentation labels, **M**, and the transformation parameters, **T**. However, we note some restrictions with this approach for our problem and thus we propose a maximum likelihood (ML) framework to overcome these restrictions. One approach for computing the MLE from incomplete data is the expectation–maximization (EM) algorithm [32]. Our first thoughts were to select the EM algorithm to compute the estimates due to its proven monotonic convergence properties, ease of programming and the fact that it does not require computation of the Hessian. Also, unlike other optimization techniques, there is no need to set a step–size parameter. Thus, the EM algorithm is used to estimate the pose parameters which also leads to an estimate of the posterior probability of the label at each pixel. However, for our problem, the EM approach has several restrictions too, as noted below. These problems lead us to propose our new minimax entropy strategy which is described in the next chapter.

4.3 Maximum A–Posteriori Formulation

An estimate of the segmentation of the portal image, **M**, can be used to help estimate pose, **T**, of the 3DCT dataset. One straight forward way to capture this strategy would be to set up a MAP (Maximum A–Posteriori) strategy as follows:

$$(\hat{T}, \hat{M}) = \arg\max_{T,M} p(T, M|X, G) \tag{4.1}$$

Taking the logarithm of the expression and assuming that the pose parameters, **T**, are uniformly distributed and that **G** is independent of **T** and **M**, equation (4.1) can be

simplified, using Bayes' rule, as follows:

$$\begin{aligned}(\hat{T}, \hat{M}) &= \arg\max_{T,M} \ln p(T, M|X, G) \\ &= \arg\max_{T,M} \ln \left(\frac{p(X, G|T, M)\ p(T, M)}{p(X, G)} \right)\end{aligned}$$

Ignoring $p(X, G)$ as this is constant for a given data set,

$$(\hat{T}, \hat{M}) = \arg\max_{T,M} \ln \left(p(X, G|T, M)\ p(T, M) \right)$$

As we are not using prior information on **T** and **M**, uniform priors can be assumed. Thus, ignoring $p(T, M)$, we can further write:

$$\begin{aligned}(\hat{T}, \hat{M}) &= \arg\max_{T,M} \ln p(X, G|T, M) \\ &= \arg\max_{T,M} \ln \left(p(X|G, T, M)\ p(G|T, M) \right)\end{aligned}$$

Now, since **G** is independent of **T** and **M**, and $p(G)$ is a constant, we can reduce the above equation further to:

$$\begin{aligned}(\hat{T}, \hat{M}) &= \arg\max_{T,M} \ln p(X|G, T, M) \\ &= \arg\max_{T,M} \{\ln p(X|Y(T), M)\} \qquad (4.2)\end{aligned}$$

because X depends on G through projected digitally reconstructed radiograph $Y(T)$. The portal image pixel intensities are not Gaussian distributed. For the development of the algorithm, we will model the joint density function, $p(X, Y)$, as a mixture den-

sity, consisting of two component densities. One component density function models the intensities of the no–bone pixels and the other component density function models the intensities of the bone pixels. The joint density function, $p(x_i, y_i)$, can thus be written in terms of the component density functions as (where P denotes probability and p denotes probability density function):

$$\begin{aligned} p(x_i, y_i) &= \sum_{a \in A} P(\mathbf{m}_i = a)\; p(x_i, y_i | \mathbf{m}_i = a) \\ &= \sum_{a \in A} P(\mathbf{m}_i = a)\; p_a(x_i, y_i) \\ &= \sum_{a \in A} P_i(a)\; p_a(x_i, y_i) \end{aligned}$$

where $\mathbf{m}_i$ is the random variable denoting the label at the ith pixel, and $p_a(x_i, y_i)$ is the joint density function of the pixel intensities, also called the *component* density function, given that the ith portal image pixel is labeled a. To simplify the notational complexity, we abbreviate $p_{\mathbf{x}_i \mathbf{y}_i}(x_i, y_i | \mathbf{m}_i = a)$ by $p_a(x_i, y_i)$ and similarly for other probability density functions (pdfs), where the random variables in the abbreviated form should be clear from the context.

Starting from equation (4.2), and assuming that the pixel intensities are statistically independent, we can write:

$$\begin{aligned} (\hat{T}, \hat{M}) &= \arg\max_{T,M} \{\ln p(X | Y(T), M)\} \\ &= \arg\max_{T,M} \left\{ \sum_i \ln p(x_i | y_i, m_i) \right\} \end{aligned} \tag{4.3}$$

This follows from the fact that under the statistical independence assumption the joint conditional density $p(X|Y(T), M)$ can be written in factored form as:

$$\begin{aligned} p(X|Y(T), M) &= \prod_i p(x_i|y_i(T), m_i) \\ &= \prod_i p(x_i|y_i, m_i) \end{aligned}$$

Now consider the conditional probability $p(x_i|y_i, m_i)$ at the ith pixel.

$$\begin{aligned} p(x_i|y_i, m_i) &= \frac{p(x_i, y_i|m_i)}{p(y_i|m_i)} \\ &= \frac{p(x_i, y_i|m_i)}{p(y_i)} \end{aligned}$$

where we assume that the pixel intensities of the DRR are independent of the portal image segmentation label. This assumption is true especially when the two images are not registered. Even when the algorithm converges, for real images some errors in the estimated registration parameters and the estimated segmentation are expected (as the DRR and the portal image are usually of different sizes). Hence, we felt it safe with the independence assumption. Thus, under this assumption we have,

$$\begin{aligned} p(x_i|y_i, m_i) &= \frac{p(x_i, y_i|m_i)}{p(y_i)} \\ &= \frac{\prod_{a \in A}[p(x_i, y_i|\mathbf{m}_i = a)]^{z_{ai}}}{p(y_i)} \\ &= \frac{\prod_{a \in A}[p_a(x_i, y_i)]^{z_{ai}}}{p(y_i)} \end{aligned}$$

or, taking logarithm on both sides of the equation yields,

$$\ln p(x_i|y_i, m_i) = \sum_{a \in A} z_{ai} \ln p_a(x_i, y_i) - \ln p(y_i) \tag{4.4}$$

Thus, from equations (4.3) and (4.4), the MAP formulation for estimating **M** and **T** becomes:

$$(\hat{T}, \hat{M}) = \arg\max_{T,M} \left\{ \sum_i \left(\sum_{a \in A} z_{ai} \ln p_a(x_i, y_i) - \ln p(y_i) \right) \right\} \tag{4.5}$$

Equation (4.5) can be solved to estimate both the optimal transformation parameters and the segmentation of the portal image.

4.3.1 Drawbacks

While the MAP strategy brings out the rationale of our approach, we feel that such a strategy is limiting for our purposes for three reasons. First, this approach requires the algorithm to estimate a unique segmentation of the portal image, for a unique estimate of the pose parameters **T**. An estimated segmentation will effect the estimate of the pose parameters. Since we feel that the true segmentation of the portal image, in general, cannot be accurately estimated, we prefer not to commit the algorithm to find an optimal transformation parameters for a particular segmentation. Second, segmentation labels are discrete variables whereas the pose parameters are effectively continuous parameters. Solving for both, discrete variables and continuous variables, will be difficult and computationally expensive. Once we have a good estimate of the pose parameters the segmentation labels can be easily estimated and so we propose estimation of pose parameters only. Third, this formulation assumes that the component density functions,

$p_a(x_i, y_i)$, and the marginal density function $p(y_i)$ are known.

Given these limitations, we instead pose our problem in a maximum–likelihood framework, with segmentation labels are viewed as hidden variables. Such an approach *could* be captured using an expectation–maximization (EM) algorithm. We will describe the limitations of this approach here and propose our information theoretic minimax entropy registration framework in the next chapter.

4.4 Maximum Likelihood Formulation

The ML estimate of the pose parameters can be formulated as:

$$\begin{aligned} \hat{T} &= \arg\max_{T} \ \ln p(T|X,G) \\ &= \arg\max_{T} \ \ln\left(\frac{p(X,G|T)\ p(T)}{p(X,G)}\right) \end{aligned} \tag{4.6}$$

Assuming uniform priors on the pose parameters $\mathbf{T}$ and ignoring the constant term $p(X,G)$, we have,

$$\begin{aligned} \hat{T} &= \arg\max_{T} \ \ln p(X,G|T) \\ &= \arg\max_{T} \ln\left(p(X|G,T)\ p(G|T)\right) \\ &= \arg\max_{T} \ln p(X|Y(T)) \\ &= \arg\max_{T} \ \left[\ln p(X,Y(T)) - \ln p(Y(T))\right] \\ &= \arg\max_{T} \sum_{i} \left[\ln p(x_i, y_i(T)) - \ln p(y_i(T))\right] \end{aligned} \tag{4.7}$$

where we ignore the term $p(G|T)$, since the 3DCT data set, $\mathbf{G}$, is statistically independent of the pose parameters, $\mathbf{T}$. In equation (4.7) we assume that the image pixels are independent. The logarithm of the likelihood function is taken to simplify the mathematical formulation. Note that we first formulate the problem of pose estimation as a MAP problem which, however, is reduced to ML estimation problem by assuming uniform priors on the pose parameters.

Maximum–likelihood estimation of the transformation parameters, as formulated above, is ideally suited for the EM framework.

4.4.1 EM Formulation

Using a mixture density model, we can now write the joint mixture density model, from equation (4.7), for the portal image and DRR as, at pixel i:

$$\begin{aligned} p(x_i, y_i) &= \sum_{a \in A} P(\mathbf{m}_i = a) \;\; p_a(x_i, y_i) \\ &= \sum_{a \in A} P_i(a) \;\; p_a(x_i, y_i) \end{aligned}$$

Using this notation, the two steps of the EM algorithm [32] (see section 3.3 of chapter 3) for the mixture model are written as, at the kth iteration (*see appendix B for the development of the two steps*):

<u>E–Step:</u>

$$Q(T, T^{(k-1)}) = \sum_i \left(\sum_{a \in A} <z_{ai}>^k \;\; \ln p_a(x_i, y_i) - \ln p(y_i) \right) \tag{4.8}$$

M–Step:

$$T^k = \arg \max_T Q(T, T^{(k-1)}) \tag{4.9}$$

where,

$$<z_{ai}>^k = \left(\frac{<z_{ai}>^{k-1} p_a(x_i, y_i)}{\sum_{b \in A} <z_{bi}>^{k-1} p_b(x_i, y_i)} \right) \tag{4.10}$$

where, $y_i = y(i, T^{(k-1)})$. Note that $<z_{ai}>^k$ can be easily shown to be the conditional probability, given the data set and the current estimate of the transformation parameters, that the ith pixel belongs to class a, that is, $P(\mathbf{m}_i = a | x_i, y_i)$.

Note that the maximum likelihood formulation overcomes some of the drawbacks of the MAP formulation. In each iteration, only the transformation parameters need to be estimated. The segmentation labels have been *integrated out* of the equation. However, the formulation also leads to an estimate of a–posteriori distribution on the segmentation labels.

4.4.2 Observations

For our purposes, the EM algorithm has some limitations as highlighted by two key observations. First, in the EM algorithm for the mixture model as formulated in equation (4.8), the function $Q(T, T^{(k-1)})$ is defined only if the component density functions $p_a(x_i, y_i)$, $\forall a$ are known (i.e. one should know whether they are Gaussian, Rayleigh, exponential etc.). The component density functions need to be determined for each specific problem. For multi-modal image registration it is difficult, at best, to know a–priori the joint density function between the pixel intensities in the two images. For example, in EBRT of

prostate cancer, the joint density functions need to be determined from the projection model, taking into account the physics of interaction of high energy X–rays with matter. If the joint density functions are not known, then usually a simplified model is assumed. In the next chapter we propose to estimate the unknown density functions from the given data instead of using simplified models. However, such an estimated density function cannot be directly used in the ML parameter estimation framework (see appendix C for details). The second observation comes from Neal *et al.* [79] who provide a view of the general EM algorithm as a coordinate descent algorithm where the coordinates axes are the joint density functions on the missing information and the parameters to be estimated. This insight into general EM as a coordinate descent algorithm is important (a similar view of the EM algorithm for parameter estimation of mixture density models has been given by Hathaway *et al.* [52]). In the EM framework, for the mixture density formulation, the prior probabilities on each pixel, $P_i(a)$, are required to be known. If these probabilities are not known, then they can also be estimated within the EM framework, assuming that the segmentation labels on each pixel are independently and identically distributed. For our problem this is definitely not the case.

In the next chapter we propose to overcome the restrictions of the EM algorithm by borrowing the idea of averaging over the estimated density function from mutual information. Mutual information as a match measure has been first proposed and successfully applied for multi–modality image registration by Viola *et al.* [117] and Collignon *et al.* [28]. However, the mutual information match measure formulation in the literature assumes that all the pixels in the image are independent and identically distributed (i.i.d.), an assumption not true in general. The minimax entropy algorithm described in the next chapter, proposes an information theoretic framework which aims at combining

the strengths of both the EM algorithm and the mutual information–based registration approach, to simultaneously segment and register images.

4.5 Summary

This chapter first formulated the problem of estimating the segmentation of the portal images and registration to the 3DCT image data in a maximum a–posteriori (MAP) framework where both the segmentation and the registration parameters are being explicitly estimated. However, as the *true* segmentation of the portal image is unknown, we do not want the estimated registration parameters to depend on a particular estimated segmentation of the portal image. Therefore, the image registration problem is posed in a maximum likelihood (ML) framework where only the transformation parameters are being explicitly estimated. In this framework, only a probability distribution on the portal image pixel labels is being evaluated. The ML registration parameters are then proposed to be estimated by using the EM algorithm. However, we note that the maximum likelihood estimates of the registration parameters require knowledge of the form of various joint and marginal density functions in the mixture model, which is almost never available, especially for multi–modal image registration. Taking cues from the formulation of the EM algorithm, the next chapter proposes our new minimax entropy registration framework.

Chapter 5

Minimax Entropy Algorithm

5.1 Introduction

In the last chapter, automatic and iterative segmentation and registration of 2D portal images to 3DCT data sets was first formulated as a *maximum a–posteriori* (MAP) problem where *both* a segmentation of the portal image and registration parameters were being explicitly evaluated. However, two drawbacks of such a strategy were pointed out, which lead to a *maximum likelihood* (ML) formulation of the problem where only the transformation parameters, $\mathbf{T}$, were explicitly estimated. To overcome the restrictions of the ML formulation, we propose the *minimax entropy* algorithm, developed in this chapter, for the estimation of both the transformation parameters,$\mathbf{T}$, and the segmentation of the portal image $\mathbf{M}$.

5.2 Minimax Entropy Formulation

The proposed minimax algorithm for solving the basic problem posed by equation (4.6), in a computational form similar to EM, has two steps, the *max* step and the *min* step,

which are evaluated iteratively to determine the registration parameters and the probability distribution of the portal image segmentation. This framework builds on the EM formulation of the ML estimates given by equations (4.8) and (4.10).

5.2.1 Formulating the MAX step

The *max* step estimates the probability distribution function $P(M)$. As mentioned in the previous chapter, the *E–step*, equation (4.10), in the EM algorithm evaluates the *conditional* probability distribution function, $P(M|X,Y)$. The principle of *maximum entropy* [58, 59], is invoked in the *max* step to estimate the distribution $P(M)$ under the given constraints.

Thus, given $P(M|X,Y)$, under the constraint $\sum_M P(M) = 1$, the *max* step estimates $P(M)$ which is *closest* to $P(M|X,Y)$ and has maximum entropy under given constraints. Thus, to estimate $P(M)$, we maximize the function h', as a function of Q:

$$h'(P(M)) = -\sum P(M) \ln P(M) + Q(P(M), P(M|X,Y))$$

Thus, following the formulation of Bevensee [16] and example 3.3, section 3.6.4 which determines the function Q, the *max* step can be concisely written as a follows:

Max Step:

$$P^k(M) \quad = \quad \arg \max_{P(M)} \left[-\sum_M P(M) \ln P(M) + \sum_M P(M) \ \ln P(M|X, Y(T^{(k-1)})) \right] \quad (5.1)$$

under the constraint $\sum_M P(\mathbf{M} = M) = 1$, where $\mathbf{M}$ is the random variable whose domain is the set of possible segmentations of the portal image, where each pixel can be labeled from the set of labels A. We assume that pixel labels are statistically independent, i.e.,

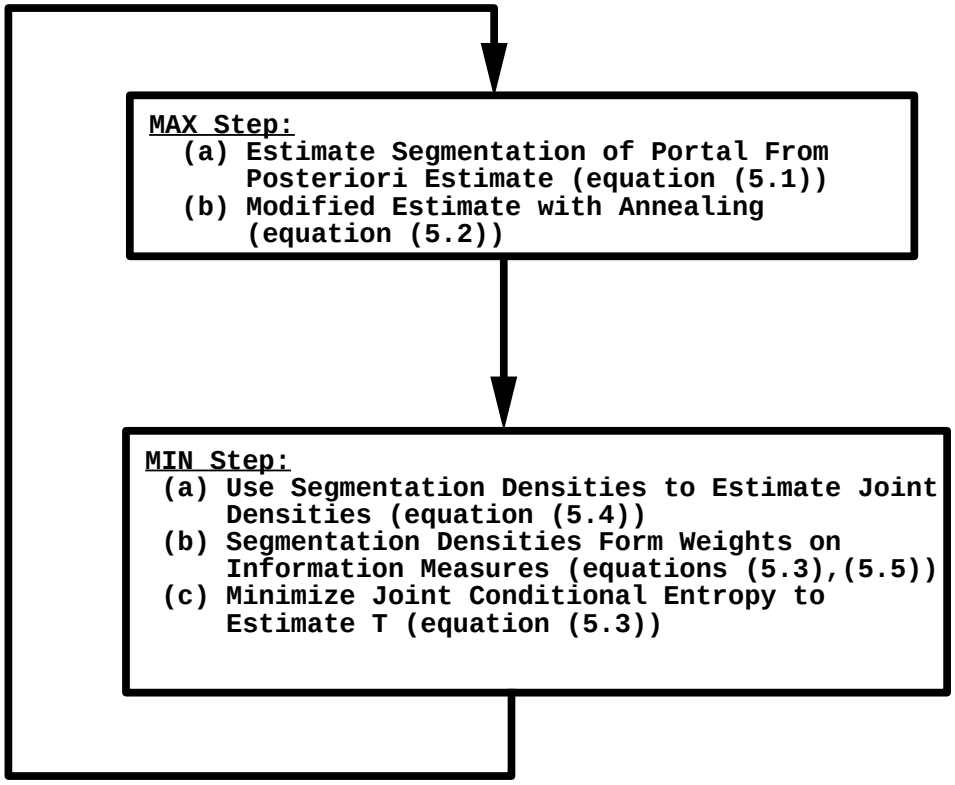


Figure 5.1: The Minimax Entropy Algorithm

$P(\mathbf{M} = M) = \prod_i \; P(\mathbf{m}_i = a) = \prod_i P_i(a)$. It can be seen that the *max* step above is simply minimizing the Kullback–Leibler (KL) divergence between the two distributions. As formulated above, the max–step simply states that the maximum entropy estimate of the probability $P(\mathbf{M} = M)$ is the posterior probability on the segmentation of the portal image, i.e $P(M|X, Y(T^{(k-1)}))$, given the current estimate of the transformation parameters, $T^{(k-1)}$, the portal image, X, and the DRR, Y [16]. This simple formulation of the estimated probability of a segmentation of the portal image allows us to systematically put constraints on the segmentation probability function, as we show below. The analytical solution to equation (5.1) estimates the probability of a segmentation label at

the ith pixel to be:

$$P_i^k(a) = \left(\frac{P_i^{k-1}(a) \;\; p_a^{k-1}(x_i, y_i)}{\sum_{b \in A} P_i^{k-1}(b) \;\; p_b^{k-1}(x_i, y_i)} \right)$$

where the component density functions, $p_a^{k-1}(x_i, y_i)$, are estimated in the next step.

Note that the $P_i^k(a)$'s, in the kth iteration, form the weighing terms in the Parzen window estimates, in equation (5.5) below, of the component density functions, $p_a(x, y)$. The component density functions, in turn, are used to estimate the joint entropies, $H_a(x, y) = -\int\int p_a(x, y) \;\; \ln p_a(x, y) \;\; dx \;\; dy$, which are minimized in the *min* step to estimate the registration parameters.

5.2.1.1 Modified Max Step

In order to better incorporate subjective information into the problem, an annealing schedule [62] is imposed on the estimated probability of a segmentation of the portal image pixel. The modified *max step*, equation (5.1), can thus be written as:

<u>*Modified Max Step*</u>:

$$P^k(M) \;\; = \;\; \arg \max_{P(M)} \left[-\frac{1}{\beta} \sum_M P(M) \ln P(M) + \sum_M P(M) \;\; \ln P(M|X, Y(T^{(k-1)})) \right] \quad (5.2)$$

under the constraint $\sum_M P(\mathbf{M} = M) = 1$, where $\beta = \frac{1}{t}$, and t is the *temperature* which determines the annealing schedule. The annealing schedule, as formulated above, can be understood in terms of estimation of the prior probability of the segmentation of the portal image (prior information should reflect our state of knowledge about the random variables [60]). Initially, for small values of β, irrespective of the conditional probability estimate, the above formulation incorporates our knowledge of total ignorance about the

pixel labels. This is essential as our initial estimate of the transformation parameters is expected to be poor and thus will lead to a poor estimate of the conditional density and the component density functions estimated from the given data set. However, as our confidence in the transformation parameters improves, the estimated component density functions will be *close* to true densities. Now the pixels can be classified into two classes using the *minimum–error–rate* classification, [37] (which is actually the *Bayes decision rule* for the zero–one loss function).

The connection to the Bayes decision rule for the zero–one loss function can be easily seen as follows. Using the annealing schedule formulation of the problem, equation (5.2), the probability that the label of the ith pixel is a, is given as

$$\begin{aligned} P_i^k(a) &= \frac{\left[P_i^{k-1}(a|x_i, y_i)\right]^\beta}{\sum_{b\in A}\left[P_i^{k-1}(b|x_i, y_i)\right]^\beta} \\ &= \frac{\left[P_i^{k-1}(a)\; p_a^{k-1}(x_i, y_i)\right]^\beta}{\sum_{b\in A}\left[P_i^{k-1}(b)\; p_b^{k-1}(x_i, y_i)\right]^\beta} \end{aligned} \tag{5.3}$$

According to the Bayes decision rule, to minimize the conditional risk using the zero–one loss function, the ith pixel should be classified as belonging to class a if the posterior probability for this class is greater that the posterior probability for the class b, that is,

Decide label a if $P_i(a|x_i, y_i) > P_i(b|x_i, y_i)$

which is equivalent to saying

Assign $P_i(a) = 1$ if $P_i(a|x_i, y_i) > P_i(b|x_i, y_i)$

If the two class posterior probabilities are the same, a pixel can be labeled as either class without increasing the conditional risk. Note that using an annealing schedule on the

segmentation probability density function, as proposed above, leads to such a classification of the pixel labels for large values of β.

Thus, the annealing schedule, as proposed above by using a simple term $\frac{1}{\beta}$, allows us to incorporate, analytically and systematically, the above discussed subjective information into the estimation of the probability of a segmentation of a portal image. Note also that the current estimate of the posterior probability is used in the *max* step thus to update our knowledge on the segmentation of the portal image. This updated knowledge of the portal image segmentation is then used as the prior information, in the next *max* step, for the estimation of the posterior probability. In summary, we are updating our information on the segmentation of the portal image using the only objective information we have, $P(M|X, Y(T^{(k-1)}))$. The subjective information about the *goodness* of this posterior probability is incorporated into the maximum entropy framework by using the $\frac{1}{\beta}$ term. Given this subjective and objective information about the segmentation of the portal image, the maximum entropy estimate of the probability is the only unbiased estimate of our state of knowledge, [60], about the segmentation of the portal image.

The estimated probability, $P(M)$, is the probability distribution for the ensemble of the portal images obtained from the patient, in a particular position. The reason why we can estimate such a distribution from a single portal image, rather than requiring a whole ensemble of images, is because we assume that the intensities of the pixels belonging to the class are distributed according to the same probability density function, that is, we are assuming a stationary random field. In other words, we assume that the intensities of the pixels of a class in a portal image are representative of the intensities at a particular pixel in the ensemble.

5.2.2 Formulating the MIN step

The EM algorithm requires knowledge of the form of the component density functions $p_a(x_i, y_i)$. To overcome this restriction, we propose estimating the component densities, $p_a(x_i, y_i)$, from the given data set at the current estimated transformation parameters (Note that we utilize the *Parzen* window method [37] for non–parametric density function). However, these estimated joint density functions cannot be used in the EM algorithm (see appendix C.1 for a discussion). Instead, we propose to estimate $p_a(x_i, y_i)$ from the given data, at the current estimate of the transformation parameters and then evaluate the expected value of $Q_i(T, T^{(k-1)})$ with respect to the density function, $p(x_i, y_i)$. An upper bound on the expected value, negative joint conditional entropy, $-H(M, X|Y)$, is then maximized (i.e. $H(M, X|Y)$ is minimized) to estimate $\mathbf{T}$ (see appendix C.2 for details).

Thus, the *min* step is developed to be:

<u>Min Step</u>:

$$\begin{aligned} T^k &= \arg\min_T H(M, X|Y) \\ &= \arg\min_T \left(\sum_{a \in A} \left(\frac{1}{N^2} \sum_{i=1}^{N^2} P_i^k(a) \right) H_a(x, y) - H(y) \right) \end{aligned} \tag{5.4}$$

The component density function for class a, $p_a(x, y)$, is estimated as the weighted sum of Gaussian kernels, $G_\psi(x) = (2\pi)^{\frac{-n}{2}} |\psi|^{\frac{-1}{2}} \exp(-\frac{1}{2} x^T \psi^{-1} x)$, using the Parzen window method as follows:

$$p_a^k(x, y) \approx \frac{1}{\sum_{(x_i, y_i) \in \mathbf{I}} P_i^k(a)} \sum_{(x_i, y_i) \in \mathbf{I}} P_i^k(a)\, G_{\Psi_a}(x - x_i, y - y_i) \tag{5.5}$$

where, $P_i^k(a) = P^k(\mathbf{m}_i = a)$ is the probability that the ith pixel in the portal image belongs to class a, estimated in the *max* step, equation (5.1), Ψ_a is a 2–by–2 covariance matrix, which is assumed to be diagonal (Note that this assumption does not means that the random variables $\mathbf{x}$ and $\mathbf{y}$ are independent). **I, J** denote sets of sizes N_I and N_J, respectively, of pixels sampled at random from the portal image, X, and the DRR, Y. The joint entropy functions, which are the expected value of the log of the joint probability density functions, are approximated as the statistical expectations using the Parzen window density estimates as follows:

$$\begin{aligned} H_a(x,y) &= -\int\int p_a(x,y)\ln p_a(x,y)\,dx\,dy \\ &\approx \left(\frac{-1}{\sum_{(x_j,y_j)\in\mathbf{J}} P_j^k(a)}\right) \\ &\quad \sum_{(x_j,y_j)\in\mathbf{J}} P_j^k(a)\ln\left(\frac{1}{\sum_{(x_i,y_i)\in\mathbf{I}} P_i^k(a)}\sum_{(x_i,y_i)\in\mathbf{I}} P_i^k(a)\,G_{\Psi_a}(x_j-x_i,y_j-y_i)\right) \end{aligned} \tag{5.6}$$

Similarly, the entropy of the DRR, $H(y)$, is estimated as:

$$\begin{aligned} H(y) &= -\int p(y)\ln p(y)\,dy \\ &\approx \left(\frac{-1}{N_J}\right)\sum_{y_j\in\mathbf{J}}\ln\left(\frac{1}{N_I}\sum_{y_i\in\mathbf{I}} G_\Psi(y_j-y_i)\right) \end{aligned}$$

It is easy to see the relation between the *min* step and the mutual information (MI) measure which is currently popular in the medical image analysis community. We have found MI to be more robust than other interesting measures (e.g. correlation. See appendix A where we present the relation between the two measures and show that the MI is more sensitive to mis–registration than correlation). While MI assumes that pixels are i.i.d., we

circumvent this assumption by using mixture densities. Studholme *et al.* [104], register images with mutual information as a match measure while incorporating segmentation information on one of the images. However, the image was pre–hand segmented and thus remains fixed throughout the registration. In our proposed algorithm, the portal image segmentation is estimated simultaneously and iteratively with the transformation parameter estimations. It should be noted that in our registration framework, the segmentation information is being utilized implicitly in the *min* step. In the initial iterations of the algorithm, when there is no information on the segmentation of the portal images, the algorithm effectively seeks to maximize the mutual information on the whole image. Later, when there is an estimate on the segmentation, the algorithm is effectively increasing the mutual information in the different segmented regions of the image.

5.2.3 Stochastic Gradient Descent Based Optimization

We follow the optimization strategy of stochastic gradient descent [117], which is described below and adapted for the inclusion of segmentation information. From equation (5.4), let the cost function be denoted by $F(T)$, that is,

$$F(T) = \sum_{a \in A} \left(\frac{1}{N^2} \sum_{i=1}^{N^2} P_i^k(a) \right) H_a(x, y) - H(y) \tag{5.7}$$

which is minimized to estimate the transformation parameters, given the current estimates of the segmentation of the portal image. The transformation parameters are updated according to the following rule, until a pre-specified number of iterations.

$$\mathbf{I} \leftarrow \text{Set of pixels, of size } N_I \text{, drawn randomly from } X, Y$$

$$\begin{aligned} \mathbf{J} &\leftarrow \text{Set of pixels, of size } N_J \text{, drawn randomly from } X, Y \\ T &\leftarrow T + \lambda \frac{d}{dT} F(T) \end{aligned} \tag{5.8}$$

where λ is called the *learning* factor. The transformation parameters are updated according to the equation (5.8) until convergence. Note again that the randomly sampled set of pixels, $\mathbf{I}$ and $\mathbf{J}$, are used to estimate, both, the Parzen window estimates of the component density functions, equation (5.5), and the joint entropy terms, equation (5.6). The parameter λ needs to be set empirically. In our simulations, λ was set to 0.02 (also see appendix C.6). The parameter update strategy, equation (5.8), is analogous to the gradient descent optimization strategy, where the stochastic nature of the updates arises from the fact that the cost function, $F(T)$, is only a stochastic approximation of the true energy function. The stochastic nature of $F(T)$ is due to the fact that a small sample of pixels, sampled at random from the images, is used to estimate the various density functions. In the gradient optimization strategy, where the gradient needs to be evaluated often, this strategy leads to considerable savings in computation. However, according to stochastic approximation theory [95, 53], the stochastic gradient can be used instead of true gradients only when the following three conditions are satisfied:

- The gradient estimate is unbiased. That is, the average of the estimated gradient should be the true gradient.

- The parameter update rate asymptotically converges to zero. This condition is required for the algorithm to converge to *a* minimum.

- The error surface is quadratic in the parameters. If the surface is smooth and non–linear, then there is no guarantee that the algorithm will converge to a global

minimum.

Appendix C.4 develops derivatives of various terms in equation (5.8). It is expected that the stochastic nature of the parameter update may help the algorithm escape a local optimum and converge to the global optimum.

5.2.4 Estimation of Covariance Matrix

The covariance matrices of the Gaussian kernels, in the Parzen window estimates of the joint density functions, are another set of parameters to be estimated. These covariance matrices are assumed to be diagonal, that is,

$$\Psi_a = \mathrm{DIAG}(\sigma^2_{axx}, \sigma^2_{ayy})$$

However, note that this assumption does not mean that the marginal density functions of the pixel intensities of the two images are independent. In the Parzen window estimates of the probability density functions, the same Gaussian kernel is used for all the sampled pixel intensities. Thus, only two parameters need to be estimated for each joint density.

The covariance parameters are estimated using a strategy similar in form to the estimation of the transformation parameters (as in [117]), by minimizing the empirical entropy using a stochastic gradient descent method.

$$\begin{aligned}
\mathbf{I} &\leftarrow \text{Set of pixels, of size } N_I \text{, drawn randomly from } X, Y \\
\mathbf{J} &\leftarrow \text{Set of pixels, of size } N_J \text{, drawn randomly from } X, Y \\
\sigma_{axx} &\leftarrow \sigma_{axx} + \lambda \frac{d}{d\sigma_{axx}} H_a(x, y) \qquad (5.9)
\end{aligned}$$

Appendix D.6 develops the derivative in equation (5.9). σ_{ayy} and σ_y are estimated likewise.

5.3 Coordinate Descent Interpretation

The minimax entropy algorithm above is developed within a probabilistic framework. However, within the optimization framework the algorithm can be viewed as a coordinate descent approach which seeks to optimize a cost function by iterative estimation of the parameters along different coordinates. Let

$$\begin{aligned} F_H(\tilde{P}, T) &= -H(M, X|Y) + H(M) \\ &= \iint dX\ dY \sum_M p(M, X, Y)\ \ln p(M, X|Y) - \sum_M \tilde{P}(M)\ \ln \tilde{P}(M) \\ &= \iint dX\ dY \sum_M p(X, Y|M)\ \tilde{P}(M)\ \ln p(M, X|Y) - \sum_M \tilde{P}(M)\ \ln \tilde{P}(M) \end{aligned}$$

Optimization of the energy function $F_H(\tilde{P}, T)$ utilizing the coordinate descent approach leads to the following two steps (see appendix C.5 for details):

Step 1:

$$\tilde{P}^k(M) = \arg\max_{\tilde{P}} F_H(\tilde{P}, T^{k-1}) \tag{5.10}$$

under the constraint $\sum_M \tilde{P}^k(M) = 1$.

Step 2:

$$T^k = \arg\max_{T} F_H(\tilde{P}^k, T) \tag{5.11}$$

Note that in step 1 the energy function $F_H(\tilde{P}, T)$ is being optimized to estimate $\tilde{P}(M)$, utilizing the transformation parameters T estimated from previous iteration, under the constraint $\sum_M \tilde{P}(M) = 1$, is equivalent to the *max* step. Thus, the estimation of the density function $\tilde{P}(M)$, a variational calculus problem within the optimization framework, is interpreted as the *maximum entropy* estimation of a density function within the probabilistic framework. Furthermore, step 2, where we optimize $F_H(\tilde{P}, T)$ to estimate $\mathbf{T}$, utilizing current estimates of $\tilde{P}(M)$, is equivalent to the *min* step as the marginal entropy term $H(M)$ and is independent of the transformation parameters $\mathbf{T}$.

5.4 Utilizing Dual Portal Images

The algorithm described above utilizes only one portal image, typically the anterior–posterior (AP) portal image, to estimate the pose of 3DCT data set. It is expected that utilizing another portal image, typically the left–lateral (LL) portal image acquired in the orthogonal direction during the treatment, will greatly enhance the accuracy of the estimated pose as while utilizing two orthogonal portal images only one rotation angle will be out–of–plane rotation angle to be estimated. Thus, we extend the algorithm to utilize two portal images, AP and LL, for the estimation of the pose. Both, the min step and the max step, are modified to incorporate the information. While estimating the segmentations of the two portal images, in the *max* step, we assume that the segmentations of the two portal images are statistically independent. In the max step for the 2–portal case, $P^k(M_{AP})$ and $P^k(M_{LL})$ are estimated by repeating equation (5.2) for the two portal images separately. Note that though there are two portal images, whose segmentation is being estimated separately, there is only one set of transformation parameters, T, which is to be estimated from both the portal images. In the initial formulation of the algorithm,

we estimate the optimal pose that is a compromise between the best map that matches each portal image. The initial formulation of the *max* step and *min* step utilizing dual portal images is now written as:

Max Step:

$$P^k(M_{AP}) = \arg \max_{P(M_{AP})} \left[-\sum_{M_{AP}} P(M_{AP}) \ln P(M_{AP}) + \sum_{M_{AP}} P(M_{AP}) \ \ln P(M_{AP}|X_{AP}, Y_1(T^{(k-1)})) \right] \tag{5.12}$$

$$P^k(M_{LL}) = \arg \max_{P(M_{LL})} \left[-\sum_{M_{LL}} P(M_{LL}) \ln P(M_{LL}) + \sum_{M_{LL}} P(M_{LL}) \ \ln P(M_{LL}|X_{LL}, Y_2(T^{(k-1)})) \right] \tag{5.13}$$

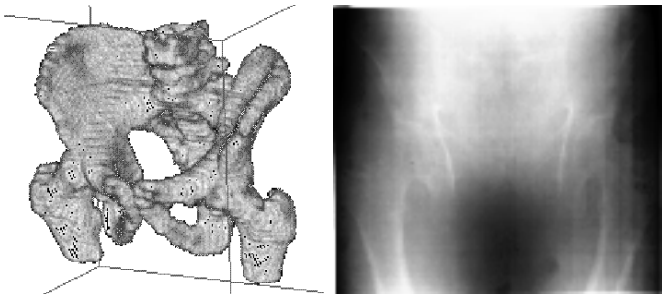
Min Step:

$$\hat{T}^k = \arg \min_T \left[H(M_{AP}, X_{AP}|Y_1) + H(M_{LL}, X_{LL}|Y_2) \right] \tag{5.14}$$

where X_{AP}, X_{LL} denote the AP and the LL portal image respectively, and Y_1, Y_2 denote the DRRs obtained from the 3DCT data set, at the current estimate of the parameters, in the AP and LL directions.

5.5 Extension to 2D/2D Image Registration

The proposed algorithm has been extended to register a 2D portal image to a 2D simulator image (see earlier discussion of simulator images). The extension is straightforward, with the random variable $\mathbf{Y}$ now denoting the *transformed* simulator image, rigidly transformed using current estimates of the transformation parameters. Note that while registering a



(a) (b)

Figure 5.2: (a) Volume rendered 3DCT phantom for which the simulated portal images were calculated. (b) Real portal image of the phantom obtained by taking the high energy X–ray of the phantom in the treatment room.

set of 2D images, we are assuming a rigid transformation between the set of images and hence the transformation parameters consist of only two translations and a rotation.

5.6 Results

This section evaluates the accuracy and robustness of the proposed minimax algorithm using both real patient and simulated data. To obtain the simulated data, a polymethyl methacrylate (PMMA) pelvic bone phantom was scanned to provide the 3DCT dataset. The phantom consists of real human pelvic bone encased in PMMA, with a density close to that of soft–tissue. The phantom was then moved to the treatment room to obtain real portal images at the treatment energy X–ray level (6 MV). The simulated portal images are obtained in the following fashion. First, the 3DCT voxel values are mapped from diagnostic energy values to the values at the treatment energy X–rays using the attenuation coefficient tables [56]. Second, the 3DCT data set is transformed by known

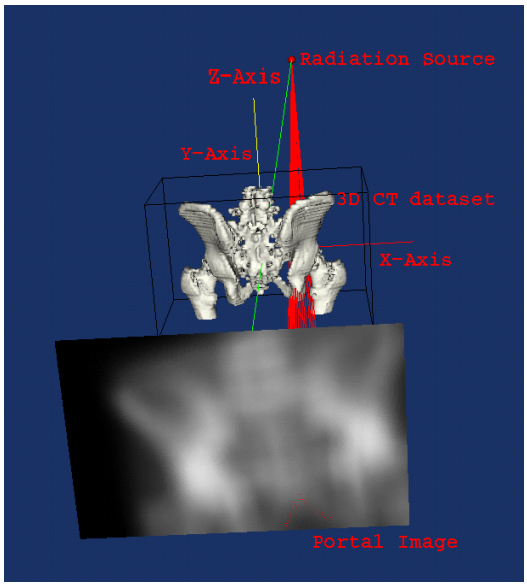


Figure 5.3: The set-up of the radiation source, 3DCT data and the simulated anterior–posterior portal image.

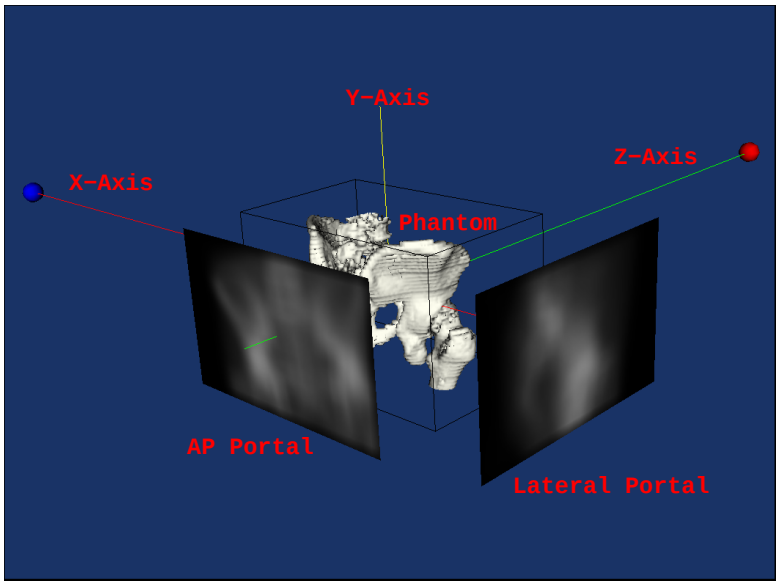


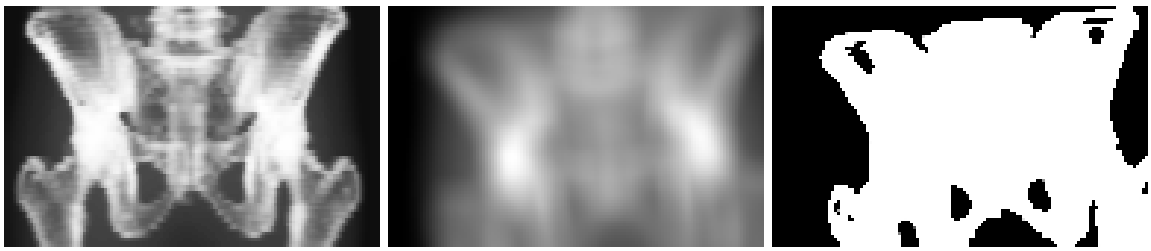
Figure 5.4: The set-up of the radiation source, 3DCT data and the simulated anterior–posterior (AP) and left lateral (LL) portal images.

transformation parameters. Third, the digitally reconstructed radiographs (DRRs) are rendered from the CT data set, both in the anterior–posterior (AP) and the left–lateral (LL) directions. Two different sets of simulated portal images are then obtained from the resulting DRRs. To obtain the first set of simulated portal images, varying amounts of i.i.d. Gaussian noise are added to the DRRs. To obtain the second set of simulated portal images, the DRRs are blurred using blurring kernels of increasing width, which simulates the finite size of the radiation source, and low contrast and low sharpness of the real portal images. Next, the 3DCT dataset is set to its untransformed position and the algorithm is executed to estimate the transformation parameters. Note again that this is a 2D/3D registration problem as the 3DCT data set is being registered to the 2D portal images for the estimation of the registration parameters. Since the true registration parameters are known for the simulated portal images, these datasets are used to study the accuracy and robustness of the algorithm under increasing noise and blur in the images. The perspective parameters are assumed to be known.

Figure 5.2 (a) shows the surface rendering of the pelvic bone of the phantom dataset. Figure 5.2 (b) shows the real portal image obtained by imaging the phantom with X–rays at treatment energy level (6 MeV). The real portal image was contrast enhanced by histogram equalization.

Patient setup verification, in many radiotherapy departments, is still carried out by aligning the 2D simulator images to the 2D portal images, also referred to as two dimensional analysis of the patient setup as such an analysis leads to the estimation of the in–plane transformation parameters only. Thus, we have have modified our registration algorithm to register 2D simulator images to 2D portal images. First we present some results registering 2D simulated portal images to 2D simulated simulator images. Next

	t_x (pix)	t_y (pix)	θ_{XY} (deg)
True	10.0	5.0	5.0
Estd.	12.11	4.735	5.461

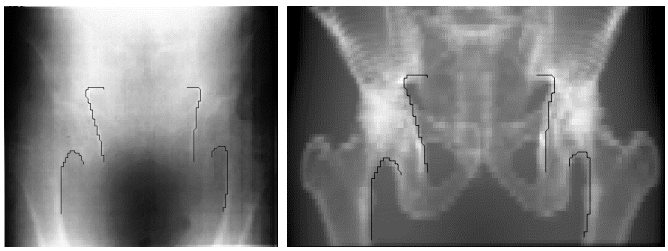


(a) (b) (c)

Figure 5.5: (a) Simulator image. (b) Simulated portal image. The simulator image was rotated and translated and then blurred using a low pass filter of width 11. (c) Segmentation of the portal image estimated by the proposed algorithm.

we show the performance of the algorithm registering 2D anterior–posterior (AP) portal image to 3DCT. Figure 5.3 shows the setup for registering single AP portal image to 3DCT data sets. As the results below suggest, the proposed algorithm is not robust to the estimation of the out–of–plane transformation parameters when using only a single AP portal image. Therefore, the proposed algorithm has been extended to utilize dual portal images, by using two forms of equation (5.1) for each portal segmentation and a summed version of equation (5.4) for both portal images. Note that the two portal images can be obtained from two different views which are not necessarily orthogonal. Figure 5.4 shows the setup of the pelvic phantom and the AP and LL portal images in three–dimensional space. As is clear from the figure, the in–plane translations for the AP

Estimated Parameters					
t_x (vox)	t_y (vox)	t_z (vox)	θ_{YZ} (deg)	θ_{XZ} (deg)	θ_{XY} (deg)
2.56	-6.69	2.34	2.51	-1.24	0.73



(a) (b)

Figure 5.6: (a) Portal image with hand–drawn contours. (b) DRR with mapped contours.

portal image consist of translations along the X and Y axes and the in–plane rotation is the rotation about the Z–axis. For the lateral portal image, the in–plane translations are the translations of the 3DCT dataset along the Y and Z axis and the in–plane rotation is the rotation about the X–axis. Note that by using two portal images, now only the rotation about the Y–axis remains as the out–of–plane rotation for both the images.

Three dimensional analysis of the patient setup was performed using both a single AP portal image and dual AP and LL portal images. 3D analysis leads to the determination of 3 translations and 3 rotations which correctly align the patient to the treatment beam in the treatment room.

(a) (b) (c)

Figure 5.7: Estimated segmentation results of the true portal image. (a) Results of using proposed minimax entropy algorithm. (b) Simple threshold. (c) Clustering algorithm in commercial package (MEDx [101]) for comparison.

5.6.1 Registration of 2D/2D Images

As noted above, for two dimensional analysis of the patient, the 2D simulator image is registered to the 2D portal image. A 2D simulator image, shown in figure 5.5 (a), is obtained by projecting the diagnostic energy 3DCT data. The simulated portal image in figure 5.5 (b) is obtained by first transforming the simulator image and then blurring it with a uniform blurring kernel of width 11. The first row of the table in figure 5.5 shows the true transformation parameters, where t_x and t_y denote, in pixels, the translations along the X and Y axis respectively. θ_{XY}, in degrees, denotes the in–plane rotation. The second row of the table shows the parameters as estimated by the proposed algorithm. Note that although the portal image was blurred by 11 pixels, the algorithm was able to estimate the X–translation within 2 pixels and Y–translation within 0.25 pixels. The rotation angle was also estimated very accurately. The segmentation of the portal image as estimated by the proposed algorithm is shown in the figure 5.5(c), showing an intuitively plausible result.

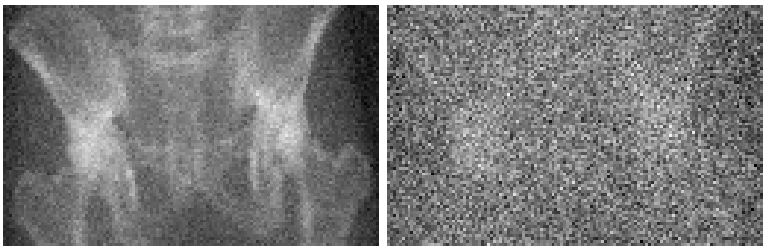
5.6.2 2D/3D Registration

Three dimensional analysis of the patient setup is done using both single (AP) and dual (AP and LL) portal images. We first present results using the real portal image and then using the simulated portal images. The simulated portal images are obtained as explained above. The proposed registration algorithm estimates six transformation parameters, where the three translations, in voxels, are denoted as t_x, t_y and t_z (along the x–axis, y–axis and z–axis respectively) and the three rotations, in degrees, are denoted as θ_{YZ}, θ_{XZ} and θ_{XY} (about the x–axis, y–axis and z–axis respectively).

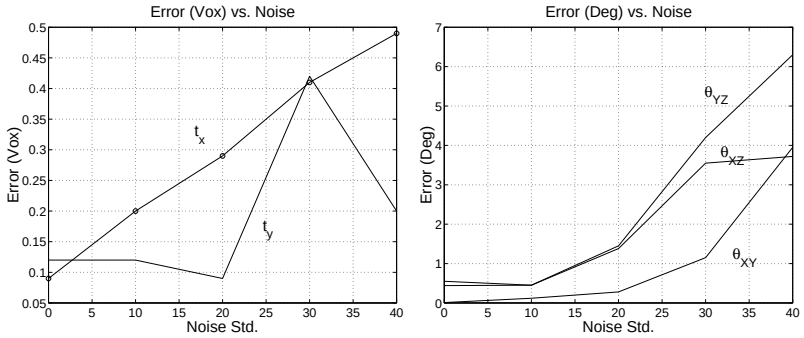
5.6.2.1 3D Analysis Using Single Portal Image

In this subsection, the results of registering a real AP portal image of the pelvic phantom to its 3DCT data set are presented. To study the effect of noise on the accuracy of the estimated parameters, we also register simulated AP portal images, with increasing noise, to the 3DCT dataset.

Figure 5.6 (a) shows the real portal image which is registered to the 3DCT dataset. The parameters estimated by the proposed algorithm, to bring the two datasets into alignment, are shown in the table in figure 5.6. Figure 5.6 (b) shows the DRR obtained from the 3DCT dataset at the parameters estimated by the minimax entropy algorithm. To verify the goodness of the estimated parameters, contours are hand drawn on the portal image, figure 5.6 (a), by an expert matching the key features (pelvic crest, femur outlines) closely. These contours are then mapped onto the DRR, figure 5.6 (b), undeformed. Note that the contours match closely to the corresponding features in the DRR, verifying the accuracy of the estimated parameters for this dataset. However, a slight mismatch indicates a rotation about the X–axis.



(a) (b)

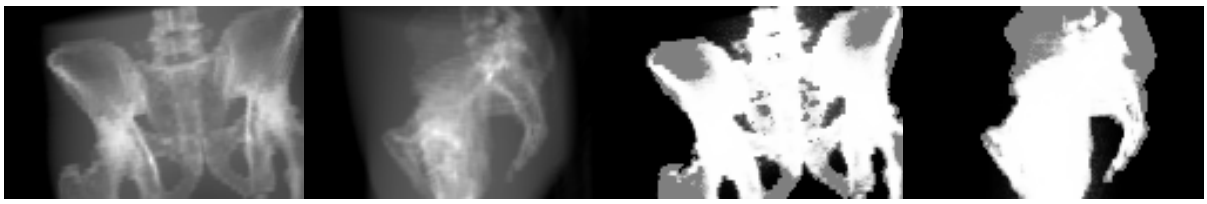


(c) (d)

Figure 5.8: (a) Simulated portal image with Gaussian noise of std (σ) 10. (b) $\sigma = 30$. (c) & (d) Error in estimated parameters with increasing noise.

Different segmentations of the portal image are shown in the figure 5.7. White pixels denote estimates of bone and the black pixels denote background. Figure 5.7(a) shows the segmentation estimated by the minimax entropy algorithm, where the segmentation labels are being estimated using the joint probability density function. Gray values denote the pixels whose label could not be determined. The estimated segmentation is compared to two other segmentations, a simple threshold based segmentation, figure 5.7(b), and segmentation obtained by a clustering algorithm in commercial program (MEDx [101]), shown in figure 5.7 (c). In this clustering algorithm, the image is

	t_x (vox)	t_y (vox)	t_z (vox)	θ_{YZ} (deg)	θ_{XZ} (deg)	θ_{XY} (deg)
True	15.0	5.0	0.0	0.0	0.0	10.0
Estd.	15.07	4.95	0.045	0.20	0.08	9.95



(a) (b) (c) (d)

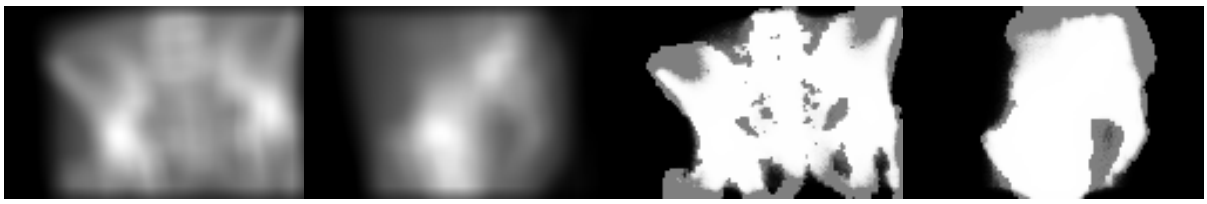
Figure 5.9: (a) Simulated AP portal image. (b) Simulated left–lateral portal image. (c) Segmentation of the AP portal image estimated by the proposed algorithm. (d) Estimated segmentation of the LL portal image. Note again that in the segmentation images, white denotes the bone, black denotes the background and gray denotes the pixels mis–classified by the algorithm. The second row of the table shows the parameters estimated by the minimax entropy algorithm.

segmented into three classes based on the nearness of the gray values to the user specified values. In this classification, light gray denotes the bone and the black pixels denote the background.

As noted earlier, to study the robustness of the algorithm against noise, simulated portal images with increasing i.i.d Gaussian noise are registered to the 3DCT dataset. Example simulated portal images with different amounts of noise are shown in the figures 5.8 (a) and (b). The image in figure 5.8 (a) is obtained by first rotating the 3DCT dataset by 5° along Z–axis and then adding Gaussian noise of standard deviation (σ)

10.0 to the rendered DRR. Gaussian noise of standard deviation (σ) 30.0 is added to obtain the image in figure 5.8 (b). Figures 5.8 (c) and (d) show the graphs of estimated parameters with increasing noise in the simulated portal image. Simulated portal images for the graph labeled t_x are obtained from the 3DCT in the following fashion. The 3DCT dataset is translated along the X–axis by 5 voxels and then a DRR is rendered. Now, varying amounts of Gaussian noise are added to the DRR. The 3DCT data set is then initialized to its untransformed position and the algorithm is run to estimate the transformation parameters. The difference in the estimated parameter and the true parameters is plotted along the Y–coordinate in the figure 5.8 (c), with noise standard deviation along the X–coordinate. From figure 5.8 (d) it appears that the error in the estimated parameter θ_{XZ} saturates for i.i.d. Gaussian noise of standard deviation 30.0 or more in the simulated portal images. However, no such interpretation can be drawn from the figure. The true rotation angle to be estimated is only 5° and for noise more than standard deviation 30.0 the algorithm is completely unable to estimate the rotation angle and remains stuck in its initial position. The results can be summarized as follows. The minimax entropy algorithm is quite robust against noise for in–plane translations of the 3DCT dataset, that is, translation along the x–axis and y–axis. However, for both in–plane and out–of–plane rotations, the estimation of the parameters becomes poor with increasing noise. This study suggests that the algorithm is robust to noise for in–plane transformations only. Thus, to increase the accuracy and robustness of the algorithm for estimating out–of–plane transformation parameters, we have extended our algorithm to use dual portal images, obtained from two different views.

	t_x (vox)	t_y (vox)	t_z (vox)	θ_{YZ} (deg)	θ_{XZ} (deg)	θ_{XY} (deg)
True	15.0	5.0	0.0	0.0	0.0	10.0
Estd.	15.62	5.19	-0.06	0.43	0.21	10.85



(a) (b) (c) (d)

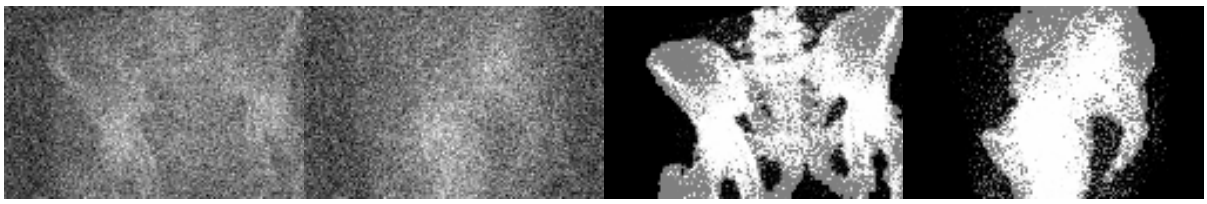
Figure 5.10: (a) Simulated AP portal image. (b) Simulated left–lateral portal image. (c) Estimated segmentation of the AP portal image. (d) Segmentation of the LL portal image estimated by the proposed algorithm. Estimated and the true parameters are shown in the table.

5.6.2.2 3D Analysis Using Dual Portal Images

Based on the conclusions from the previous section, the algorithm was further developed (as described in section 5.4) to utilize dual portal images simultaneously to estimate the transformation parameters. In this section, dual simulated portal images in the AP and LL directions are used to study the accuracy and robustness of the algorithm.

Figure 5.9 shows the simulated AP and LL portal images registered to the 3DCT dataset. The first row of the table in the figure shows the true registration parameters and the second row shows the parameters estimated by the algorithm. Note that the estimated parameters are within half a voxel of the true parameters. The rotations have also been accurately estimated. The estimated segmentations of the portal images are

	t_x (vox)	t_y (vox)	t_z (vox)	θ_{YZ} (deg)	θ_{XZ} (deg)	θ_{XY} (deg)
True	15.0	5.0	0.0	0.0	0.0	10.0
Estd.	15.36	5.23	-0.02	-0.06	0.11	10.03



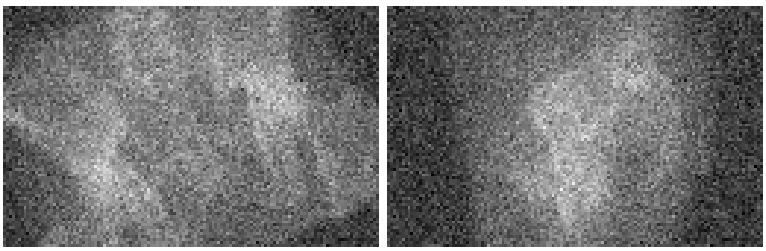
(a) (b) (c) (d)

Figure 5.11: Simulated portal images with noise. (a) AP with std 30.0 (b) Left–lateral with std 30.0 Estimated segmentation of (c) AP portal image. (d) LL portal image. The table show the true and the parameters estimated by the algorithm.

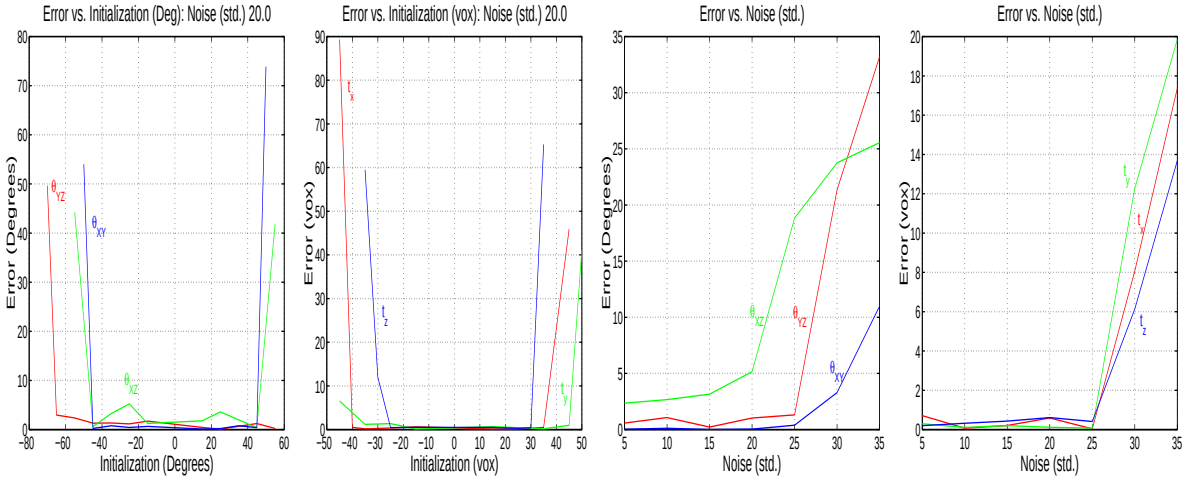
shown in figure 5.9 (c), (d).

The simulated portal images are blurred by using a uniform blurring kernel of width 11 to obtain the portal images in figure 5.10 (a), (b). Figure 5.10 (c), (d) show the corresponding segmentation of the portal images estimated by the algorithm. Again, the table in the figure shows the true and the estimated parameters. Note that the estimated translations are within 1 voxel of the true values, even in the presence of a blur of 11 pixels. The estimates of the rotation parameters are within 0.5^o of the true values.

Figure 5.11 (a), (b) shows the simulated portal images with Gaussian noise of standard deviation = 30.0. Note again the accuracy of the parameters estimated by the algorithm. Figure 5.11 (c), (d) shows the segmentation of the portal images as estimated by the algorithm.



(a) Simulated AP portal (b) Simulated lateral portal



(c) (d) (e) (f)

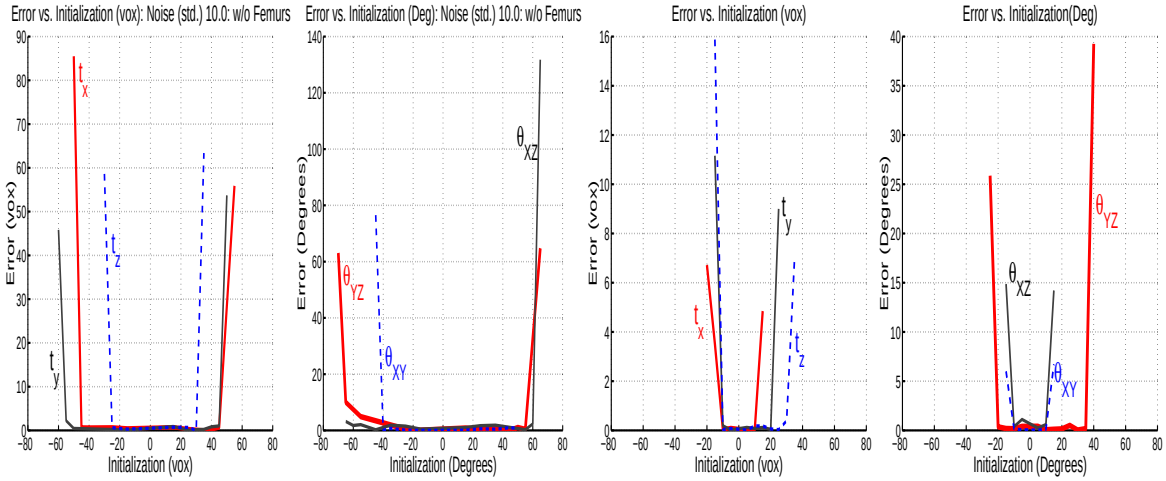
Figure 5.12: (a), (b) Example simulated portal images with Gaussian noise of 20.0. (c) The graphs show the error in estimated versus the true rotation angles. (d) These are the graphs of error in estimated translation versus the true translations. (e) Error in estimated angles with increasing noise. (f) Error in estimated translation with increasing noise.

Figure 5.12 (c) and (d) shows the graphs of error in the estimated parameters when the algorithm is initialized with varying amounts of rotational and translational setup variations, in the presence of Gaussian noise of standard deviation ($\sigma = 20.0$) in the simulated portal images. These graphs are obtained in the following fashion. First, the 3DCT dataset is transformed by a known amount and, AP and LL DRRs are obtained. Then i.i.d. Gaussian noise of standard deviation 20.0 is added to the DRRs to obtain the simulated portal images. Example portal images used for these graphs are shown in figures 5.12 (a) and (b). Then, for the graph labeled θ_{YZ}, only the parameter θ_{YZ}, which denotes rotation about the X–axis, is varied to obtain the DRRs. All other parameters are kept fixed at the true values. The 3DCT is then reset to its untransformed position and the algorithm is run to estimate the transformation parameters. Note that in this step all parameters are allowed to vary and are being estimated. The error in the estimated parameter with respect to the true parameters is then plotted. The graphs show that, for this dataset, the algorithm could estimate the rotation angles up to 50° accurately. These figures also show that either the algorithm is quite accurate in estimating the parameters or it breaks down completely, that is, the estimated parameters are completely different from the true parameters. This shows that the algorithm gets trapped into a local minimum if the global minimum is very far from the initial starting position. For large translations, very little pelvic structure remains in the image, making it easier for the algorithm to get stuck into a local minimum. Graphs in figure 5.12 (e) and (f) show the performance of the algorithm under increasing noise. The AP and LL portal images, for example for the graph labeled θ_{XY}, are obtained by first rotating the 3DCT data by 15° about the Z–axis and then rendering the DRRs both in the AP and the LL directions. Varying amounts of noise are then added to the DRRs to obtain the simulated portal

images. The 3DCT data set is then initialized to its untransformed position and the algorithm is run to estimate the transformation parameters. The graph shows the error in estimated transformation parameters for various amounts of noise. Similarly, for the graphs labeled $\theta_{YZ}, \theta_{XZ}, t_x, t_y, t_z$, the 3DCT data set was transformed by 30°, 25°, 20 voxels, 20 voxels and 15 voxels respectively to obtain the DRRs in the AP and the LL directions.

5.6.3 Performance with Femur Edited Out

The registration of the 3DCT dataset to the portal images, in the proposed minimax entropy registration framework, is guided primarily by the bony anatomy visible in the images. Some of the prominent bony features visible in the portal images are due to the femurs. As the femurs could move with respect to the pelvic cage between treatment planning and the delivery of the treatment, it may not be a good idea to include them in the registration. This section evaluates the performance of the minimax entropy framework, and compares it to the performance of the Gilhuijs' [44] ridge–based registration algorithm, using image dataset where the femurs have been *edited out* manually from the 3DCT dataset only. However, the femurs are visible in the simulated portal images. The simulated AP and LL portal images are obtained at known transformation parameters with Gaussian noise of $\sigma = 10.0$ and are registered to only the 3DCT image data in the pelvic cage. The results for a range of different translations and rotations are shown in the figures 5.13 (a) and (b). Comparing these results to the femur–included registration performed in figures 5.12 (c) and (d), indicate that the algorithm performs almost equally well with or without femurs. Figures 5.13 (c) and (d) show the performance of the Gilhuijs' ridge–based registration algorithm on the same image dataset. However, this approach required significant manual editing of the ridge–enhanced images to remove the



(a) (b) (c) (d)

Figure 5.13: All the results were performed using simulated AP and LL portal images with noise level set to $\sigma = 10.0$. (a) and (b) show recovery of varying translation and rotational setup error when using the 3DCT–to–Dual Portal minimax entropy registration framework with *femurs edited out* from the 3DCT image. (c) and (d) show the same result when the algorithm designed by Gilhuijs [44] is run on the same data. However, the Gilhuijs algorithm required human interaction. Note the smaller range of zero error for each parameter for the Gilhuijs' [44] algorithm.

spurious ridges. Note that the Gilhuijs' ridge–based algorithm has a smaller range of *close–to–zero* error for each parameters.

5.6.4 Performance on Real Data

Figure 5.14 shows the results of running the proposed algorithm on real patient data. Figures 5.14 (a) and (b) show histogram equalized AP and LL portal images, respectively. The DRRs projected through the 3DCT data in its original pose are shown in

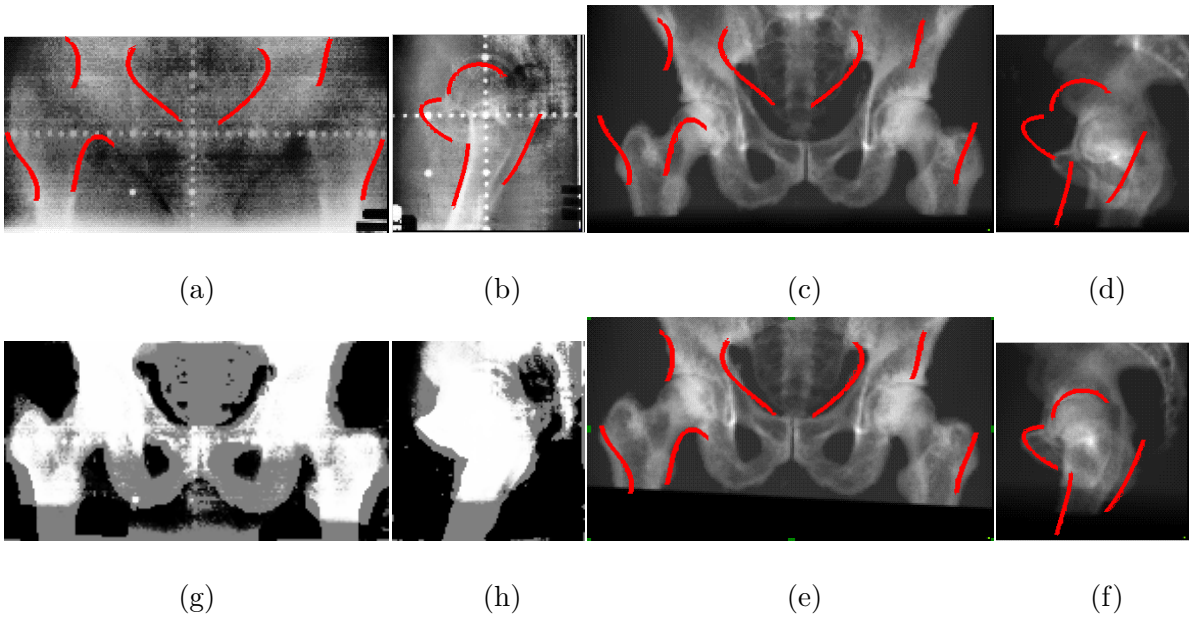


Figure 5.14: Recovery of setup variation using *actual patient data* and an early version of the registration algorithm proposed in sec. D.1. (a,b) Portal images, (c,d) DRR's of 3DCT in original pose, (e,f) DRR's in corrected pose, (g,h) implicit segmentation of portals. See more detailed explanation in the text.

the figures 5.14 (c) and (d). Running the algorithm estimates a new pose of the 3DCT dataset, which differs from the original pose by $\theta_{XY} = -3.2^o$, $\theta_{YZ} = 2.93^o$, $\theta_{XZ} = -1.93^o$, $t_x = 4.47$ voxels, $t_y = -27.5$ voxels and $t_x = -14.54$ voxels. The DRR projections in the new pose are shown in figures 5.14 (e) and (f). Segmentations of the AP and LL portal images, estimated by the algorithm, are shown in figures 5.14 (g) and (h), respectively. Note that the segmentations of the portal images are being used implicitly by the algorithm. Because of the poor quality of these *digitized portal film* images, the segmentation step was initialized manually in several regions where the background and bone were hard

to distinguish (We were able to run algorithm completely automatedly after the X–ray films were digitized using a laser scanner). To access the accuracy of the estimated pose of the 3DCT data set, contours are hand drawn on the portal images, matching visible features. These contours are then mapped onto the DRRs, in figures 5.14 (c), (d), (e), (f) undeformed (note that the contours are used only to visually assess the goodness of the estimated pose). Note that the contours match closely to the features in DRRs obtained at the pose estimated by the algorithm.

5.7 Summary

This chapter presents the proposed information theoretic framework in which segmentation and registration are carried out simultaneously and iteratively, with segmentation results helping in the registration and vice–versa. Feature–based registration methods proposed in the literature carry out portal image segmentation as a pre–processing step in the registration process. Our approach of simultaneously segmenting and registering the images, using a unified framework, leads to an accurate and a robust algorithm.

The mutual information based registration methods overcome the assumption of a linear relationship between the pixel intensities of the images to be registered, an underlying assumption in the normalized cross–correlation match method. In these methods, the relationship between the pixel intensities is estimated from the given data itself and can thus register images from different modalities. At an estimated set of transformation parameters, a joint density between the images to be registered can be estimated from the given data. The mutual information assigns a number to each such estimated density. The transformation parameters corresponding to the density having the largest mutual information are chosen as the parameters estimated by the algorithm. The EM

algorithm provides an iterative framework to estimate the parameters of a distribution, in the presence of missing data. However, the EM algorithm requires that the parametric form of the distribution be known.

The proposed minimax entropy algorithm overcomes this restriction of the EM algorithm, by borrowing the idea from the mutual information method of estimating the joint distribution from the given data. This allows us to invoke other constraints on the distribution systematically, which we use to impose an annealing schedule. In the following chapters, we will impose regularization constraints on the estimated segmentation of the portal images within the same framework.

This chapter studied the algorithm's robustness against noise by increasingly adding Gaussian i.i.d. noise to the simulated portal image. In the initial stages, with no segmentation information on the portal image, the proposed algorithm basically searches for the transformation parameters which increases the mutual information. In the later stages, the algorithm maximizes the mutual information in the separate regions estimated in the entropy maximization step.

Our experiments showed that using a single portal image led to erroneous estimation of the out–of–plane transformation parameters. Thus, we extended our algorithm to use dual portal images taken at two views, where the views need not be orthogonal. The results of the algorithm using dual simulated portal images are very encouraging. The algorithm was also extended to register a 2D simulator image to a 2D portal image. For the 2D registration experiments, the algorithm leads to very accurate registration and reasonably good segmentation.

Chapter 6

Performance Analysis and Comparisons

6.1 Introduction

The previous chapter developed the minimax entropy (MIME) registration framework where the segmentation and registration of the portal images was carried out simultaneously and iteratively, with segmentation helping the registration and vice–versa. It is the hypothesis of this chapter that incorporation of the segmentation information into the registration framework leads to a more accurate and robust algorithm. To prove the hypothesis, in this chapter we first present the results, using energy function plots, which show that incorporating *true* segmentation information does lead to a more robust and accurate algorithm, in the presence of both noise and blur in the portal images. Next, this chapter compares and evaluates the performance of MIME with the performance of MI and the ridge–based (GL) algorithm [44]. The ridge–based algorithm uses bony ridges, extracted from the portal images, as the features for registration. We chose these

algorithms for two reasons. First, these algorithms are well–accepted algorithms in the medical image analysis area and in the radiotherapy treatment area respectively. Second, these two algorithms are intended for completely automated use and are from the opposite ends of the spectrum of registration strategies. Dense field based registration methods form one end, feature–based registration methods form the other end of the registration spectrum. Another popular registration method in the external beam radiotherapy (EBRT) literature is gray–scale pixel intensity correlation. However, recently in the medical image analysis literature, mutual information–based algorithms have gained prominence and perform robustly for multi–modal image registration. The *min* step in our algorithm is closely related to the MI algorithm. Thus, we chose to compare the performance of our algorithm to MI rather than the normalized cross–correlation based algorithms.

6.2 Performance Evaluation Strategy

In order to evaluate the performance of our algorithm, we use a treatment planning 3DCT data set obtained from a pelvic phantom and the corresponding simulated portal images. The simulated portal images are obtained by rendering the 3DCT data in the anterior–posterior (AP) and the left–lateral (LL) directions, in a manner as described in the previous chapter. The following subsections briefly discusses the mutual information and the ridge–based algorithms that will be used for comparison purposes.

6.2.1 Ridge–Based Method

As we have already pointed out, we chose the ridge–based algorithm described in [44] for comparison because it has been well–received in the EBRT literature as being a completely

automated registration algorithm for patient setup verification.

In describing this algorithm, we shall closely follow the notation in [45]. We first describe the cost function, then the optimization procedures and lastly the procedures for delineation of the bony structures. It should be noted that this algorithm is specifically designed for the radiotherapy application, that is registration of the CT data set with portal images.

The cost function which this algorithm seeks to minimize, is based on the insight that the high intensity ridges in the transmission images are located at the points where the projection X–rays have local maxima in the trajectory through bone, that is, rays pass through locally maximal bone structure. Thus, first bone ridges are detected and localized in the portal images. The rays from detected ridge features are back–projected to the radiation source through the transformed 3DCT. The negated average of the distance along the back–projected rays through the bone is then minimized to estimate the optimal set of transformation parameters. We have implemented an automated method to detect and localize the bone ridges in the transmission images. The ridges are first enhanced using the morphological top–hat transformation [102] of the portal images. The pixels with high values in the transformed images are assumed to correspond to the bone ridges.

In this formulation, the patient setup verification is being carried out by using two portal images obtained from two different views. Let $I_1(\mathbf{p})$ and $I_2(\mathbf{q})$ be the two transmission images obtained from two different views, with $\mathbf{p} = (i, j)$ and $\mathbf{q} = (u, v)$ denoting the pixel locations in the two images respectively. Let the set of bone ridge locations in the two transmission images be L_1 and L_2 containing N and M elements

respectively. Thus, the two sets are defined as:

$$L_1 = \{\mathbf{p_i} | \mathbf{p_i} \in D(I_1) \wedge i = [1, N]\}$$

$$L_2 = \{\mathbf{q_i} | \mathbf{q_i} \in D(I_2) \wedge i = [1, M]\}$$

where $D(I_1)$ and $D(I_2)$ are the domains over which the pixel locations of the two images are defined. The cost function is defined to be

$$C(T) = -\left(\frac{\sum_{\mathbf{p} \in L_1} r_b(\mathbf{p}, T)}{N} + \frac{\sum_{\mathbf{q} \in L_2} r_b(\mathbf{q}, \mathbf{T})}{M} \right) \tag{6.1}$$

where $\mathbf{T}$ indicates the pose of the patient. The function $r_b(\mathbf{p}, \mathbf{T})$ denotes the distance through the bone on the projection line which connects the ridge located at point $\mathbf{p}$ with the focus, where the 3DCT data set has been transformed according to the transformation $\mathbf{T}$. The CT data set is now segmented into two groups using a threshold. Voxels above the threshold, denoting bone, are set to 1 and the other voxels are set to value 0. In order to evaluate the distance of a projection ray through the bone we just need to sum the voxel values along the path of the ray. A more complex extraction scheme can also be used to account for the partial volume effects. Two different minimization strategies, downhill simplex minimization [80] and Powell's minimization [19], have been implemented and investigated. The minimization procedures are restarted once from the first found local minimum.

6.2.2 Mutual Information

Due to both its popularity in the literature and the fact that it exemplifies a full–intensity–based strategy, we have implemented a mutual information match measure–based registration algorithm for the registration of the portal images and the 3DCT data. In the medical image processing literature, the method was first described by Viola *et al.* and Collignon *et al.* [117, 28]. Here we follow the work of Viola *et al.* [117]. In this approach, the optimal transformation parameters are determined as:

$$\hat{T} = \arg\min_{T} \left[H(I_1(\mathbf{p})) + H(I_2(T(\mathbf{p}))) - H(I_1(\mathbf{p}), I_2(T(\mathbf{p}))) \right] \tag{6.2}$$

where $\mathbf{p} = (x, y)$ are the coordinate locations of the pixels in the image I_1 and I_2 denotes the image which is being transformed, $M(\cdot)$ is the mutual information between the pixel intensities of the two images, defined in terms of the entropies of the two images as, $M(I_1(\mathbf{p}), I_2(T(\mathbf{p}))) = H(I_1(\mathbf{p})) + H(I_2(T(\mathbf{p}))) - H(I_1(\mathbf{p}), I_2(T(\mathbf{p})))$, [83]. The entropy, $H(x)$, of a random variable, $\mathbf{x}$, is defined to be $H(x) = -\int p(x) \ \ln p(x) dx$, and the joint entropy of the random variables is defined, in a similar fashion, to be $H(x, y) = -\int\int p(x, y) \ \ln p(x, y) \ dx \ dy$.

In the mutual information–based methods, as implemented in the literature, it is assumed that the pixel intensities are independently and identically distributed (i.i.d), an assumption which need not be true. The joint intensity density function, $p(I_1(\mathbf{p}), I_2(T(\mathbf{p})))$ and the marginal density function, $p(I_2(T(\mathbf{p})))$ are estimated from the data using the Parzen density method [37]. A stochastic gradient descent method is used as the optimization strategy.

This method differs from the ridge–based method in the following important ways. First, it is general and is not restricted to this application alone. Since the density is estimated from the dataset, nothing about the form of the joint or the marginal density function is assumed. Thus, it can be directly applied for the registration of multi–modality images where the pixel intensities may be related by a non–linear function. Second, it computes a registration directly from the pixel intensities, thus it is a *dense field* method, without requiring segmentation of the images. Hence, it can be easily automated and is robust against noise, as demonstrated below.

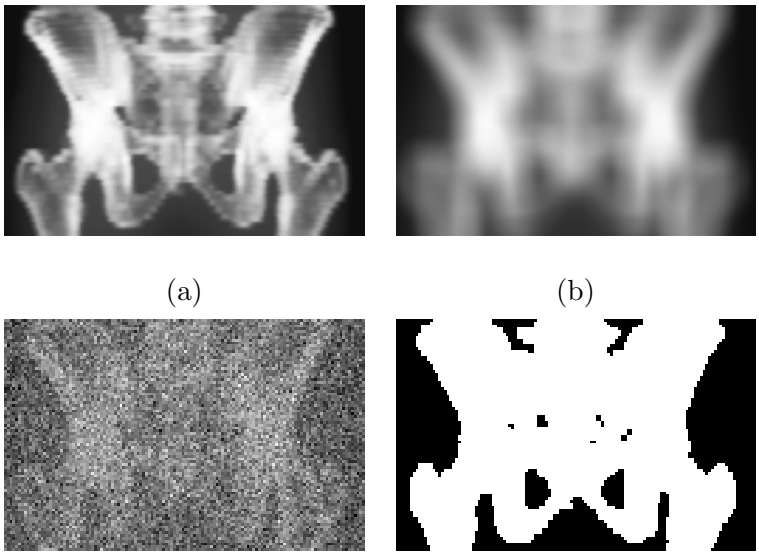
This method has been implemented to register portal and simulator images, as well as to carry out 2D/3D registration using single, and also dual, portal images and 3DCT image data sets.

6.3 Comparisons for 2D/2D Image Registration

This subsection, compares MI and MIME algorithms for the rigid registration of 2D images. Here, a simulator image and simulated portal images, with increasing noise and increasing blur, are registered. Earlier, we noted that our algorithm is an extension of the mutual information–based algorithm where the segmentation information is also evaluated and implicitly used to help registration and vice–versa. It is our working hypothesis that incorporation of segmentation information in the registration framework can lead to robust and accurate estimation of the registration parameters. To help verify our hypothesis, in the following example, we have plotted the MI and the MIME energy functions. In these figures, the portal image is kept fixed and the simulator image is translated, both along the X–axis and the Y–axis, from the true registered position, (0,0). At each new translated version of the simulator image, the MI energy function is evaluated

using the transformed simulator image and the portal image. The MIME energy function is evaluated using, in addition to the portal and transformed simulator images, the true segmentation of the portal image (assumed to be known for demonstration purposes). The evaluated values are then plotted against the translations of the simulator image. In the MIME algorithm, when all the pixels are equally likely to be bone or background, we are effectively maximizing the mutual information between the two images, i.e. minimizing $H(x, y) - H(y)$. This is true as $H(x)$, the entropy of the portal image, is independent of the transformation parameters. Also, to be able to compare the MI to the MIME based method, we have plotted $H(x, y) - H(y)$ for MI and $\sum_{a \in A} \left(\frac{\sum_{i=1}^{N^2} P_i^k(a)}{N^2} \right) H_a(x, y) - H(y)$ for the MIME algorithms.

Figures 6.1 (b) and 6.1 (c) show the simulated portal images, with blur and noise respectively, used for evaluating the two cost functions. In the MI algorithm, as implemented in the literature [117, 27], every pixel intensity is assumed to be i.i.d, with no information on the pixel location. Intuitively, segmentation information removes this uncertainty, to some extent, by using a *tag* on each pixel denoting the region in the image to which the pixel belongs. Pixels in the same region should have closely related pixel intensities. When registration parameters are estimated, pixels from the two images within each region are compared to estimate the homogeneity of the pixel intensities. This leads to accurate and robust estimation of the registration parameters as compared to the parameters estimated by the MI algorithm. We will also see that the MIME estimates rotations more accurately and robustly as compared to the MI algorithm. In this example, as the MIME algorithm is implicitly using the segmentation information of the portal images shown in the figures 6.1 (d), we expect that the cost function of the MIME algorithm to be robust to both noise and blur, with the global optima at



(a) (b)

(c) (d)

Figure 6.1: (a) Simulator Image (b) Simulated portal image, obtained by blurring the simulator image in (a) by a blurring kernel of width 11 pixels. (c) Simulated portal image, obtained by adding i.i.d Gaussian noise of std. (σ) 30.0 to the simulator image in (a). (d) True segmentation of image in (b) and (c). This information is implicitly used by the MIME algorithm.

the true optima. Also, for the MI algorithm, it is expected that increasing blur in the portal image will make the energy function flat, whereas incorporation of noise will lead to many local minima. However, the MIME energy function is expected to stay peaked, with a global optimum at the true parameters, that is, it is robust to increasing noise and blur. To show this, we have plotted the energy functions on the X and Y axes, in an anecdotal snapshot of the parameter search space around the global optimum, as shown in figures 6.2 and 6.3 for large blur and noise respectively. Figures 6.2 and 6.3 support our hypothesis, for the current dataset, that incorporation of the segmentation

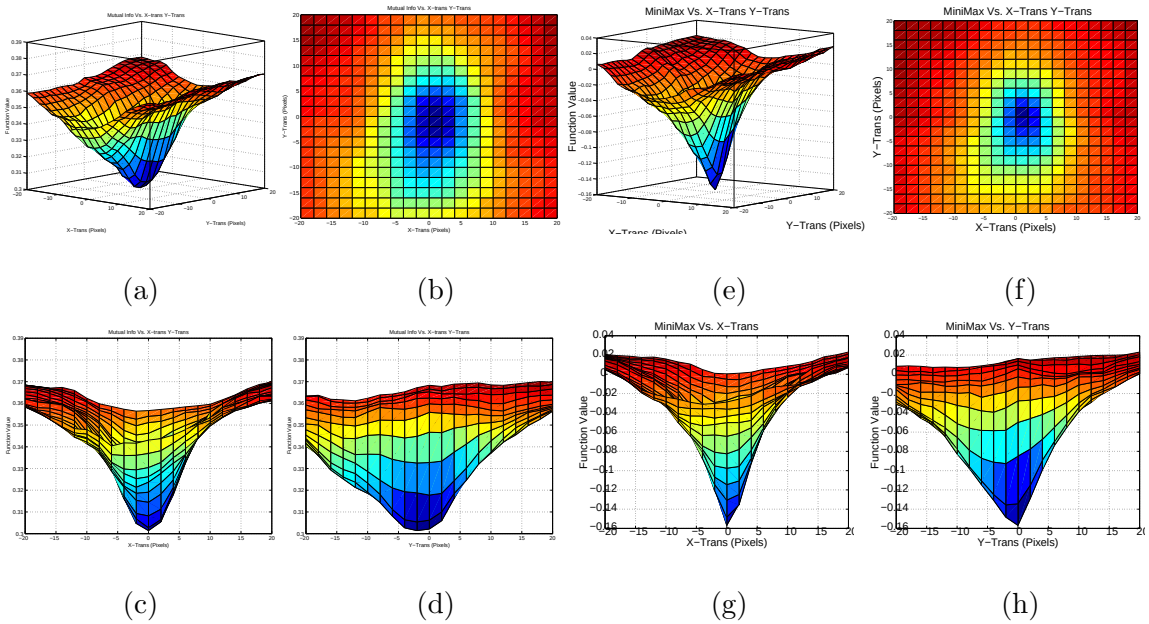


Figure 6.2: Surface plots of the energy functions, shown for only the X and Y translations, for the MI and MIME energy functions respectively. The blurred portal image, figure 6.1 (b), is being registered to the simulator image, figure 6.1(a). (a) shows the energy function of MI as a function of X and Y translation. (b) shows the energy function as viewing it from the top. (c) shows the energy function as viewed along X–axis. (d) shows the energy function as viewd along the Y–axis. (e) is the MIME energy function as a function of X and Y translations. (f) shows the top, (g) shows the view along X–axis and the (h) shows the view along the Y–axes of the MIME energy function.

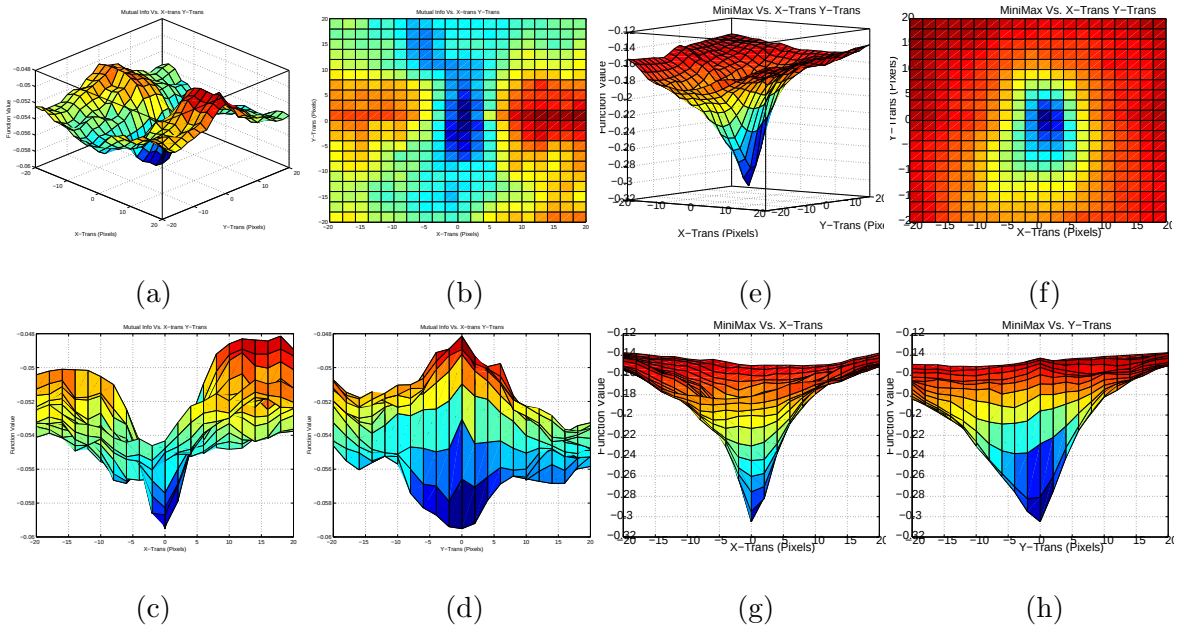
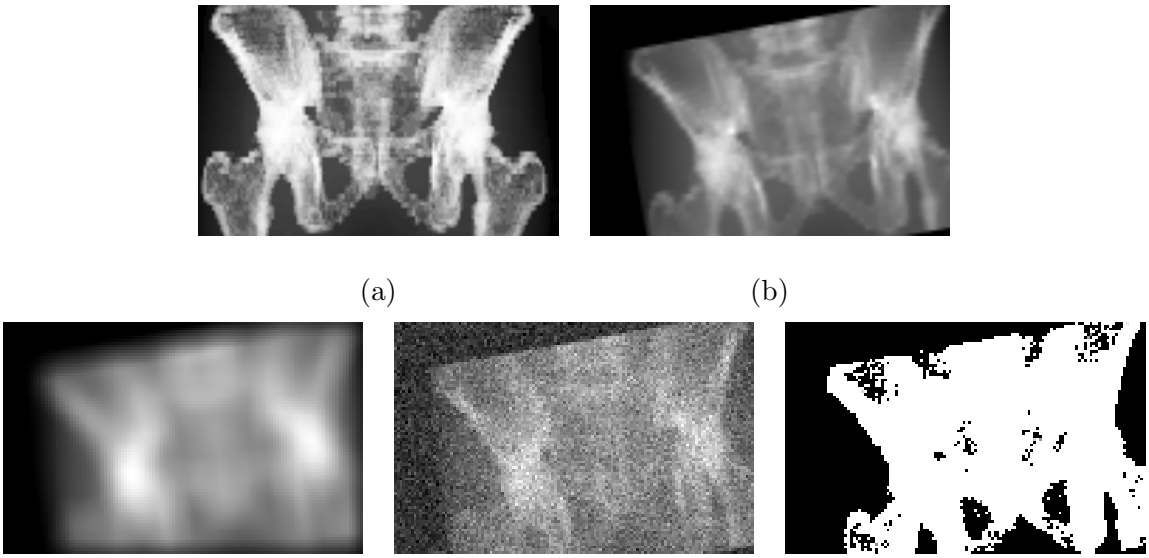


Figure 6.3: Surface plots of the MI and MIME energy functions. Here, the noisy portal image, $\sigma = 30.0$, figure 6.1 (c), is being registered to the simulator image, figure 6.1 (a), (b), (c), (d) are plots of MI as a function of X and Y translation respectively. (e), (f), (g), (h) are the corresponding plots of the MIME energy function.

information can lead to robust and accurate registration. If the portal image is blurred, then the MI energy function becomes smooth and flat, as shown in figures 6.2 (a), (b), (c), (d). Whereas, in the presence of noise in the portal images, the MI energy function shows many local optima and is flatter, as seen in figures 6.3 (a), (b), (c), (d). On the other hand, the MIME energy function is robust to both, the noise and blur in the portal images. Figures 6.2 (e), (f), (g), (h) and 6.3 (e), (f), (g), (h) show that, for this dataset, the form of MIME energy function remains unchanged. The global optimal parameters are seen to be the true registration parameters for the MIME energy function, whereas, there is a small shift in the local of the global optimal parameters for the MI energy



(a) (b)

(c) (d) (e)

Figure 6.4: (a) Simulator image. (c) Simulated portal image. (c) Simulated portal image blurred with kernel of width 11. (d) Simulated portal image with Gaussian noise of std. (σ) 20.0 added. (e) Segmentation of the portal image in (d) as estimated by the MIME algorithm.

function. On the down side, the true segmentation of the portal image is rarely, if at all, available. Thus, in the MIME algorithm, the segmentation of the portal image is estimated simultaneously, using the current estimates of the transformation parameters and vice–versa, each helping the other.

After verifying our working hypothesis using the true segmentation of the portal images, we will next compare the MI and MIME algorithms where the MIME algorithm iteratively estimates the segmentation information. This estimated segmentation information will be used to plot the MIME energy function and will be compared to the energy function of the MI algorithm. Figure 6.4 shows the example images used for comparing

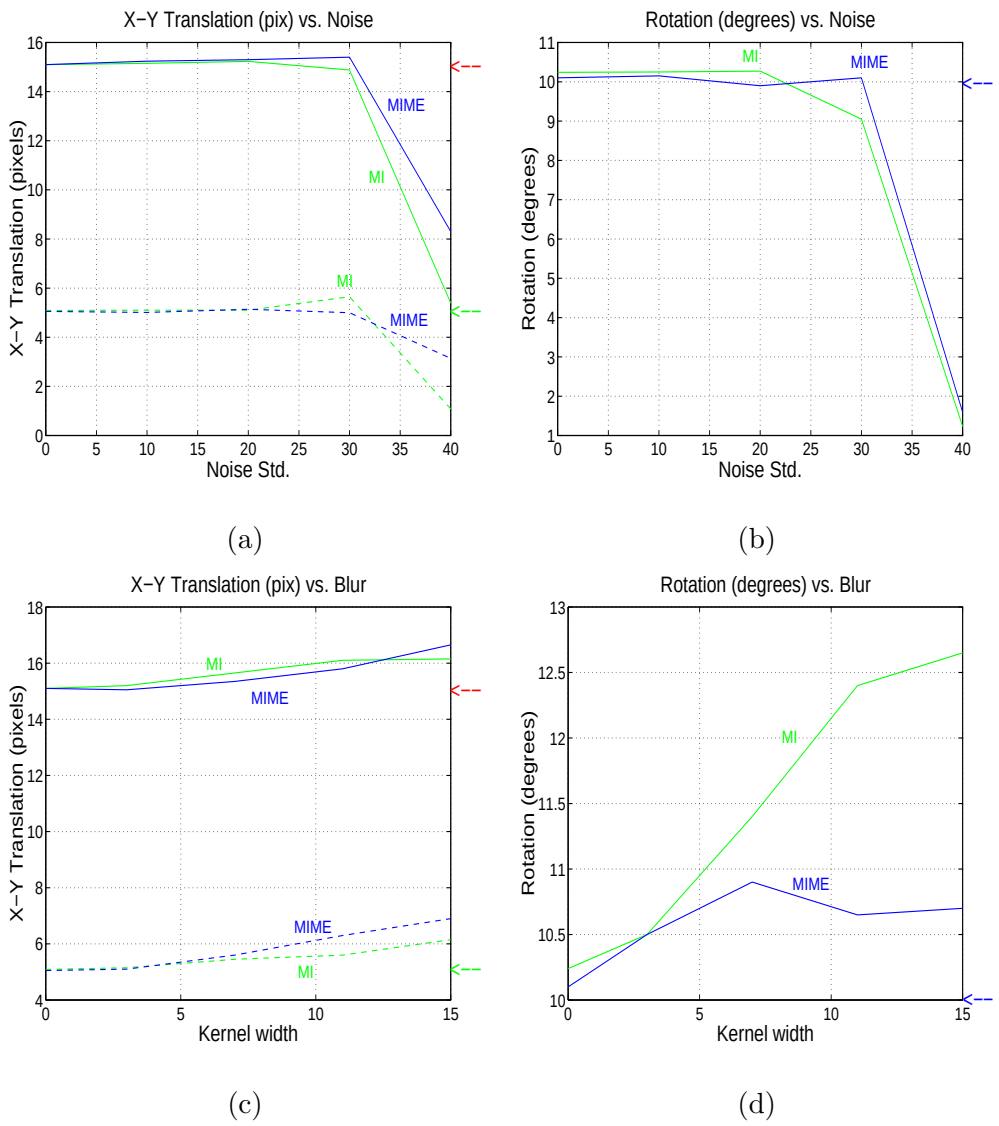


Figure 6.5: Parameters estimated by the two algorithms. MI: mutual information based algorithm. MIME: minimax entropy algorithm. Corresponding graphs and labels are color encoded. Using noisy portal images (a) estimated X, Y translations. Solid lines are the estimated X–translations. Dash lines are the estimated Y–translations. (b) estimated in–plane rotations. Using blurred portal images (c) estimated X, Y translations. (d) estimated in–plane rotations. Arrows on the right are the true parameters, which are 15 (red ←) and 5 (green ←) pixels translation along X and Y axis, respectively, and 10° (blue ←) in–plane rotation.

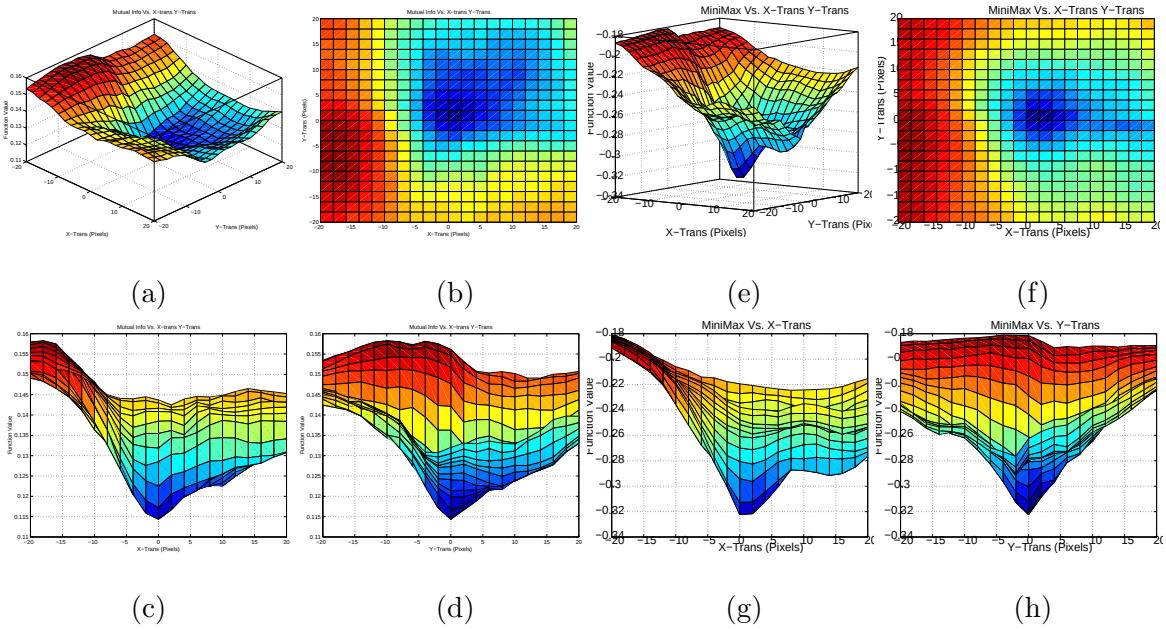


Figure 6.6: Surface plots of the MI and MIME energy functions, respectively, for the noisy portal image. (a), (b), (c), (d) are the plots of MI based energy as a function of X and Y translation respectively. (e), (f), (g), (h) are the corresponding plots of the MIME energy function. The MIME algorithm implicitly used the estimated segmentation of the portal image.

and evaluating the performance of the MI and MIME algorithms on two dimensional images. The noise added to the simulated portal image is i.i.d Gaussian distributed with zero mean and standard deviation (σ) ranging from 0, i.e. no noise, 10.0, 20.0, 30.0, and 40.0. To obtain blurred simulated portal images, blurring kernels of widths 3, 7, 11, 15 are used. The true registration parameters between the portal and the simulator images are 15 pixels translation along the X–axis, 5 pixels along Y–axis and 10° in–plane rotation. Different amount of mis-registration along the X–axis and Y–axis were purposefully selected. This should not be taken to mean that the algorithm can estimate only small

mis-registrations along the Y–axis. Figure 6.5 shows the transformation parameters estimated by the two algorithms under increasing noise and blur. Figures 6.5 (b) and (d) show that the MIME algorithm is more robust and accurate in estimating rotation angle as compared to the MI algorithm, both in the presence of noise and blur. However, figures 6.5 (a) and (c) show that the performance of the two strategies is comparable when estimating the translation parameters.

Further, we have plotted the energy functions for the two algorithms, for the example when i.i.d Gaussian noise of standard deviation (σ) 20.0 is added to the portal image. Figures 6.4 (b), (d) show the images used in plotting the MI energy functions. The MIME algorithm, in addition, implicitly uses the segmentation information, shown in figure 6.4 (e). This segmentation is iteratively estimated by the MIME algorithm. The corresponding energy functions are shown in figure 6.6. Note that the MIME energy function is more peaked with a global optimum at the true transformation parameters. For the MI algorithm, the global optimum for X–translation and Y–translation, figure 6.6 (b), is shifted from the true value.

6.4 Comparisons for 2D/3D Image Registration

In this subsection, we compare the accuracy and the robustness of three registration algorithms, the ridge (GL) based, MI and MIME, first by using a single anterior–posterior (AP) portal image and then by using anterior–posterior (AP) and left–lateral (LL) simulated portal images and the 3DCT dataset. Six transformation parameters, three translations and three rotations of the 3DCT data set, need to be estimated. The simulated portal images are obtained by first transforming the 3DCT data set by translating it along the X–axis by 15 voxels, Y–axis by 5 voxels and rotating it about the Z–axis by

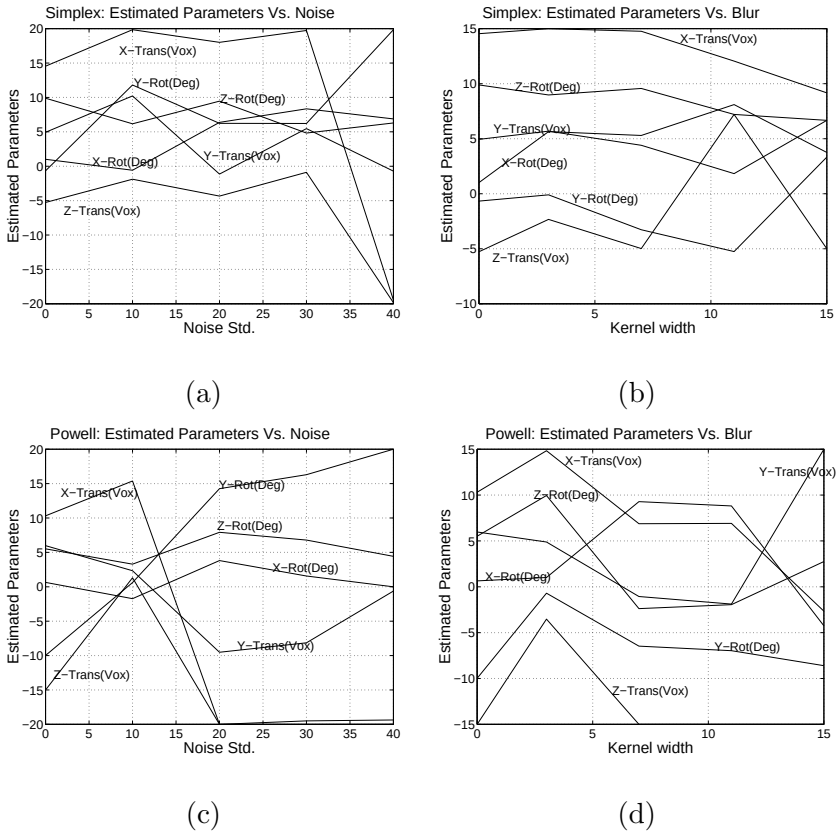


Figure 6.7: Parameters estimated by the ridge–based [44] algorithm. (a), (b) show estimates using the downhill simplex method with increasing noise and blur respectively. (c), (d) show the estimates when the Powell's conjugate direction optimization method is employed. True transformation parameters are 15 voxels translation along the X–axis, 5 voxels along the Y–axis and 10° in–plane rotation.

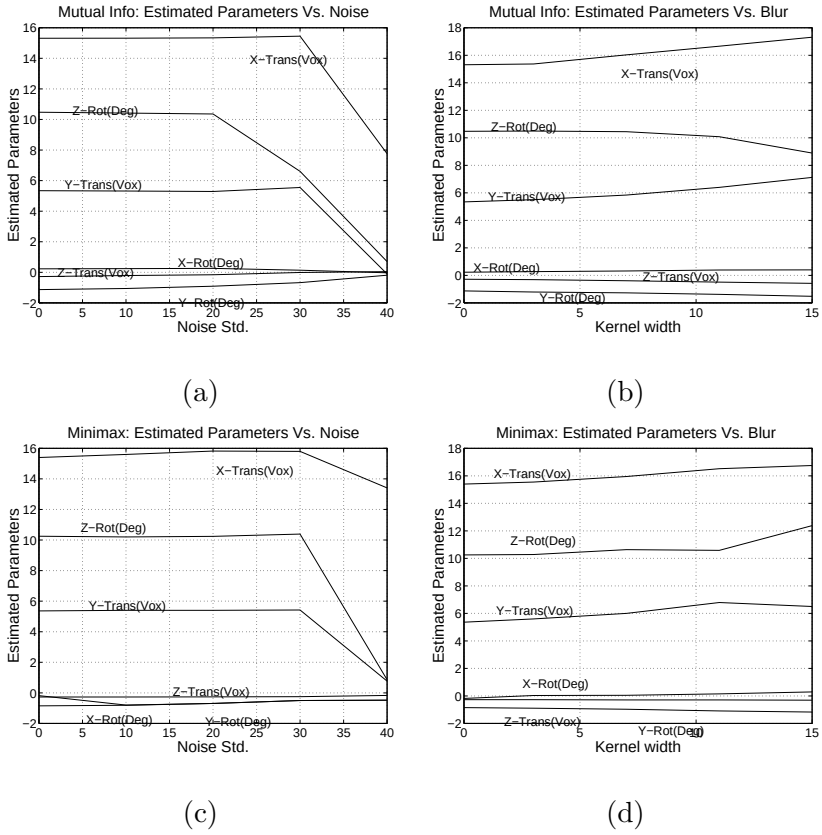
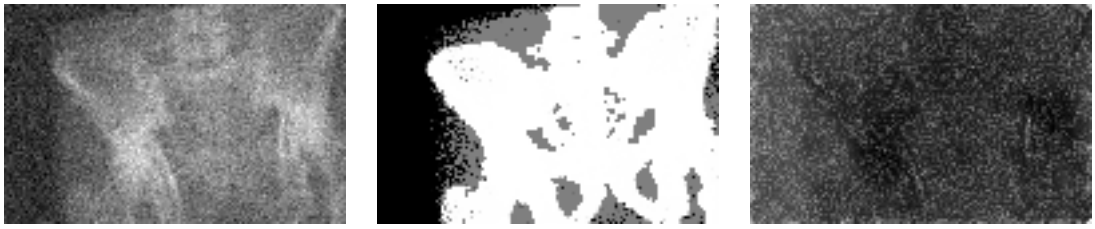
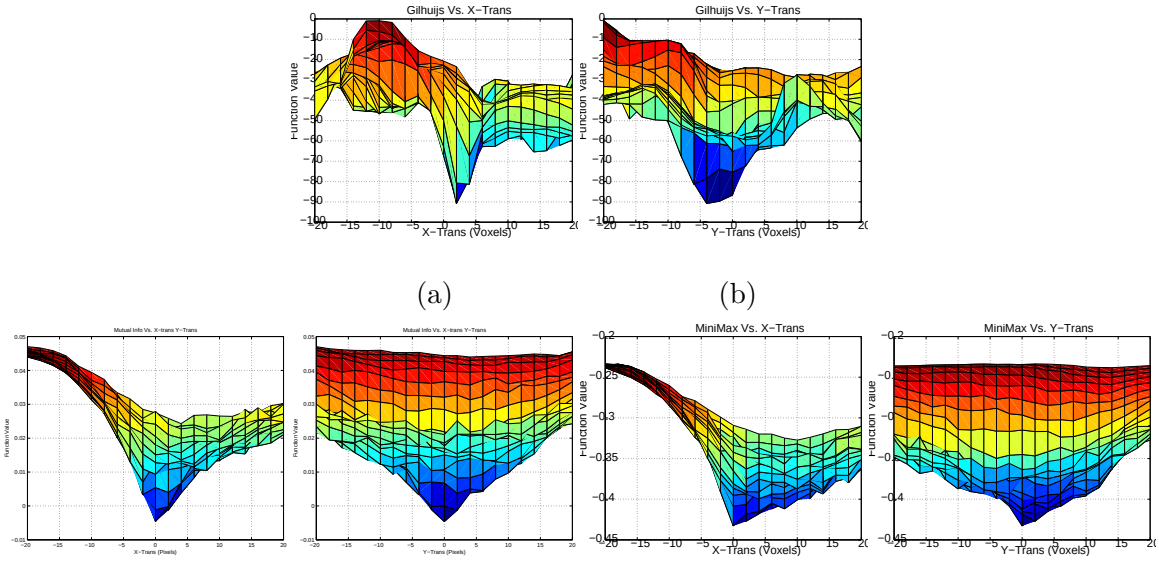


Figure 6.8: Parameters estimated by (a), (b) MI algorithm under increasing noise and blur respectively. (c), (d) MIME algorithm with increasing noise and blur respectively. Parameters to be estimated are $t_x = 15voxels$, $t_y = 5voxels$, $t_z = 0voxels$, $\theta_{XY} = 10^o$, $\theta_{YZ} = 0^o$ and $\theta_{XZ} = 0^o$.



(a) (b) (c)

Figure 6.9: (a) Simulated portal image with i.i.d. Gaussian noise of std. (σ) 20.0. (b) Segmentation of the image in (a) estimated by the MIME algorithm. (c) Morphological top–hat (ridge) transformation of the image in (a).



(a) (b)

(c) (d) (e) (f)

Figure 6.10: (a), (b) are plots of the ridge based method energy as a function of X and Y translation respectively. (c), (d) are the corresponding plots of MI based energy function. (e), (f) are the corresponding plots of the MIME energy function. Energy functions are plotted for the images in figure (6.9). The MIME algorithm implicitly uses the estimated segmentation of the portal image.

10°. Then DRRs in the orthogonal, AP and LL, directions are rendered by projecting the 3DCT data set under known perspective parameters. The simulated portal images are obtained by adding varying amounts of i.i.d Gaussian noise and blur to the DRRs.

6.4.1 3D Analysis Using Single Portal Image

Figure 6.7 shows the transformation parameters estimated by the Gilhuijs' [44] ridge–based registration algorithm with increasing noise and blur in the simulated portal image. The parameters are estimated using both, the downhill simplex method and the Powell's conjugate directions method. The optimization strategy is restarted once to escape the local minima. For both optimization strategies, the algorithm accurately estimates the registration parameters in the absence of noise and blur, except for estimation of the translation along the Z–axis. However, the algorithm is not robust to either noise or blur. With increasing noise and blur in the portal images, human intervention is required for detection and localization of the ridges in the portal images for any meaningful estimation of parameters. Otherwise, as figure 6.7 shows, if the ridges are automatically chosen, the estimated parameters can vary over a large range. This behavior can be attributed to the fact that with increasing noise or blur, many spurious ridge points are detected by the automated ridge enhancement transformation of the portal images.

Figure 6.8 shows the transformation parameters estimated using the mutual information (MI) and the minimax entropy (MIME) algorithm. As for the two–dimensional analysis, the MIME algorithm is more robust to noise as compared to the MI algorithm. The performance of the two algorithms is comparable under increasing blur.

We have plotted the energy functions of the three algorithms as a function of mis–registration along the X–axis and Y–axis. Figure 6.9 (a) shows the simulated portal image used for evaluating the energy function. The energy functions are evaluated as

follows. From the true registered position, the 3DCT data set is translated along X–axis and the Y–axis. At each new translated position the energy functions are evaluated and plotted. The MIME algorithm implicitly uses the estimated segmentation of the portal image, shown in figure 6.9 (b). The segmentation was previously estimated by the MIME algorithm. Figure 6.9 (c) shows the top–hat transformed portal image. Note that in this transformed image it is very difficult to automatically pick the ridge points, necessitating human interaction in detection and localization of the ridge points. The energy functions are shown in the figure 6.10. Note that both MIME and MI algorithms have their global optima at the true registration parameters. Also note that though the ridge based method had poor estimates for the transformation parameters at this noise level ($\sigma = 20.0$), its global optimum is close to the true parameters. The performance of the algorithm described in [44] may therefore be improved by using an optimization method other than downhill simplex method.

6.4.2 3D Analysis Using Dual Portal Images

The previous chapter showed that the in–plane parameter estimations are more accurate than the out–of–plane parameter estimations. Therefore, for patient setup verification, using portal images obtained from two or more different views will likely lead to more accurate and robust results. We have extended the MIME and MI algorithms to use anterior–posterior (AP) and left–lateral (LL) portal images. The ridge–based (GL) algorithm [44] registers two orthogonal, AP and LL, portal images to the 3DCT dataset to estimate the transformation parameters. Example simulated AP and LL portal images are shown in the figures 6.12 (a), (b), with i.i.d Gaussian noise of σ 20.0 is added.

Figures 6.11 (a), (b) , 6.12 (a), (b) and 6.13 (a), (b) show example simulated portal images in the AP and the LL directions. Corresponding ridge–enhanced portal

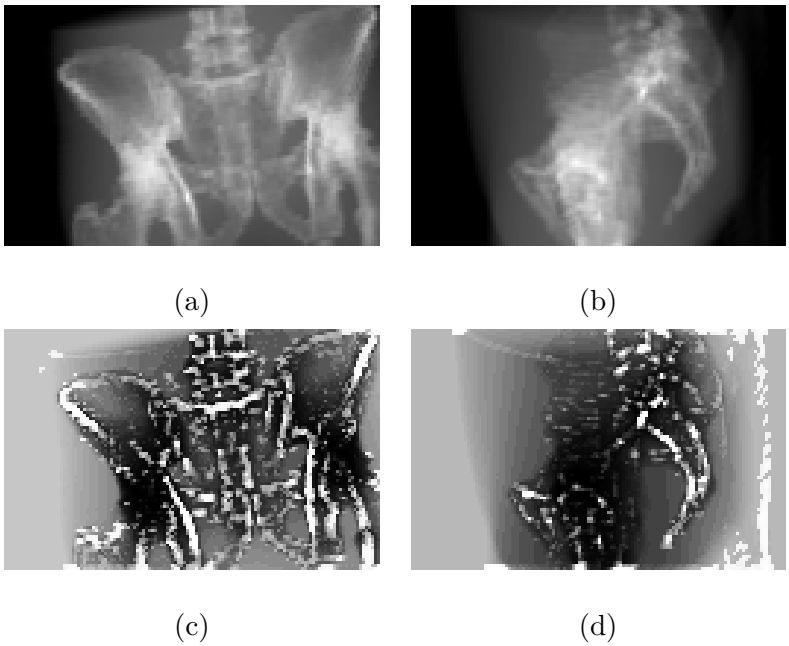


Figure 6.11: (a), (b) Digitally reconstructed radiographs (DRRs) rendered in anterior–posterior (AP) and left–lateral (LL) directions, respectively. (c), (d) the ridge–enhanced AP and LL DRRs, respectively, using the morphological *top–hat* operator.

images are shown in the figures 6.11 (c), (d) , 6.12 (c), (d) and 6.13 (c), (d). These figures show the difficulty in automatically choosing the ridge features from the ridge enhanced portal images and hence point to the necessity of human interaction in the ridge–based registration algorithm.

Figure 6.14 shows the parameters estimated by the ridge–based (GL) algorithm, for both increasing noise and increasing blur in the portal images. Figures 6.14 (a), (b) shows the parameters estimated by using the simplex method of optimization, whereas, in the figures 6.14 (c), (d), the parameters were estimated using the Powell's methods. These graphs show that the simplex method leads to more accurate estimation of

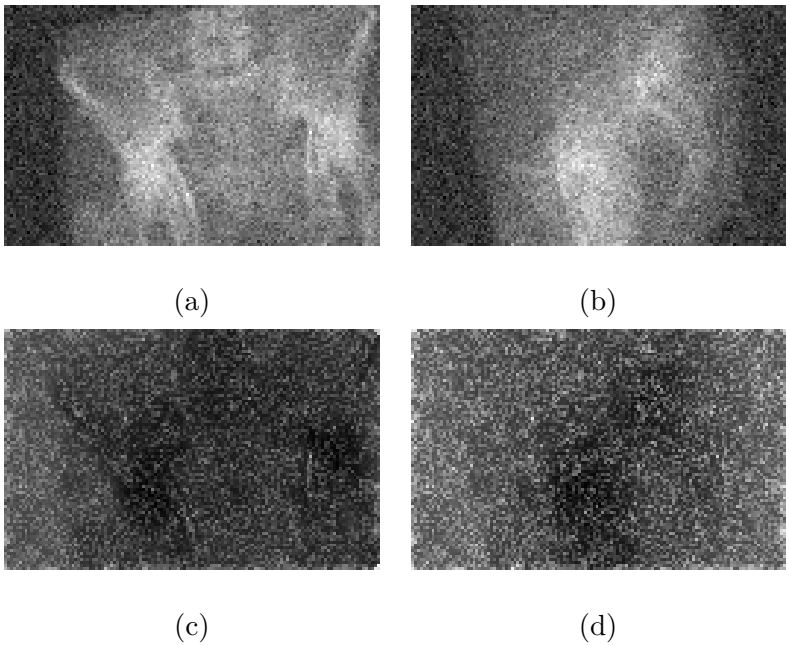


Figure 6.12: (a), (b) Simulated AP and LL portal images, respectively, with i.i.d. Gaussian noise of standard deviation (σ) 20.0. (c), (d) Corresponding ridge–enhanced AP and LL simulated portal images.

the transformation parameters. Thus, for comparing the ridge–based (GL) registration methods to the MI and MIME registration methods, results using the simplex methods of optimization will be used.

Figure 6.16 shows the parameters estimated by the three algorithms, GL , MI and MIME algorithms, with increasing noise and blur. For the ridge–based algorithm, the parameters are estimated using the downhill simplex method. The optimization strategy is restarted once, from the first found optimum, to escape local optima. Stochastic gradient descent is used to estimate the parameters in the MI and MIME algorithms. As with the two–dimensional analysis, the MIME algorithm seems to be more robust

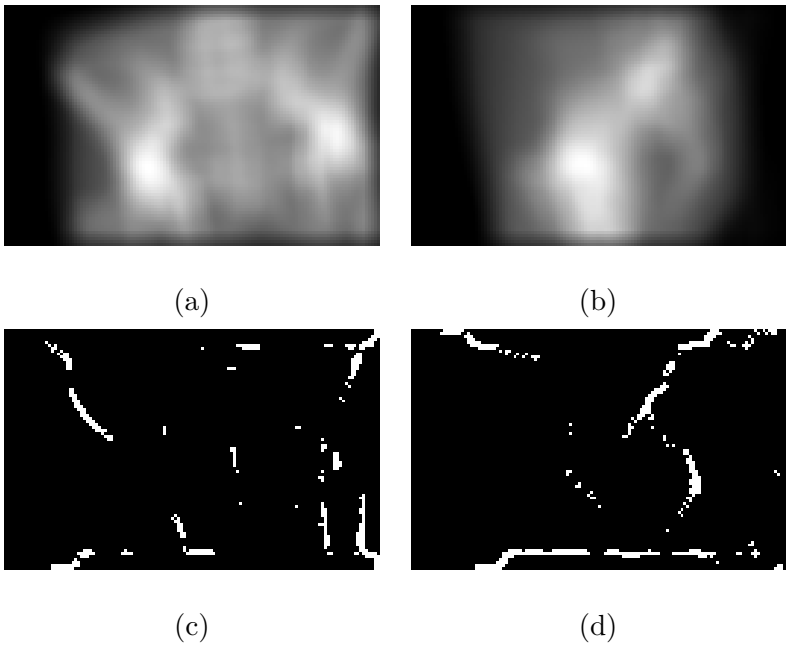


Figure 6.13: (a), (b) AP and LL portal images, simulated by blurring the DRRs by 11 pixels. (c), (d) show the corresponding ridge–enhanced AP and LL portal images, using the morphological top–hat operator.

to noise as compared to the MI algorithm. The performance of the two algorithms is comparable under increasing blur. The GL algorithm leads to accurate parameter estimation in the absence of noise and blur. However, with increasing noise and blur, the parameters estimated by this algorithm seem to be random, requiring human intervention for detection and localization of the ridge–features, for meaningful estimation of the parameters.

As in the two–dimensional analysis, we expect the MI energy function to be increasingly flat and have many local optima with increasing noise. On the other hand, the MIME energy function is expected to have a well–defined global optimum at the true

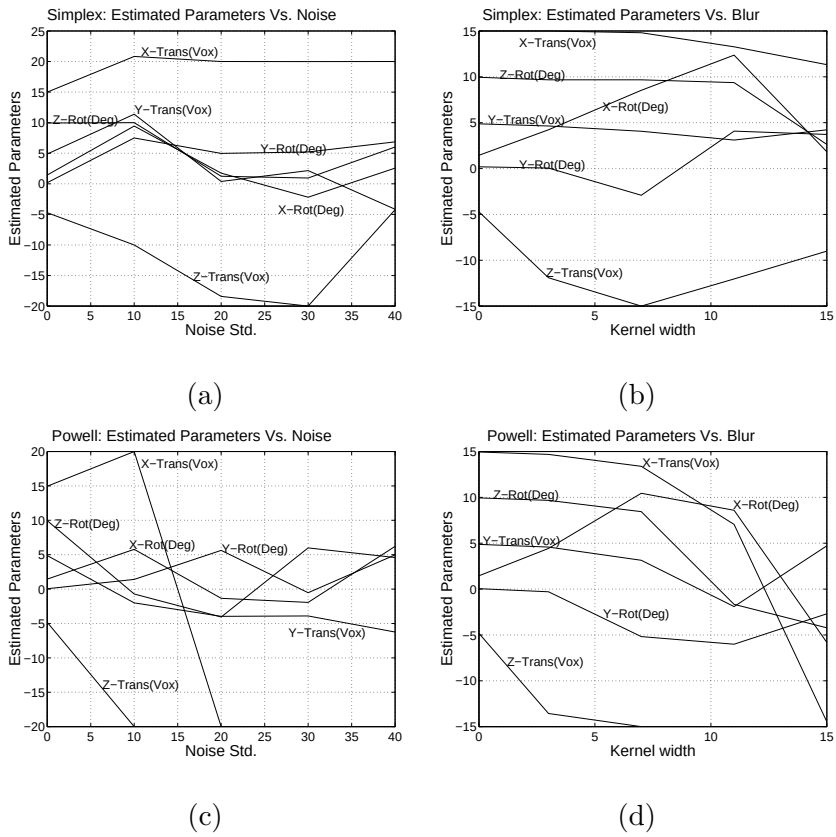


Figure 6.14: Parameters estimated by the ridge–based [44] algorithm using dual AP and LL portal images. (a), (b) show estimates using the downhill simplex method with increasing noise and blur respectively. (c), (d) show the estimates when the Powell's conjugate direction optimization method is employed. True transformation parameters are 15 voxels translation along the X–axis, 5 voxels along the Y–axis and 10° in–plane rotation.

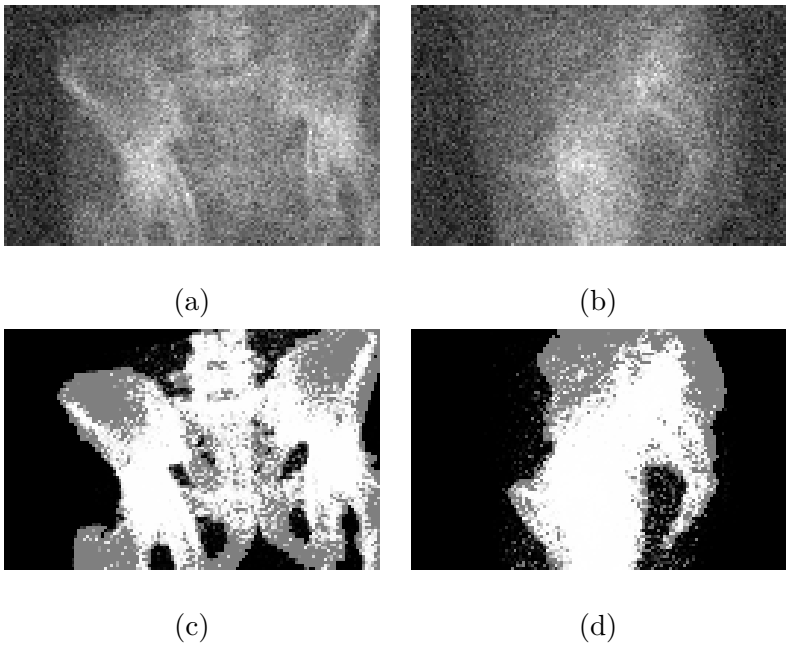
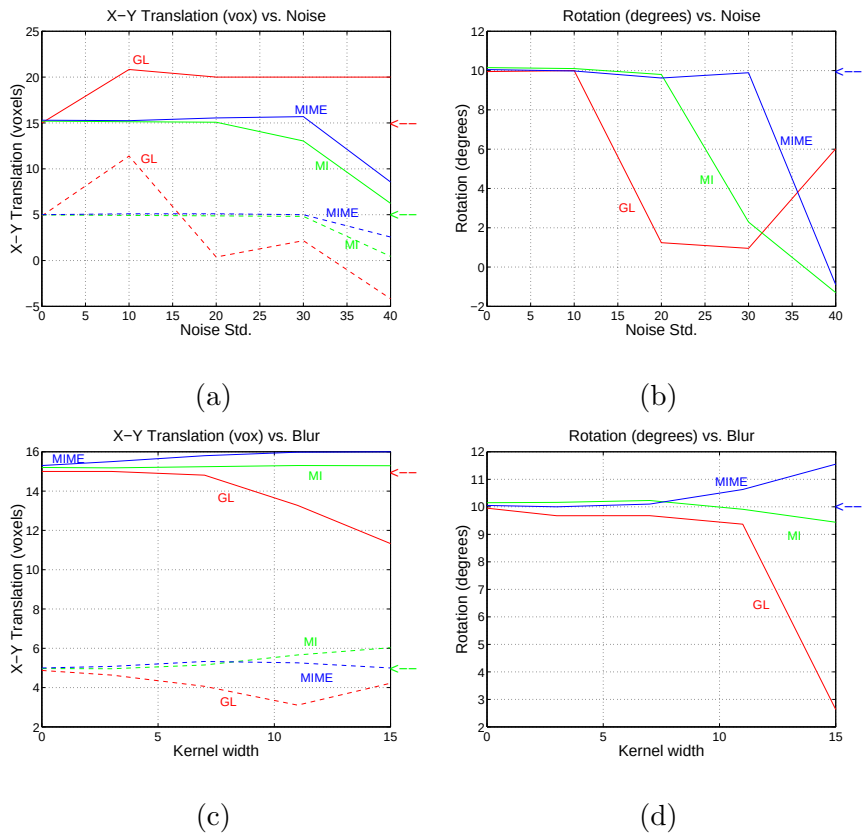


Figure 6.15: (a), (b) show the simulated AP and LL portal images, respectively, with Gaussian noise of std. (σ) 20.0. (c), (d) show the estimated segmentation, by the MIME algorithm, of the AP and the LL portal images, respectively.

registration parameters. To verify this hypothesis, we plotted anecdotal snapshots of the energy functions in the parameter search space around the true optima. Figures 6.15 (a), (b) show the dual simulated portal images used for plotting the energy functions. The minimax entropy algorithm implicitly uses the segmentation information shown in figure 6.15(c) and (d). Figures 6.17 (c), (d), (e) and (f) show the energy functions of the MI and MIME algorithm for this dataset. The true parameters are (0,0). Note that for MIME, the energy function is peaked, with the global optimum at the true parameters. The MI energy function is flatter with the global optimum for the X–translation being shifted from the true value. It is instructive to see the energy function for the ridge based



(a) (b) (c) (d)

Figure 6.16: Parameters estimated using dual portal images. Ridge–based (GL) method used simplex method of optimization to estimate the registration parameters. The dotted lines in figures (a) and (c) show the estimated Y–translation, t_y. The solid lines are the estimated X–translation, t_x. The arrows show the true parameters which are $t_x = 15$ voxels, $t_y = 5$ voxels and $\theta_{XY} = 10^o$.

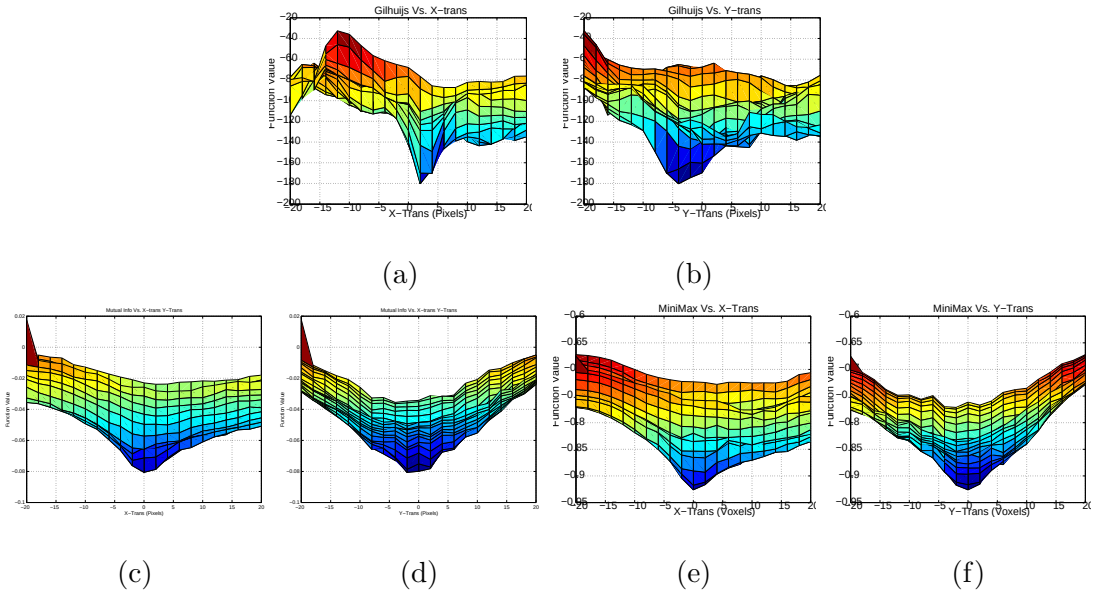


Figure 6.17: (a), (b) are plots of the ridge based method energy, using dual AP and LL simulated portal images, as a function of X and Y translation respectively. (c), (d) are the corresponding plots of MI based energy function. (e), (f) are the corresponding plots of the MIME energy function. Energy functions are plotted for the dual portal images shown in figure 6.15 (a), (b). The MIME algorithm implicitly uses the segmentation information shown in figure 6.15 (c), (d).

algorithm [44], in figures 6.17 (a) and (b). Note that the energy function has a global optimum close to the true registration parameters. Still, the simplex method was unable to find this optimum. This indicates that an optimization strategy other than simplex might lead to better registration results.

6.5 Summary

In this chapter, we presented our working hypothesis that incorporating segmentation information in the registration framework can lead to robust and accurate estimation of the registration parameters. The previous chapter developed the registration framework, [4], where the segmentation and registration of the portal images to a 3DCT dataset is done simultaneously and iteratively. Using this framework, and comparing its performance with other proposed automated registration algorithms, we verified our hypothesis under increasing noise and blur.

First, using the true segmentation of the portal image, we verified that segmentation information can lead to a peaked energy function, which is more robust to both noise and blur, as compared to the MI energy function. The MI energy function becomes flatter with increasing blur and noisier, with many local optima, with increasing noise. Then we compared the performance of MIME to MI when the segmentation information is estimated automatically within the MIME framework. These comparisons validate our hypothesis to be true.

The results from three–dimensional analysis also support our hypothesis, that MIME is more robust to noise as compared to the MI algorithm. For blur, the performance of the two algorithms is comparable, with MIME leading to more accurate estimates of the rotation parameters. The ridge–based (GL) algorithm accurately estimates the transformation parameters in the absence of noise or blur. However, with increasing noise and blur the ridge–enhanced portal images show many spurious ridges and thus human intervention is needed to obtain meaningful estimates of the parameters.

The ridge–based algorithm points to the fact that the feature–based algorithms can yield accurate registration if the features can be accurately detected and localized.

Therefore, we will extend our work to incorporate edge and whole boundary information in the registration framework, for obtaining better segmentation and accurate registration of the portal images to the 3DCT dataset.

Chapter 7

Incorporating Pixel Intensity Correlations

7.1 Introduction

Mutual information–based registration methods, as proposed in the literature [117, 122, 28], assume that pixel intensities, in the images to be registered, are independent and identically distributed (i.i.d.). Such an assumption is not true in general. The minimax entropy registration framework developed in chapter 5 overcomes the assumption of identical distributions by iteratively and simultaneously estimating the registration parameters and the segmentation of portal images. Pixel intensities in the two segmented regions are assumed to be distributed according to two different density functions. However, the pixels were still assumed to be independently distributed. Such an assumption can lead to poor segmentation of the portal image, especially in the presence of noise in the images.

Images usually consist of regions in which the pixel intensities vary slowly. In other

words, there is a strong correlation between the neighboring pixel intensities in an image, except for the pixels at the boundary of two regions. Markov random fields (MRFs) have been proposed to model such a local dependence between the pixel intensities and have been successfully applied to image segmentation [67, 43]. MRF's allow us to incorporate prior information and thus help to regularize the ill–posed problem of image segmentation.

This chapter develops a minimax entropy registration framework while incorporating correlation among the neighboring pixel intensities in both the *max* step and the *min* step to help better segment and register the portal images to the 3DCT image data. To reduce the computational complexity in estimating the registration parameters, each portal image will be modeled as a 1D Markov random process only, with rows of pixels stacked one after the other. Each pixel intensity is thus modeled to be correlated only to either the pixel immediately before or after it. Since the portal images are to be segmented into two regions, *bone* and *background*, a 1D Ising model [85] is sufficient to model the desired correlation between the pixels. In 1D Ising model, each site in a 1D lattice can take one of the two values (red/white, 1/0, +/-, ...) and neighboring sites have an energetic preference to be the same value. *Line processes* [43] are incorporated into the 1D MRF to model boundaries between the regions to be segmented.

7.1.1 Outline of the Approach

As mentioned above, correlation among the portal image pixel intensities is modeled using a 1D Ising model. Geman *et al.* [43] proposed a simulated annealing approach to estimate the segmentation of the images using MRFs as prior information. Even though the simulated annealing–based optimization method has been proven to converge to a global optimal solution, such a method tends to be computationally expensive. Blake *et al.* [17] proposed a *graduated non–convexity* (GNC) approach for image reconstruction.

Deterministic annealing approaches [90, 68] have also been proposed for image reconstruction. GNC and deterministic annealing based approaches tend to be computationally less expensive, compared to simulated annealing, however, these approaches have remained *ad hoc* and guarantee convergence only to a local optimum.

In our approach, we first model the correlation between the neighboring pixels as a 1D MRF. In order not to over–penalize the boundaries between regions, line processes or edges are introduced into the formulation. The MRF model is used to formulate the prior density function on the portal image segmentation. Using the data term, the posterior density function can then easily be formulated. Since the pixels are correlated, the posterior density function cannot be written in factored form, making it difficult to manipulate it analytically. Thus, to simplify the posterior density function, we first propose to *integrate out* the edges (i.e. sum over all possible combinations of the edges) from the posterior density function and then approximate it, using the maximum entropy principle [58], to a density function which can be written into a factored form. The factored density function can be used to estimate the expected label of each pixel. Mean field approximations [85], using the original posterior density function, of the expected pixel labels are derived to show the goodness of the factored density function. Further, the correlations between neighboring pixel segmentation labels are estimated using a procedure similar to the mean field approximations. Even though the edges have been integrated–out, an estimate on the edges is obtained as a by–product of the optimization process. Once an estimate of the pixel labels is obtained, this information is used in the entropy minimization step, the *min* step, to estimate the transformation parameters. It should be noted that the reason the edges are integrated out, instead of the segmentation labels, is due to the fact that edges are assumed to be independent of each other. Thus,

analytically, it is possible to integrate out the edges in the posterior density function, as shown below, and still write the resulting posterior density function in a closed form expression.

7.2 Mathematical Formulation of the MAX step

The formulation of the minimax entropy registration framework in chapter 5 assumed that the pixels in the images are independently distributed. The formulation of the minimax entropy framework in this chapter overcomes the assumption of independence and models the correlation using a 1D MRF. Line processes or the edges are also introduced into the model to achieve better segmentation. Incorporation of neighboring pixel intensity correlation information in the *max* step will regularize the ill–posed problem of image segmentation, especially in the presence of noise.

As mentioned above, images are assumed to be 1D MRF formed by concatenating rows. Thus, only the pixels immediately preceding or following a pixel in a row are considered to be neighbors. Further, we assume that the first and the last pixel in the image are neighbors. Further, note that as the portal images are to be segmented into two regions only and only the 1st order neighborhood is being considered in the 1D MRF, the estimated segmentation of an image can be modeled using a 1D Ising model. Note that such a 1D model can capture only the vertical edges in the image. The horizontal edges will be lost, that is, they cannot be estimated by the algorithm. To detect horizontal edges also, a 2D Ising model is necessary. However, there are two reasons why only a 1D model is chosen. First, a 2D Ising model will be computationally expensive when estimating the transformation parameters. Second, this is our initial effort.

Let $\mathbf{e}_i$ be a random variable denoting an edge between the i and the $i+1$ pixel,

with $\mathbf{e}_i \in \{0, 1\}$.

$$\mathbf{e}_i = \begin{cases} 1, & \text{if there is an edge} \\ 0, & \text{otherwise} \end{cases}$$

Let, $\mathcal{E}$ denote the set of all edges, that is, $\mathcal{E} = \{\mathbf{e}_i, \forall i = 1, \ldots, N\}$, where N is the number of pixels in the image to be segmented. Thus, a segmentation of a portal image can be described by the set $\{\mathbf{Z}, \mathcal{E}\}$, where $\mathbf{Z}$ is the segmentation of the portal image as defined in chapter 4. Using a 1D Ising model [85], the energy associated with each segmentation of a portal image can be written as:

$$\begin{aligned} E(\mathbf{Z}, \mathcal{E}) &= -\left[\frac{1}{2}\sum_i \left(\sum_{v \in A} z_{vi} z_{vi+1}\right)(1 - e_i) + \lambda \sum_i e_i + \sum_i \left(\sum_{v \in A} h_{vi} z_{vi}\right)\right] \\ &= \sum_i \left[E_{1i}(\mathbf{Z}) + E_{2i}(\mathbf{Z}, \mathcal{E})\right] \end{aligned} \tag{7.1}$$

where,

$$\begin{aligned} E_{1i}(\mathbf{Z}) &\equiv -\sum_{v \in A} h_{vi}\, z_{vi} \\ E_{2i}(\mathbf{Z}, \mathcal{E}) &\equiv -\left[\frac{1}{2}\left(\sum_{v \in A} z_{vi} z_{vi+1}\right)(1 - e_i) + \lambda e_i\right] \end{aligned} \tag{7.2}$$

The optimal configuration is the one which minimizes the energy function in the equation (7.1). The term $E_{1i}(\mathbf{Z})$ is also sometimes called the *data term* in the image processing literature. We define $h_{vi} \equiv \ln p(\mathbf{m}_i = v | x_i, y_i')$, called the *external field*, in the data term. Using this definition of the external field, it can be easily shown that the *max* step developed in this chapter reduces to the *max* step formulation in the previous chapter. The

term $E_{2i}(\mathbf{Z}, \mathcal{E})$ has a regularization effect on the estimated segmentation. The first term on the right hand side of equation (7.2) favors same segmentation labels for the neighboring pixels. If there is an edge between two pixels, then the two pixels can have different labels. λ acts as a weighting factor. For large values of λ, a configuration with many edges is favored and vice–versa for small values. A simple analysis, see appendix D.4, puts a bound on possible values of λ to be, $0.0 < \lambda < 0.5$.

Exploiting equivalence between Gibbs distribution and MRF [43], the posteriori probability density function of the segmentation of a portal image can be written as a Gibbs distribution:

$$\begin{aligned}
P^{Gb}(E(\mathbf{Z}, \mathcal{E})) &= \frac{1}{Z(\beta)} \exp\left[-\beta\, E(\mathbf{Z}, \mathcal{E})\right] \\
&= \frac{1}{Z(\beta)} \exp\left[-\beta \sum_i \left(E_{1i}(\mathbf{Z}) + E_{2i}(\mathbf{Z}, \mathcal{E})\right)\right] \\
&= \frac{1}{Z(\beta)} \exp\left[-\beta \sum_i E_{1i}(\mathbf{Z})\right] \exp\left[-\beta \sum_i E_{2i}(\mathbf{Z}, \mathcal{E})\right] \\
&= \frac{1}{Z(\beta)} \exp\left[-\beta \sum_i E_{1i}(\mathbf{Z})\right] \prod_i \exp\left[-\beta\, E_{2i}(\mathbf{Z}, \mathcal{E})\right] \qquad (7.3)
\end{aligned}$$

where β is inverse of the temperature. The probability density function, in equation (7.3), is difficult to manipulate analytically for two reasons. First, the density function $P^{Gb}(E(\mathbf{Z}, \mathcal{E}))$ cannot be further factored as the neighboring pixels are correlated. Second, the presence of edges requires estimation of both the segmentation labels and the edges, thus increasing the computational complexity of the problem. Thus, to simplify the posteriori probability density function, we first propose to *integrate out* the edges and then approximate the resulting density function by a probability density function that can be written in a factored form. The principle of maximum entropy [58] is employed to determine the

simplified density function.

7.2.1 Integrating out Edge Information

The posterior distribution of a segmentation of a portal image, given by the Gibbs distribution in equation (7.3), is first simplified by integrating out the edges. As will be shown below, although the edges have been integrated out, an estimate of the edges will be obtained as a by–product of the optimization process. Thus, integrating out the edges from the probability density function $P^{Gb}(E(\mathbf{Z}, \mathcal{E}))$, we have,

$$
\begin{aligned}
&P^{Gb}(E(\mathbf{Z})) \\
&= \sum_{\mathcal{E}} P^{Gb}(E(\mathbf{Z}, \mathcal{E})) \\
&= \sum_{\mathcal{E}} \frac{1}{Z(\beta)} \exp\left[-\beta\, E(\mathbf{Z}, \mathcal{E})\right] \\
&= \frac{1}{Z(\beta)} \exp\left[-\beta \sum_i E_{1i}(\mathbf{Z})\right] \sum_{\mathcal{E}} \prod_i \exp\left[-\beta\, E_{2i}(\mathbf{Z}, \mathcal{E})\right] \\
&= \frac{1}{Z(\beta)} \exp\left[-\beta \sum_i E_{1i}(\mathbf{Z})\right] \prod_i \sum_{\mathbf{e}_i \in \{0,1\}} \exp\left[-\beta\, E_{2i}(\mathbf{Z}, \mathcal{E})\right] \\
&= \frac{1}{Z(\beta)} \exp\left[-\beta \sum_i E_{1i}(\mathbf{Z})\right] \prod_i \sum_{\mathbf{e}_i \in \{0,1\}} \exp\left[\beta \left(\frac{1}{2}\left(\sum_{v \in A} z_{vi} z_{vi+1}\right)(1 - e_i) + \lambda e_i\right)\right] \\
&= \frac{1}{Z(\beta)} \exp\left[-\beta \sum_i E_{1i}(\mathbf{Z})\right] \prod_i \left[\exp\left(\beta \frac{1}{2} \sum_{v \in A} z_{vi} z_{vi+1}\right) + \exp(\beta\lambda)\right] \\
&= \frac{1}{Z(\beta)} \exp\left[-\beta \sum_i E_{1i}(\mathbf{Z})\right] \exp\left[\beta \frac{1}{2} \sum_i \sum_{v \in A} z_{vi} z_{vi+1}\right] \cdot \\
&\qquad \prod_i \left[1 + \exp\left(-\beta \left(\frac{1}{2} \sum_{v \in A} z_{vi} z_{vi+1} - \lambda\right)\right)\right] \\
&= \frac{1}{Z(\beta)} \exp\left[-\beta \sum_i E_{1i}(\mathbf{Z})\right] \exp\left[\beta \frac{1}{2} \sum_i \sum_{v \in A} z_{vi} z_{vi+1}\right] \cdot
\end{aligned}
$$

$$
\begin{aligned}
&\prod_i \exp\left[\ln\left\{\left[1+\exp\left(-\beta\left(\frac{1}{2}\sum_{v\in A} z_{vi}z_{vi+1}-\lambda\right)\right)\right]\right\}\right] \\
=\ & \frac{1}{Z(\beta)}\ \exp\left[-\beta\ \sum_i\left(E_{1i}(\mathbf{Z})-\frac{1}{2}\sum_{v\in A} z_{vi}z_{vi+1}-\right.\right. \\
&\left.\left.\frac{1}{\beta}\ln\left\{\left[1+\exp\left(-\beta\left(\frac{1}{2}\sum_{v\in A} z_{vi}z_{vi+1}-\lambda\right)\right)\right]\right\}\right)\right] \\
\equiv\ & \frac{1}{Z(\beta)}\ \exp\left[-\beta\left(E_1(\mathbf{Z})+E_3(\mathbf{Z})+E_4(\mathbf{Z})\right)\right]
\end{aligned}
$$

where $Z(\beta)$ denotes the partition function which is a function only of the temperature parameter β. Also, we define:

$$
\begin{aligned}
E_1(\mathbf{Z}) &\equiv \sum_i E_{1i}(\mathbf{Z}) = -\sum_i\sum_{v\in A} h_{vi}\ z_{vi} \\
E_3(\mathbf{Z}) &\equiv \sum_i E_{3i}(\mathbf{Z}) = -\sum_i\left[\frac{1}{2}\sum_{v\in A} z_{vi}z_{vi+1}\right] \\
E_4(\mathbf{Z}) &\equiv \sum_i E_{4i}(\mathbf{Z}) = -\sum_i\frac{1}{\beta}\ln\left\{\left[1+\exp\left(-\beta\left(\frac{1}{2}\sum_{v\in A} z_{vi}z_{vi+1}-\lambda\right)\right)\right]\right\}
\end{aligned}
$$

Note that we have followed the approach in [68] to integrate out the edges to simplify the probability density function by reducing the number of variables. Thus, the energy of the configuration, with the edges integrated out, is defined as

$$
E(\mathbf{Z}) = E_1(\mathbf{Z}) + E_3(\mathbf{Z}) + E_4(\mathbf{Z}) \tag{7.4}
$$

and the probability density function of a particular segmentation of a portal image can now be written as:

$$P^{Gb}(E(\mathbf{Z})) = \frac{1}{Z(\beta)} \exp\left[-\beta\, E(\mathbf{Z})\right] \tag{7.5}$$

The resulting Gibbs distribution, if anything, analytically seems to be more complex. To be able to manipulate the density function analytically, it is approximated by a density function which can be written in a factored form into the next section.

7.2.2 Maximum Entropy Estimation

Analytically, $P^{Gb}(E(\mathbf{Z}))$, is still difficult to manipulate because the image pixels are correlated. Thus, $P^{Gb}(E(\mathbf{Z}))$ is *approximated* with a probability density function, $P^{Gb}(E_I(\mathbf{Z}))$, which can be written into a factored form. The principle of maximum entropy [58] is utilized to estimate $P^{Gb}(E_I(\mathbf{Z}))$.

Prior to determining the density function $P^{Gb}(E_I(\mathbf{Z}))$, define a variable l_i to be:

$$l_i \equiv \left(\frac{1}{1 + \exp[-\beta(\frac{1}{2}\sum_{\alpha \in A} z_{\alpha i}\ z_{\alpha i+1} - \lambda)]} \right) \tag{7.6}$$

which can be shown to be the conditional probability that the random variable $\mathbf{e}_i = 0$ (See appendix D.4).

Define an energy function $E_I(\mathbf{Z})$, which is a function of variables ε_{vi}, as

$$E_I(\mathbf{Z}) = -\sum_i \sum_{v \in A} \varepsilon_{vi} z_{vi} \tag{7.7}$$

and the corresponding Gibbs distribution as

$$P^{Gb}(E_I(\mathbf{Z})) = \frac{1}{Z_I(\beta)} \exp\left[-\beta\, E_I(\mathbf{Z})\right] \tag{7.8}$$

Since the energy $E_I(\mathbf{Z})$ is defined as in equation (7.7), the probability density function $P^{Gb}(E_I(\mathbf{Z}))$ in equation (7.8) can be easily written in a factored form. However, we need to determine the variables ε_{vi} in equation (7.7). The variables ε_{vi} are estimated such that the density function $P^{Gb}(E_I(\mathbf{Z}))$ is *close* to the density function $P^{Gb}(E(\mathbf{Z}))$. The maximum entropy principle [58, 59] provides a principled way of estimating a density function under given constraints. Subscript I, standing for Independence, in $E_I(\mathbf{Z})$ is used to emphasize that the probability distribution $P^{Gb}(E_I(\mathbf{Z}))$ can be written in the factored form. Thus, we will follow the maximum entropy principle to estimate the variables ε_{vi}. Before proceeding further, we write the density functions $P^{Gb}(E_I(\mathbf{Z}))$ and $P^{Gb}(E(\mathbf{Z}))$ as:

$$\begin{aligned}
P^{Gb}(E(\mathbf{Z})) &= \frac{1}{Z(\beta)} \exp\left[-\beta\, E(\mathbf{Z})\right] \\
&= \frac{1}{\exp\left(\ln Z(\beta)\right)} \exp\left[-\beta\, E(\mathbf{Z})\right] \\
&= \exp\left[-\beta \left\{ E(\mathbf{Z}) + \frac{1}{\beta} \ln Z(\beta) \right\}\right] \\
&= \exp\left[-\beta \left\{ E(\mathbf{Z}) - F(\beta) \right\}\right]
\end{aligned}$$

where,

$$F(\beta) \equiv -\frac{1}{\beta} \ln Z(\beta)$$

Similarly, we have,

$$\begin{aligned}
P^{Gb}(E_I(\mathbf{Z})) &= \frac{1}{Z_I(\beta)} \exp\left[-\beta\ E_I(\mathbf{Z})\right] \\
&= \frac{1}{\exp\left(\ln Z_I(\beta)\right)} \exp\left[-\beta\ E_I(\mathbf{Z})\right] \\
&= \exp\left[-\beta \left\{E_I(\mathbf{Z}) + \frac{1}{\beta} \ln Z_I(\beta)\right\}\right] \\
&= \exp\left[-\beta \left\{E_I(\mathbf{Z}) - F_I(\beta)\right\}\right]
\end{aligned}$$

where,

$$F_I(\beta) \equiv -\frac{1}{\beta} \ln Z_I(\beta)$$

The variables, ε_{vi}, are estimated such that the density $P^{Gb}(E_I(\mathbf{Z}))$ is a maximum entropy distribution under the constraint that it is close to the true density function, $P^{Gb}(E(\mathbf{Z}))$. This is equivalent to reducing the Kullback–Leibler (KL) divergence between the two distributions, where the KL divergence is given as:

$$\begin{aligned}
KL\left[P^{Gb}(E_I(\mathbf{Z}))||P^{Gb}(E(\mathbf{Z}))\right] &= \sum_{\mathbf{Z}} P^{Gb}(E_I(\mathbf{Z}))\ \ln\left[\frac{P^{Gb}(E_I(\mathbf{Z}))}{P^{Gb}(E(\mathbf{Z}))}\right] \\
&= \beta \sum_{\mathbf{Z}} P^{Gb}(E_I(\mathbf{Z}))\ \left[(F_I(\beta) - F(\beta)) + (E(\mathbf{Z}) - E_I(\mathbf{Z}))\right] \\
&= \beta\left[F_I(\beta) - F(\beta) + <E(\mathbf{Z}) - E_I(\mathbf{Z})>\right] \qquad (7.9)
\end{aligned}$$

where $<E(\mathbf{Z}) - E_I(\mathbf{Z})>$ is defined to be the expected value of $E(\mathbf{Z}) - E_I(\mathbf{Z})$ with respect to the density function $P^{Gb}(E_I(\mathbf{Z}))$. In general, the expected value of a random variable with respect to the density function $P^{Gb}(E_I(\mathbf{Z}))$ will be denoted by $< \cdot >$. The optimal

variables, denoted by ε^*_{vi}, minimize the KL divergence in equation (7.9). Thus, the variables ε^*_{vi} are estimated by solving the equations:

$$\frac{\partial}{\partial \varepsilon_{vi}} KL\left[P^{Gb}(E_I(\mathbf{Z}))||P^{Gb}(E(\mathbf{Z}))\right] = 0 \tag{7.10}$$

Thus, from equations (7.9) and (7.10), we have:

$$\frac{\partial}{\partial \varepsilon_{vi}} \left[F_I(\beta) + <E(\mathbf{Z}) - E_I(\mathbf{Z})>\right] = 0 \tag{7.11}$$

Solving equations (7.11), it can be shown that the optimal parameters, ε^*_{vi}, are given as (see appendix D.1 for the intermediate steps):

$$\varepsilon^*_{\alpha i} = \left[\left\langle \frac{z_{\alpha i-1} + z_{\alpha i+1}}{2} + h_{\alpha i} - \frac{1}{\beta} \ \ln(l_i l_{i-1}) \right\rangle\right] \tag{7.12}$$

Thus, the optimal estimate of the variables $\varepsilon_{\alpha i}$ leads to an estimate of the probability density function $P^{Gb}(E_I(\mathbf{Z}))$ which is *closest*, in the sense of KL divergence, to the density function $P^{Gb}(E(\mathbf{Z}))$ and has maximum entropy. Since $P^{Gb}(E_I(\mathbf{Z}))$ can be written into a factored form, it can be used to find the probability that a pixel i has a label a and also to evaluate the expected value of a label at each pixel.

7.2.2.1 Mean Field Approximation

In the previous section, a density function has been estimated which can be written into a factored form. This estimated density function is optimal in the sense that it is a maximum entropy distribution. A factored form of the density function is desirable as it makes it easy to manipulate it analytically. However, the question still remains as to

how good the estimates are. This section shows, using mean field approximations to the expected value of pixel label, that the maximum entropy estimates of the expected values are accurate up to first order variations.

After integrating out the edges, the energy function of the probability density function, $P^{Gb}(E(\mathbf{Z}))$, given by the equation (7.4), is:

$$E(\mathbf{Z}) = -\sum_i \left[\sum_{\alpha \in A} h_{\alpha i} z_{\alpha i} + \frac{1}{2} \sum_{\alpha \in A} z_{\alpha i} z_{\alpha i+1} - \frac{1}{\beta} \ln l_i \right]$$

Let $E_t(\mathbf{Z})$ denote all the terms in the above summation except for the terms containing $z_{\alpha i}$. We can rewrite the energy function as follows:

$$\begin{aligned}
E(\mathbf{Z}) &= -\left[E_t(\mathbf{Z}) + \sum_{\alpha \in A} h_{\alpha i} z_{\alpha i} + \sum_{\alpha \in A} \frac{z_{\alpha i-1} + z_{\alpha i+1}}{2} z_{\alpha i} - \frac{1}{\beta} \ln l_{i-1} - \frac{1}{\beta} \ln l_i \right] \\
&= -\left[E_t(\mathbf{Z}) + \sum_{\alpha \in A} h_{\alpha i} z_{\alpha i} + \sum_{\alpha \in A} \frac{z_{\alpha i-1} + z_{\alpha i+1}}{2} z_{\alpha i} - \frac{1}{\beta} \ln(l_{i-1} l_i) \sum_{\alpha \in A} z_{\alpha i} \right] \\
&= -\left[E_t(\mathbf{Z}) + \sum_{\alpha \in A} z_{\alpha i} \left(h_{\alpha i} + \frac{z_{\alpha i-1} + z_{\alpha i+1}}{2} - \frac{1}{\beta} \ln(l_{i-1} l_i) \right) \right] \\
&= -\left[E_t(\mathbf{Z}) + \sum_{\alpha \in A} z_{\alpha i} \tilde{\varepsilon}_{\alpha i} \right]
\end{aligned}$$

where $\tilde{\varepsilon}_{\alpha i}$ is as defined to be:

$$\tilde{\varepsilon}_{\alpha i} \equiv \left[\frac{z_{\alpha i-1} + z_{\alpha i+1}}{2} + h_{\alpha i} - \frac{1}{\beta} \ \ln(l_i l_{i-1}) \right]$$

Thus, the expected value of z_{vi}, defined as $< z_{vi} >_{E(\mathbf{Z})}$, with respect to the density

function $P^{Gb}(E(\mathbf{Z}))$ is given as:

$$
\begin{aligned}
< z_{vi} >_{E(\mathbf{Z})} &= \sum_{\mathbf{Z}} z_{vi} P^{Gb}(E(\mathbf{Z})) \\
&= \frac{\sum_{\mathbf{Z}} z_{vi} \exp(-\beta E(\mathbf{Z}))}{\sum_{\mathbf{Z}'} \exp(-\beta E(\mathbf{Z}'))} \\
&= \frac{1}{\sum_{\mathbf{Z}'} \exp(-\beta E(\mathbf{Z}'))} \sum_{\mathbf{Z}} \exp(-\beta E(\mathbf{Z})) \frac{\sum_{\{Z_i\}} z_{vi} \exp(-\beta E(\mathbf{Z}))}{\sum_{\{Z_i\}} \exp(-\beta E(\mathbf{Z}))} \qquad (7.13)
\end{aligned}
$$

where $\{Z_i\}$ denotes all possible values in the ith row of the matrix $\mathbf{Z}$, that is, $\{(0,1),(1,0)\}$. This follows from the constraints $\sum_{\alpha \in A} z_{\alpha i} = 1$ and $z_{\alpha i} \in \{0,1\}; \forall \alpha \in A$.

The last term in equation (7.13) can be simplified further as:

$$
\begin{aligned}
\frac{\sum_{\{Z_i\}} z_{vi} \exp(-\beta E(\mathbf{Z}))}{\sum_{\{Z_i\}} \exp(-\beta E(\mathbf{Z}))} &= \frac{\sum_{\{Z_i\}} z_{vi} \exp\left(\beta \left(E_t(\mathbf{Z}) + \sum_{\alpha \in A} z_{\alpha i} \tilde{\varepsilon}_{\alpha i}\right)\right)}{\sum_{\{Z_i\}} \exp\left(\beta \left(E_t(\mathbf{Z}) + \sum_{\alpha \in A} z_{\alpha i} \tilde{\varepsilon}_{\alpha i}\right)\right)} \\
&= \frac{\sum_{\{Z_i\}} z_{vi} \exp\left(\beta \sum_{\alpha \in A} z_{\alpha i} \tilde{\varepsilon}_{\alpha i}\right)}{\sum_{\{Z_i\}} \exp\left(\beta \sum_{\alpha \in A} z_{\alpha i} \tilde{\varepsilon}_{\alpha i}\right)} \\
&= \frac{\exp\left(\beta \tilde{\varepsilon}_{vi}\right)}{\sum_{\alpha \in A} \exp\left(\beta \tilde{\varepsilon}_{\alpha i}\right)} \qquad (7.14)
\end{aligned}
$$

Therefore, from equations (7.13) and (7.14) the expected value $< z_{vi} >_{E(\mathbf{Z})}$ can now be written as:

$$
\begin{aligned}
< z_{vi} >_{E(\mathbf{Z})} &= \frac{1}{\sum_{\mathbf{Z}'} \exp(-\beta E(\mathbf{Z}'))} \sum_{\mathbf{Z}} \exp(-\beta E(\mathbf{Z})) \frac{\exp\left(\beta \tilde{\varepsilon}_{vi}\right)}{\sum_{\alpha \in A} \exp\left(\beta \tilde{\varepsilon}_{\alpha i}\right)} \\
&= \sum_{\mathbf{Z}} \frac{\exp\left(\beta \tilde{\varepsilon}_{vi}\right)}{\sum_{\alpha \in A} \exp\left(\beta \tilde{\varepsilon}_{\alpha i}\right)} P^{Gb}(E(\mathbf{Z})) \\
&= \left\langle \frac{\exp\left(\beta \tilde{\varepsilon}_{vi}\right)}{\sum_{\alpha \in A} \exp\left(\beta \tilde{\varepsilon}_{\alpha i}\right)} \right\rangle_{E(\mathbf{Z})} \qquad (7.15)
\end{aligned}
$$

The equation (7.15) is also called the *Markov blanket identity*. Neglecting second and

higher order terms (see appendix D.3 for details), and assuming that the variation in $\tilde{\varepsilon}_{vi}$ (denoted by $\delta\tilde{\varepsilon}_{vi} = \tilde{\varepsilon}_{vi} - < \tilde{\varepsilon}_{vi} >$) from its mean value $< \tilde{\varepsilon}_{vi} >$ is small, the expected value $< z_{vi} >_{E(\mathbf{Z})}$ can now be approximated to be, see appendix D.3 for details (also note that the mean value of the variation $\delta\tilde{\varepsilon}_{vi}$ is zero, that is, $< \delta\tilde{\varepsilon}_{vi} >=< \tilde{\varepsilon}_{vi} - < \tilde{\varepsilon}_{vi} >>= 0$):

$$\begin{aligned} < z_{vi} >_{E(\mathbf{Z})} &\approx \frac{\exp\left(\beta < \tilde{\varepsilon}_{vi} >\right)}{\sum_{\alpha \in A} \exp\left(\beta < \tilde{\varepsilon}_{\alpha i} >\right)} \\ &= \frac{\exp\left(\beta \varepsilon^*_{vi}\right)}{\sum_{\alpha \in A} \exp\left(\beta \varepsilon^*_{\alpha i}\right)} \end{aligned} \tag{7.16}$$

Thus, the mean field estimate of the expected value $< z_{vi} >_{E(\mathbf{Z})}$, in equation (7.16), obtained by ignoring the second and higher order variations, is exactly the same as the expected value of z_{vi} estimated using the density function $P^{Gb}(E_I(\mathbf{Z}))$. Thus, the maximum entropy density function $P^{Gb}(E_I(\mathbf{Z}))$ is accurate up to first order variations. The relation between the expected value, $< z_{vi} >_{E(\mathbf{Z})}$, and $P^{Gb}(E_I(\mathbf{Z}))$ is easy to show as the variable $\mathbf{z}_{vi}$ takes on only two values, $\{0, 1\}$ and hence,

$$\begin{aligned} < z_{vi} >_{E(\mathbf{Z})} &= \sum_{\mathbf{Z}} z_{vi} \;\; P^{Gb}(E(\mathbf{Z})) \\ &= \sum_{\alpha \in \{0,1\}} \alpha \; P(\mathbf{z}_{vi} = \alpha) \\ &= P(\mathbf{z}_{vi} = 1) \end{aligned} \tag{7.17}$$

The density function $P^{Gb}(E(\mathbf{Z}))$ is approximated by $P^{Gb}(E_I(\mathbf{Z}))$ which can be written in a factored form as

$$P^{Gb}(E_I(\mathbf{Z})) = \frac{1}{Z_I(\beta)} \exp\left[-\beta \; E_I(\mathbf{Z})\right]$$

$$
\begin{aligned}
&= \frac{1}{Z_I(\beta)} \exp\left[\beta \sum_i \sum_{v \in A} \varepsilon^*_{vi} z_{vi}\right] \\
&= \frac{\exp\left[\beta \ \sum_i \sum_{v \in A} \varepsilon^*_{vi} z_{vi}\right]}{\sum_Z \exp\left[\beta \ \sum_i \sum_{v \in A} \varepsilon^*_{vi} z_{vi}\right]} \\
&= \frac{\prod_i \exp\left[\beta \sum_{v \in A} \varepsilon^*_{vi} z_{vi}\right]}{\sum_Z \prod_i \exp\left[\beta \sum_{v \in A} \varepsilon^*_{vi} z_{vi}\right]} \\
&= \frac{\prod_i \exp\left[\beta \sum_{v \in A} \varepsilon^*_{vi} z_{vi}\right]}{\prod_i \sum_{v \in A} \exp\left[\beta \varepsilon^*_{vi}\right]} \\
&= \prod_i \frac{\exp\left[\beta \sum_{v \in A} \varepsilon^*_{vi} z_{vi}\right]}{\sum_{v \in A} \exp\left[\beta \varepsilon^*_{vi}\right]} \\
&= \prod_i P_I(Z_i)
\end{aligned}
$$

where the subscript I is used to clarify that the distribution on the ith pixel labels, $P_I(Z_i)$ is according to the distribution $P^{Gb}(E_I(\mathbf{Z}))$ which can be written in the factored form. Therefore, $P_I(\mathbf{z}_{vi} = 1)$ is given as

$$
P_I(\mathbf{z}_{vi} = 1) = \frac{\exp\left[\beta \varepsilon^*_{vi}\right]}{\sum_{v \in A} \exp\left[\beta \varepsilon^*_{vi}\right]} \tag{7.18}
$$

Therefore, approximating $P(\mathbf{z}_{vi} = 1)$ in equation (7.17) by $P_I(\mathbf{z}_{vi} = 1)$ in equation (7.18), the approximated expected value $< z_{vi} >_{E(\mathbf{Z})}$ is given to be:

$$
< z_{vi} >_{E(\mathbf{Z})} \approx \frac{\exp\left[\beta \varepsilon^*_{vi}\right]}{\sum_{v \in A} \exp\left[\beta \varepsilon^*_{vi}\right]}
$$

which is the same as that approximated in equation (7.16) using mean field approximations.

7.2.3 Estimating Correlations

Correlation between the random variables $\mathbf{z}_{vi}, \mathbf{z}_{\alpha j}$, defined as the expected value $< z_{vi} z_{\alpha j} >_{E(\mathbf{Z})}$, is estimated using a procedure similar to the estimation of the expected value of a label using the mean field approximations. Appendix D.2 develops the approximated values of the correlations which are then used in the *min* step of the proposed framework to estimate the transformation parameters. Since $P^{Gb}(E_I(\mathbf{Z}))$ assumes that the pixels labels are independently distributed, this probability distribution function cannot be used to estimated the correlation.

7.3 Formulation of the MIN step

In the *min* step, the registration parameters are estimated by minimizing the joint conditional entropy [4]. Chapter 5 assumes the pixel intensities to be independently distributed. In this chapter, the images are modeled as 1D Markov random process where the neighboring pixel intensities are assumed to be correlated. The correlation among the neighboring pixels is incorporated into the *min* step for estimating the registration parameters.

The transformation parameters are estimated as:

$$\begin{aligned} \hat{T} &= \arg\min_{T} H(M, X|G, T) \\ &= \arg\min_{T} \left[H(X, Y(T)|M) - H(Y(T))\right] \end{aligned} \tag{7.19}$$

where the equation (7.19) follows from the fact that entropy of the segmentation, $H(M)$, remains constant while estimating the transformation parameters, $\mathbf{T}$. Note that $\mathbf{X}$ and $\mathbf{Y}$ denote 1D Markov random processes. The joint, $p(X, Y, M)$, the conditional, $p(X, Y|M)$,

and the marginal, $p(Y)$, density functions are given as:

$$
\begin{aligned}
p(Y) &= \prod_{i=2}^{N} p(y_i|y_{i-1}) \cdot p(y_1) \\
p(X,Y,M) &= p(X,Y|M)P(M) \\
p(X,Y|M) &= p(x_1,x_2,\ldots,x_n,y_1,y_2,\ldots,y_n|m_1,m_2,\ldots,m_n) \\
&= p(x_n,y_n|x_{n-1},y_{n-1},m_n,m_{n-1}) \cdots p(x_2,y_2|x_1,y_1,m_2,m_1) \cdot p(x_1,y_1|m_1) \\
&= \prod_{i=2}^{N} p(x_i,y_i|x_{i-1},y_{i-1},m_i,m_{i-1}) \cdot p(x_1,y_1|m_1) \qquad (7.20)
\end{aligned}
$$

Taking logarithm on both sides of equation (7.20), we get,

$$
\ln p(X,Y|M) = \sum_{i=2}^{N} \ln p(x_i,y_i|x_{i-1},y_{i-1},m_i,m_{i-1}) + \ln p(x_1,y_1|m_1)
$$

Hence, the joint conditional entropy $H(X,Y(T)|M)$ can be shown to be, see appendix D.5.1,

$$
H(X,Y|M) \approx (N-1)\left[\sum_{\{\alpha,\beta\}} H_{\alpha\beta}(x_2,y_2,x_1,y_1) < \overline{z_\alpha z_\beta} > - \sum_{\alpha \in A} H_\alpha(x,y) < \overline{z_\alpha} >\right]
$$

and the entropy of DRR, Y, is approximated to be (The intermediate steps are developed in appendix D.5.2):

$$
\begin{aligned}
H(Y) &= \sum_{i=2}^{N} H(y_i|y_{i-1}) + H(y_1) \\
&\leq \sum_{i=1}^{N} H(y_i) \\
&= N \cdot H(y)
\end{aligned}
$$

To estimate the joint and marginal density functions, a set of pixels is sampled at random from the images. Non–parametric density estimation, using the Parzen window method which places isotropic Gaussian kernels at each sampled pixel intensity, is used to estimate various density functions. A stochastic gradient descent method is implemented to estimate the optimal registration parameters. Appendix D.6 gives details of Parzen window estimates and the derivatives of various terms used in the implementation of the stochastic gradient descent method. Appendix D.6 details the derivatives of various terms for the estimation of covariance matrices of the Gaussian kernels used in the Parzen window estimates of the various density functions.

7.4 Summarizing the Algorithm

To summarize, the *max* step and the *min* step of the minimax entropy registration framework, developed in this chapter, can be written as:

MAX step:

$$< z_{vi} >_{E(\mathbf{Z})} \approx \frac{\exp\left(\beta\varepsilon^*_{vi}\right)}{\sum_{\alpha\in A}\exp\left(\beta\varepsilon^*_{\alpha i}\right)}$$

where,

$$\varepsilon^*_{\alpha i} = \left[\left\langle \frac{z_{\alpha i-1} + z_{\alpha i+1}}{2} + h_{\alpha i} - \frac{1}{\beta}\ \ln(l_i l_{i-1})\right\rangle\right]$$

and the correlations are evaluated as in appendix D.2. The *min* step is developed to be:

MIN step:

$$\hat{T} = \arg\min_{T} \left[H(X, Y(T)|M) - H(Y(T)) \right]$$

where, the various joint and marginal entropy terms are estimated as

$$\begin{aligned} H(X, Y|M) &\approx (N-1) \left[\sum_{\{\alpha,\beta\}} < \overline{z_\alpha z_\beta} > H_{\alpha\beta}(x_2, y_2, x_1, y_1) - \sum_{\alpha \in A} < \overline{z_\alpha} > H_\alpha(x, y) \right] \\ H(Y) &\leq N \cdot H(y) \end{aligned}$$

7.5 Convergence of the Minimax Algorithm

The proposed minimax entropy registration framework is an iterative algorithm and hence it is necessary to show, at least intuitively, that the algorithm will converge to a local optimum. The convergence of the minimax entropy algorithm is easy to see by first considering the convergence of the probability distribution of a segmentation of a portal image. Note that an estimated segmentation distribution, $P^k(M)$, is a function of inverse temperature parameter β. At high temperatures, that is small values of β, the estimated distribution is forced to be uniform. For low temperatures, that is large values of β, the probability that a pixel belongs to a particular class is forced to converge to either a 0 or a 1. Once the probability converges to 0 (or 1), it remains at 0 (or 1) as the estimated probability in the step k is used as a prior probability in the $k+1$th step. Thus, once the estimated densities in the *max* step converge, the estimated transformation parameters in the *min* step will also converge to a local optimum.

7.6 Reduction of Formulation Under Pixel Independence

This chapter assumes that the neighboring pixels are correlated whereas in the earlier formulation of the minimax entropy framework the image pixels were assumed to be independently distributed. Therefore, it is interesting to see if the minimax entropy registration framework developed in this chapter reduces to the earlier formulation if the pixels are assumed to be independent. Mathematically, from equation (7.1), the image pixel labels will be independently distributed if the random variables $\mathbf{e}_i$ are set equal to 1 for all i, that is, $\mathbf{e}_i = 1, \forall i$.

Assuming that the pixel labels are independently distributed, it is easy to show that $\varepsilon^*_{vi} = h_{vi} = \ln\ P(\mathbf{m}_i = v|x_i, y_i)$. For this case the expected value $< z_{vi} >_{E(\mathbf{Z})}$ can be simplified to be:

$$
\begin{aligned}
< z_{vi} >_{E(\mathbf{Z})} &= \left(\frac{\exp\left(\beta \varepsilon^*_{vi}\right)}{\sum_{\alpha \in A} \exp\left(\beta \varepsilon^*_{\alpha i}\right)} \right) \\
&= \left(\frac{\exp\left(\beta h_{vi}\right)}{\sum_{\alpha \in A} \exp\left(\beta h_{\alpha i}\right)} \right) \\
&= \left(\frac{\exp\left(\beta\ \ln P(\mathbf{m}_i = v|x_i, y_i)\right)}{\sum_{\alpha \in A} \exp\left(\beta\ \ln P(\mathbf{m}_i = \alpha|x_i, y_i)\right)} \right) \\
&= \left(\frac{P(\mathbf{m}_i = v|x_i, y_i)^\beta}{\sum_{\alpha \in A} P(\mathbf{m}_i = \alpha|x_i, y_i)^\beta} \right) \\
&= \left(\frac{[p_v(x_i, y_i)\ P_i(v)]^\beta}{\sum_{\alpha \in A} [p_\alpha(x_i, y_i)\ P_i(\alpha)]^\beta} \right)
\end{aligned}
$$

which is the same expression as the *max step* developed in equation 5.3.

7.7 Summary

This chapter develops the minimax registration framework while incorporating the correlation among the neighboring pixel intensities. The correlation is modeled using a 1D Markov random process with only a 1st order neighborhood system. Though the correlation could be modeled as a 2D MRF and higher order neighborhood system, evaluating the registration parameters under such a model would be computationally very expensive. The probability distribution function of the segmentation of a portal image is then approximated by a distribution function which can be written in a factored form, using the principle of maximum entropy. The factored distribution function allows it to be manipulated analytically. The *min* step is also modified accordingly to incorporate the correlation among the pixel intensities. The next chapter evaluates and compares the performance of the proposed registration framework with the mutual information based registration algorithm using the algorithm developed in this chapter.

Chapter 8

Performance Analysis of the Algorithm Incorporating Pixel Correlation

8.1 Introduction

Chapter 7 developed the minimax entropy registration framework which models the correlation among the neighboring pixels in the image using a 1D Markov random field (MRF). In the *min* step, a maximum entropy probability distribution of the segmentation of the portal images is obtained. The probability of an edge is estimated as a *by–product* of this step. The *min* step, utilizing the segmentation information estimated in the *max* step, estimates the transformation parameters. This chapter evaluates the robustness and accuracy of this registration framework utilizing both simulated and real portal images.

The simulated portal images are obtained using the procedure as described in the

chapter 6. For real patient data, the 2D portal images, both in the *anterior–posterior* (AP) and *left–lateral* (LL) views are obtained using the treatment energy X–rays. The X–ray films are then digitized using a laser digitizer. The dynamic range of the pixel intensities in the digitized images is 0 to 255.

8.2 2D Simulator to 2D Portal Image Registration

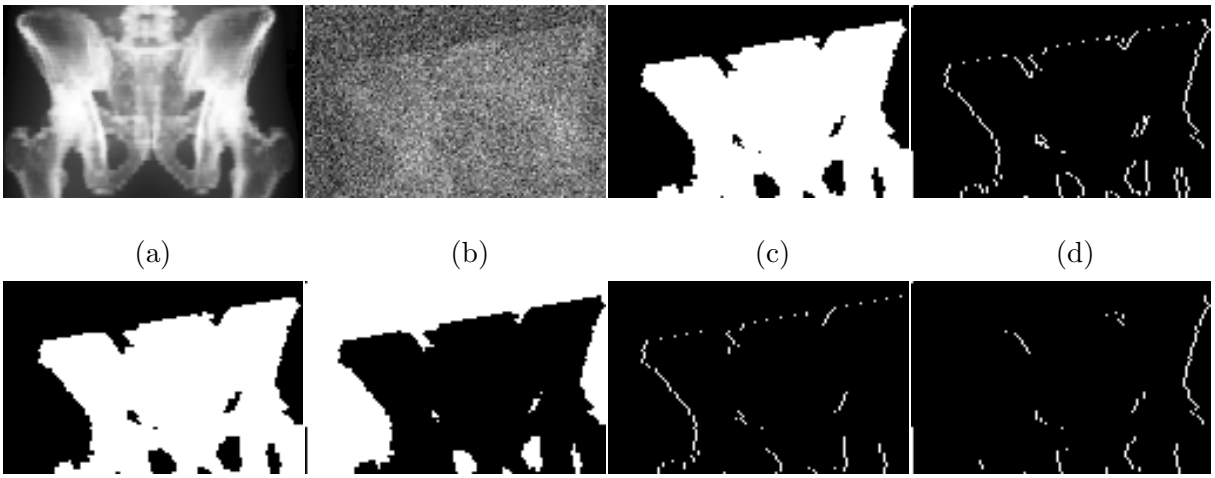
This section evaluates the performance of the minimax (MIME) entropy registration framework and compares it to the performance of the mutual information (MI)–based registration framework. Simulated portal and simulator images, with known rigid transformation parameters, are used in the evaluation of the performance of the two strategies. The simulated portal image is obtained by first rotating the simulator image by 10° and translating it by 15 pixels along both the X–axis and the Y–axis. Then varying amounts of i.i.d. Gaussian noise and blur are added.

To compare the performance of the MIME algorithm to the MI–based algorithm, we first analyze the performance of the two methods given the *true* segmentation of the portal image is known (as the portal images are simulated images, the true segmentation is known). Then we compare their performance when the portal image segmentations are not known. Finally, we present the performance of the algorithms with increasing noise and blur in the portal images.

8.2.1 Using True Segmentation of the Portal Images

Figure 8.1 (a) and (b) show the simulator image and the portal images, respectively. An i.id. Gaussian noise of standard deviation (σ) 30.0 was added to the portal image. The true segmentation of the portal image used is shown in the figure 8.1 (c). Figure 8.1

	t_x (pix)	t_y (pix)	θ_{XY} (deg)
True	15.0	15.0	10.0
MI	4.86	-0.744	3.37
MIME	15.02	14.35	10.14

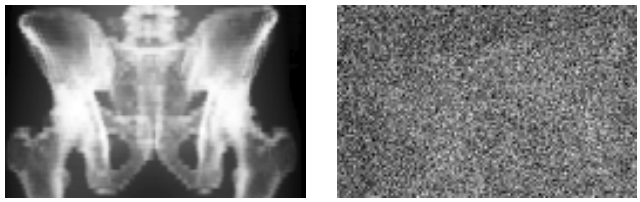


(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 8.1: (a) Simulated simulator image. (b) Simulated portal image. An i.i.d. Gaussian noise of standard deviation (σ) 30.0 added to the image. (c) True segmentation of (b). (d) True vertical edges. (e), (f), (g) and (h) are the actual joint probability distributions of bone–bone, nobone–nobone, bone–nobone and nobone–bone respectively. Note that joint probability distribution bone–bone means probability that ith pixel is a bone and $(i-1)$th pixels is also a bone. Joint probability distribution bone–nobone means probability that ith pixel is a bone and $(i-)1$th pixel is no–bone. Similarly for the other joint probability distributions.

	t_x (pix)	t_y (pix)	θ_{XY} (deg)
True	15.0	15.0	10.0
MI	0.277	-6.04	-1.40
MIME	15.21	14.21	10.23

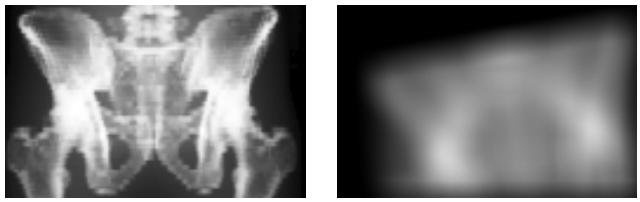


(a) (b)

Figure 8.2: (a) Simulated simulator image. (b) Simulated portal image with added i.i.d. Gaussian noise of standard deviation (σ) 35.0. The table above shows the true registration parameters in the first row, parameters as estimated by the MI–based registration algorithm in the second row and the third row shows the parameters estimated by the proposed MIME algorithm (incorporating the 1D MRF).

(d) shows the estimated edges of the portal image. Note that although the algorithm was formulated to detect only the vertical edges, most of the edges have been correctly detected and localized. Figures (8.1) (e), (f), (g) and (h) show the joint probability distributions of bone–bone, nobone–nobone, bone–nobone and nobone–bone, respectively. That is, the figure (8.1) (h) shows the joint probability that the ith pixel is no–bone and the $(i-1)$th pixel is a bone. The table in figure 8.1 shows that the MI–based method gets trapped in a local minimum far from the true parameters whereas the MIME–based algorithm (incorporating the 1D MRF model as developed in chapter 7) estimated the

	t_x (pix)	t_y (pix)	θ_{XY} (deg)
True	15.0	15.0	10.0
MI	16.36	16.97	9.93
MIME	15.13	14.54	10.02



(a) (b)

Figure 8.3: (a) Simulated simulator image. (b) Blurred simulated portal image. A blurring kernel of width 15 employed to blur the image. Again, the table above shows the true and the estimated registration parameters. The MIME algorithm incorporates the 1D MRF.

registration parameters with sub–pixel accuracy. Figure 8.2 (b) shows the portal image where i.i.d. Gaussian noise with $\sigma = 35.0$ is added to the portal image. The results in figure 8.2 also confirm that the MI–based algorithm gets stuck into a local minimum whereas the MIME approach correctly estimates the parameters with sub–pixel accuracy. Figure 8.3 (b) shows a portal image which has been blurred using a kernel of width 15 pixels. Figure 8.3 (a) is the simulator image. The table at the top of figure 8.3 shows that even for the blurred portal image, the MIME approach successfully estimates the registration parameters with sub–pixel accuracy. The MI–based algorithm is off by as much as 2 pixels in estimating translation along Y–axis.

Figures 8.4 and 8.5 show the surface plots of the energy functions of the MI and

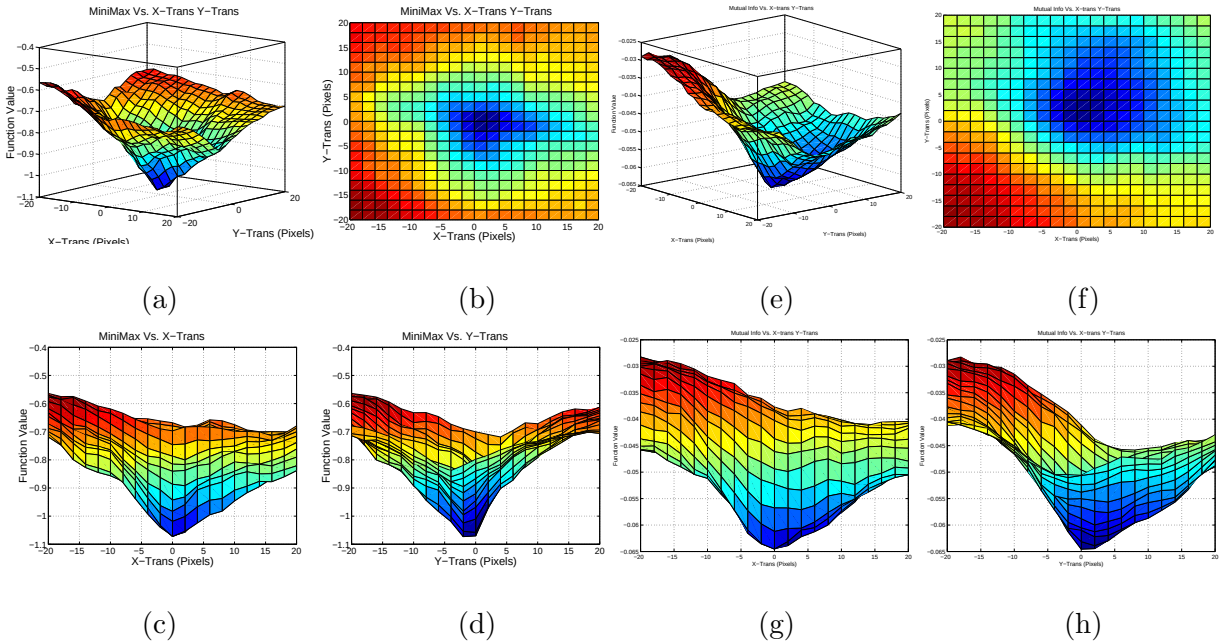


Figure 8.4: Energy plots for noisy ($\sigma = 20.0$ pixels) simulated portal images (figures 8.1 (a), (b)). (a), (b), (c) and (d) show the surface plots of the MIME energy function. (c), (d), (e) and (f) show the surface plots of the MI energy function. The global optimum position is (0,0). The MIME algorithm used the true segmentation of the portal image.

MIME algorithm for varying amounts of translations along the X–axis and Y–axis. The MIME algorithm utilized the true segmentations of the portal image shown in figures 8.1 (c), (d), (e), (f), (g), (h). The top view of the energy function plot for the MIME algorithm in figure 8.4 (b) shows that the global optimum point is located at the true optimum whereas, for the MI algorithm, figure 8.4 (f) shows that the global optimum is *shifted* from the true optimum. One reaches the same conclusion from the surface plots, shown in figures 8.5 (b) and (f), for the blurred portal images.

The results of this section show that, given the true segmentation of the portal

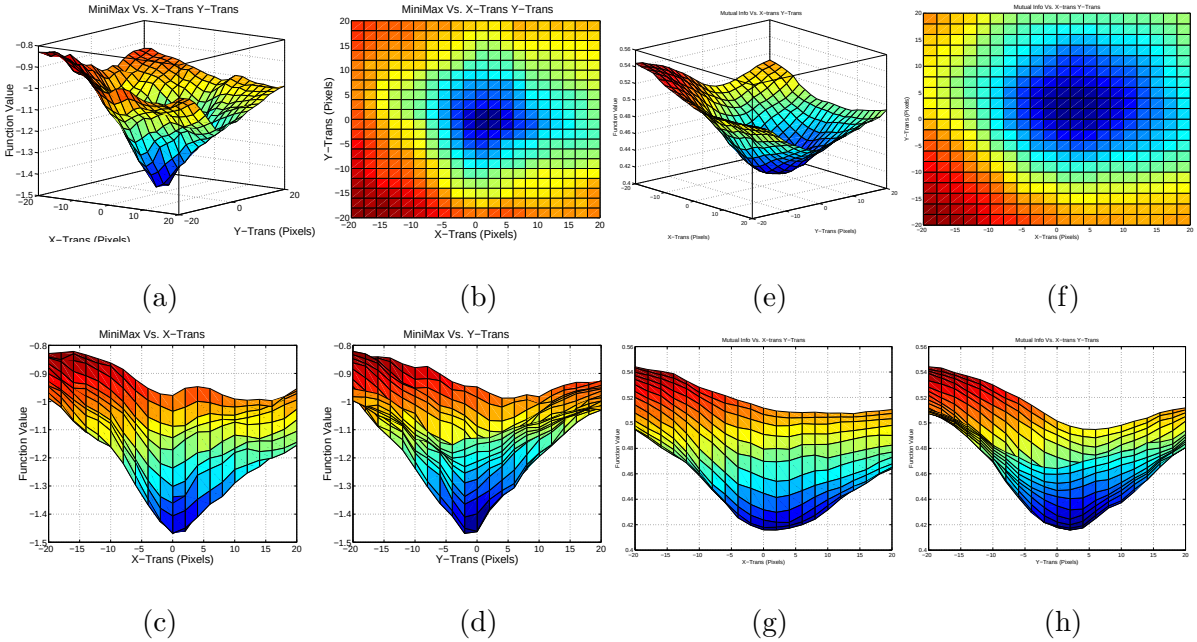
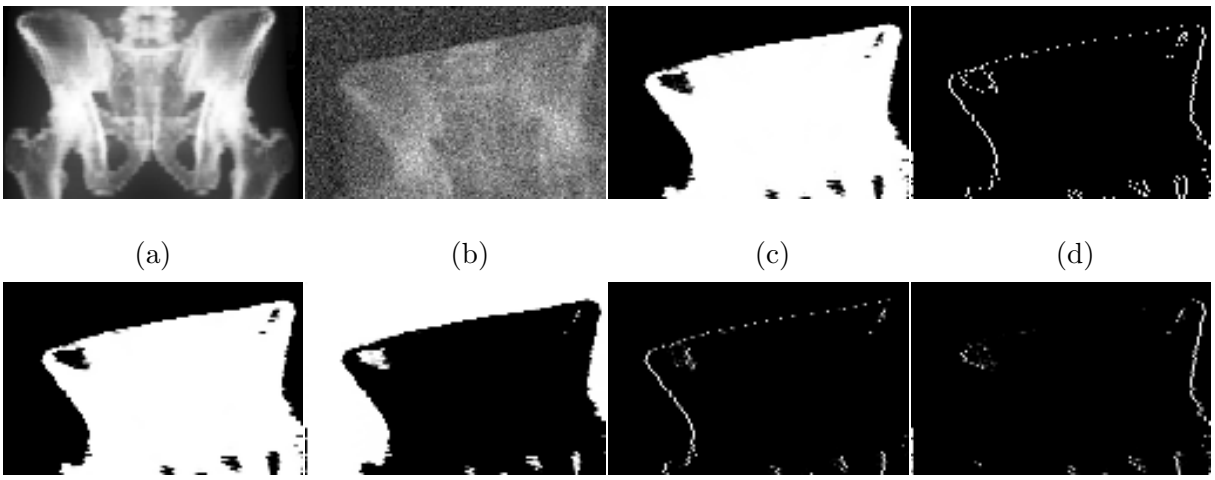


Figure 8.5: Energy plots for blurred (11 pixels) simulated portal images (figures 8.3 (a) and (b)). (a), (b), (c) and (d) show the surface plots of the MIME energy function. (c), (d), (e) and (f) show the surface plots of the MI energy function. The global optimum position is (0,0). The MIME algorithm used the true segmentation of the portal image.

image, the MIME–based method is robust to both noise and blur and can estimate the parameters with sub–pixel accuracy. Thus, incorporation of the segmentation information into the registration framework can lead to robust and accurate estimation of the transformation parameters. However, the true segmentation of the portal image is almost never available. Thus, the MIME registration framework simultaneously and iteratively estimate the segmentation and registration of the portal image to the 2D simulator or 3DCT images.

	t_x (pix)	t_y (pix)	θ_{XY} (deg)
True	15.0	15.0	10.0
MI	15.31	15.28	9.94
MIME	15.28	15.02	9.91

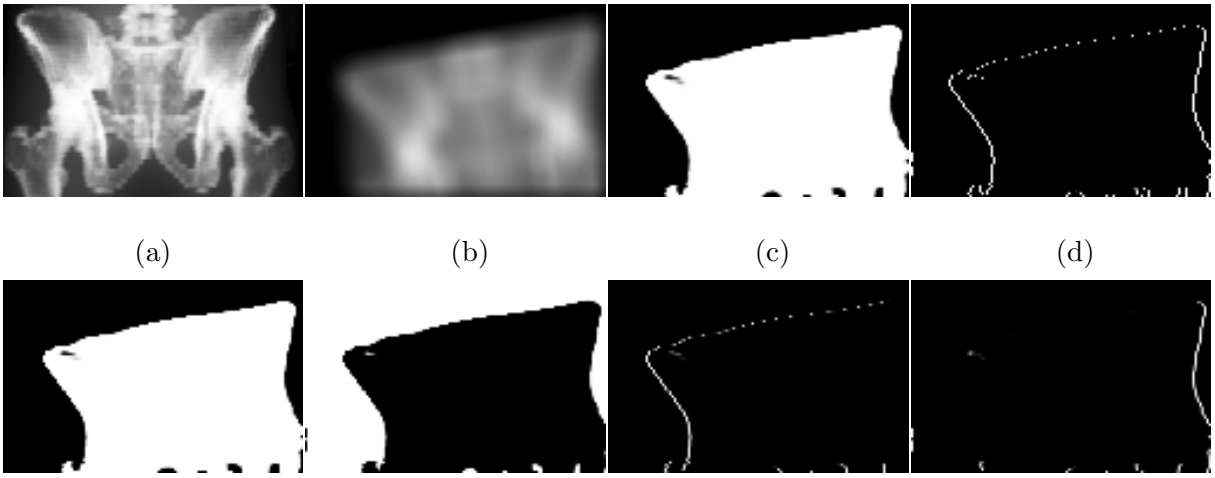


(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 8.6: (a) Simulated simulator image. (b) Simulated portal image with i.i.d. Gaussian noise of σ 20.0 added. (c) Estimated segmentation of portal image in (b). (d) Estimated edges. (e), (f), (g) and (h) show the estimated joint probability distribution of bone–bone, nobone–nobone, bone–nobone and nobone–bone, respectively.

	t_x (pix)	t_y (pix)	θ_{XY} (deg)
True	15.0	15.0	10.0
MI	16.53	16.308	10.26
MIME	15.47	15.72	10.03



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 8.7: (a) Simulated simulator image. (b) Simulated portal image blurred using a blurring kernel of width 11 pixels. (c) Estimated segmentation of portal image in (b). (d) Estimated edges. (e), (f), (g) and (h) show the estimated joint probability distribution of bone–bone, nobone–nobone, bone–nobone and nobone–bone, respectively.

8.2.2 Using Estimated Segmentation of the Portal Images

In the previous subsection, having established that incorporation of the segmentation information into the registration leads to a robust and accurate registration framework, this subsection compares the performance of the MIME and MI–based algorithms where the segmentation of the portal image is being estimated automatically and iteratively within the MIME registration framework.

As a first example, figures 8.6 (a), (b) show the simulator and the portal images, respectively, registered using both the MI and MIME methods. The portal image is obtained by transforming the simulator image by known transformation parameters and then adding an i.i.d. Gaussian noise of standard deviation 20.0. The table at the top of the figure 8.6 shows the true and the estimated parameters as estimated by the two algorithms. This table shows that in the presence of noise the performance of the MIME and MI algorithms is comparable for this dataset, with both algorithms estimating the registration parameters with sub–pixel accuracy. Figure 8.6 (c) shows the segmentation of the portal image as estimated by the MIME algorithm and the estimated edges are shown in figure 8.6 (d).

The surface plots for the energy functions of the MI and MIME methods for this data set is shown in the figure 8.8. The top views of the energy functions, in figure 8.8 (f), show that the the global optimum for the MI energy function is shifted from the true optimum point, which is location (0,0).

As a second example, consider the simulator and the portal images shown in figures 8.7 (a) and (b) respectively. The portal image is obtained by transforming the simulator image and then blurring it by 11 pixels. The table at the top of the figure shows the true parameters and the parameters estimated by the MI and MIME algorithms. Note

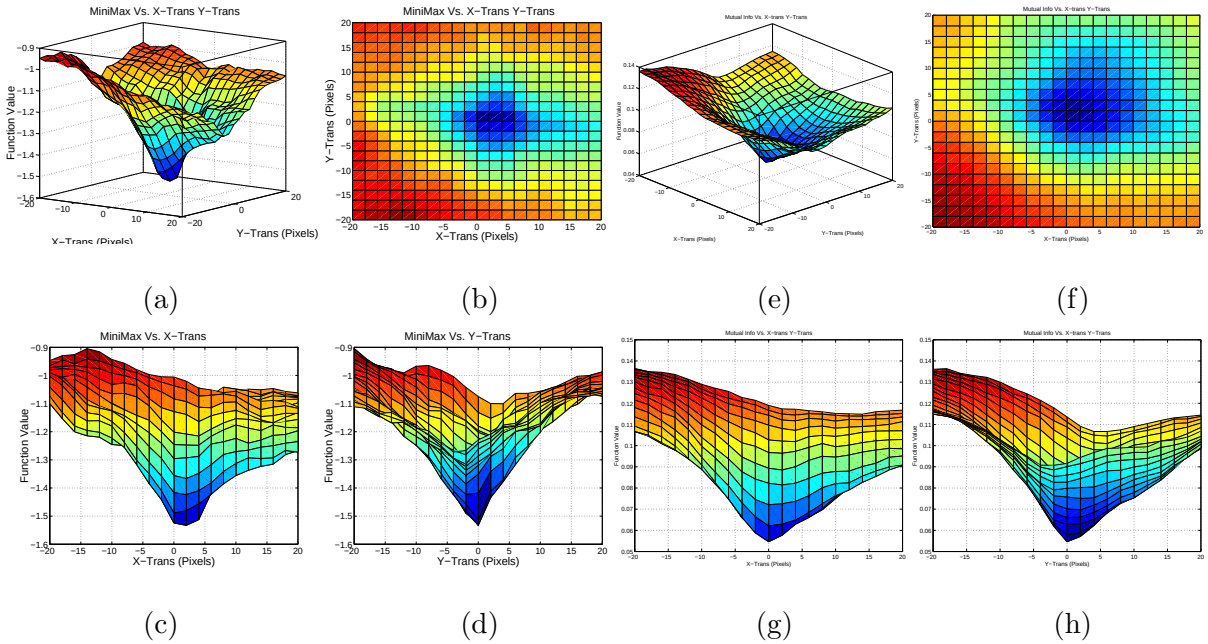
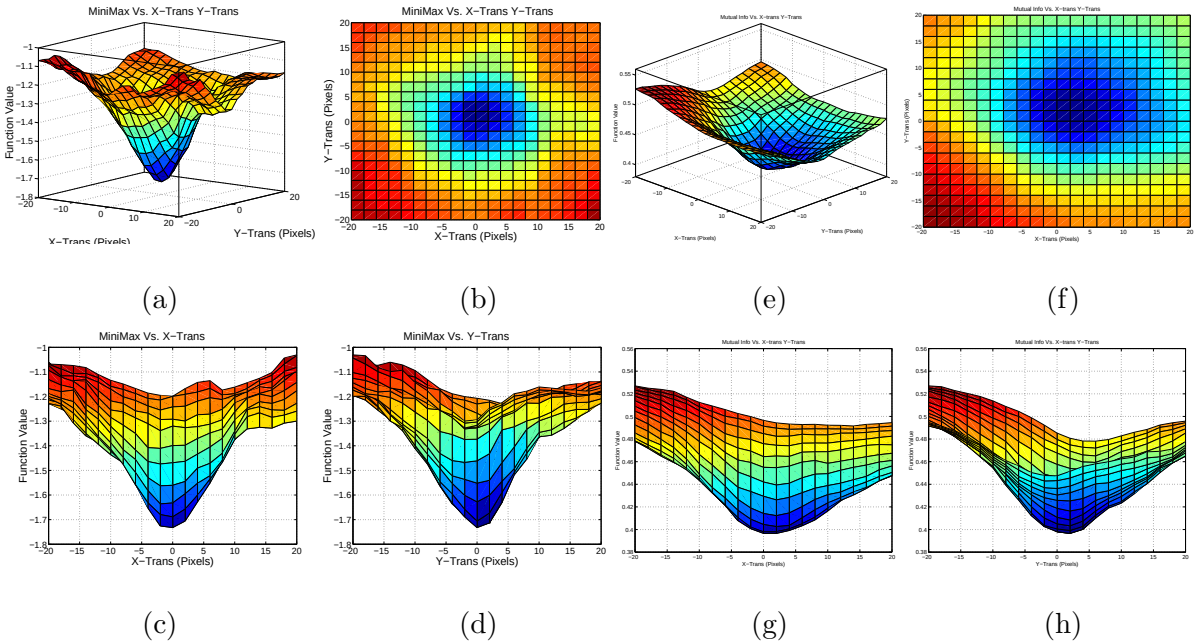


Figure 8.8: Energy function for noisy ($\sigma = 20$ pixels) simulated portal images (figure 8.6 (a) and (b)). (a), (b), (c), (d) show the surface energy plots for the MIME based algorithm. (e), (f), (g), (h) is the surface plots of the energy function of the MI–based registration algorithm. The MIME algorithm, in addition used the various joint probability distributions shown in figure 8.6. The optimum registration is at position (0,0).

that the estimated translation along the X–axis by the MI algorithm is off by more than 1.5 pixels. However, the MIME algorithm for this dataset estimates the parameters with sub–pixel accuracy. The segmentation of the portal image, as estimated by the MIME algorithm, is shown in figure 8.7 (c) and figure 8.7 (d) shows the estimated edges.

The surface plots of the energy functions of the MI and MIME–based algorithms are shown in figure 8.9. It is clear from the surface plots, figures 8.9 (b) and (f), that while the global optimum for the MIME energy function is located at the true optimum,



(a) (b) (e) (f)

(c) (d) (g) (h)

Figure 8.9: Energy function for blurred simulated portal images (figure 8.7 (a) and (b)). (a), (b), (c), (d) show the surface energy plots for the MIME based algorithm. (e), (f), (g), (h) is the surface plots of the energy function of the MI–based registration algorithm. The MIME algorithm, in addition used the various joint probability distributions shown in figure 8.7. The optimum registration is at position (0,0).

the optimum point of the MI energy function is again shifted by a few pixels from the true optimum point. The MIME algorithm, in addition, uses the segmentation of the portal image shown in figure 8.9 (c).

The results of this subsection show that the MIME algorithm is more accurate in estimating the pixels, in the presence of blur, as compared to the MI–based registration algorithm. However, in the presence of noise, the performance of the two algorithms is comparable, with both algorithms leading to sub–pixel accuracy in the estimated regis-

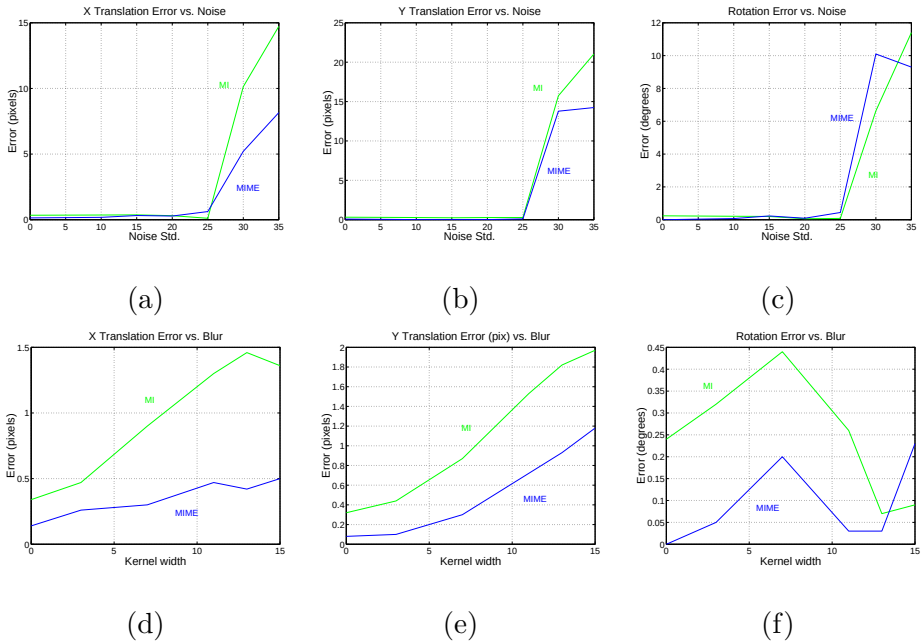


Figure 8.10: Performance of the MIME (purple) and MI (green) based registration algorithms under increasing noise and blur in the portal images. (a), (b) and (c) show the graphs of the performance of the two algorithms under increasing noise. (d), (e), (f) show the performance under increasing blur.

tration parameters.

8.2.3 Performance Under Increasing Noise and Blur

In this subsection we will compare the performance of the MIME and the MI based algorithms under increasing noise and blur in the portal images. Figure 8.10 (a), (b), and (c) show the error in estimated registration parameters as a function of increasing noise. From these figures, it is clear that the performance of the two algorithms is comparable with both algorithms estimating the parameters with sub–pixel accuracy. However, the

two algorithms can correctly estimate the transformation parameters for noise of standard deviations up to 25.0 only.

Figure 8.10 (d), (e), and (f) shows the performance of the two algorithms under increasing blur in the portal images. These figures show that the MIME algorithm leads to more accurate estimates of the transformation parameters in the presence of blur as compared to the MI–based algorithm. This result is also expected as the MIME algorithm explicitly models the correlation between the pixels and blurring the images will only increase the correlation.

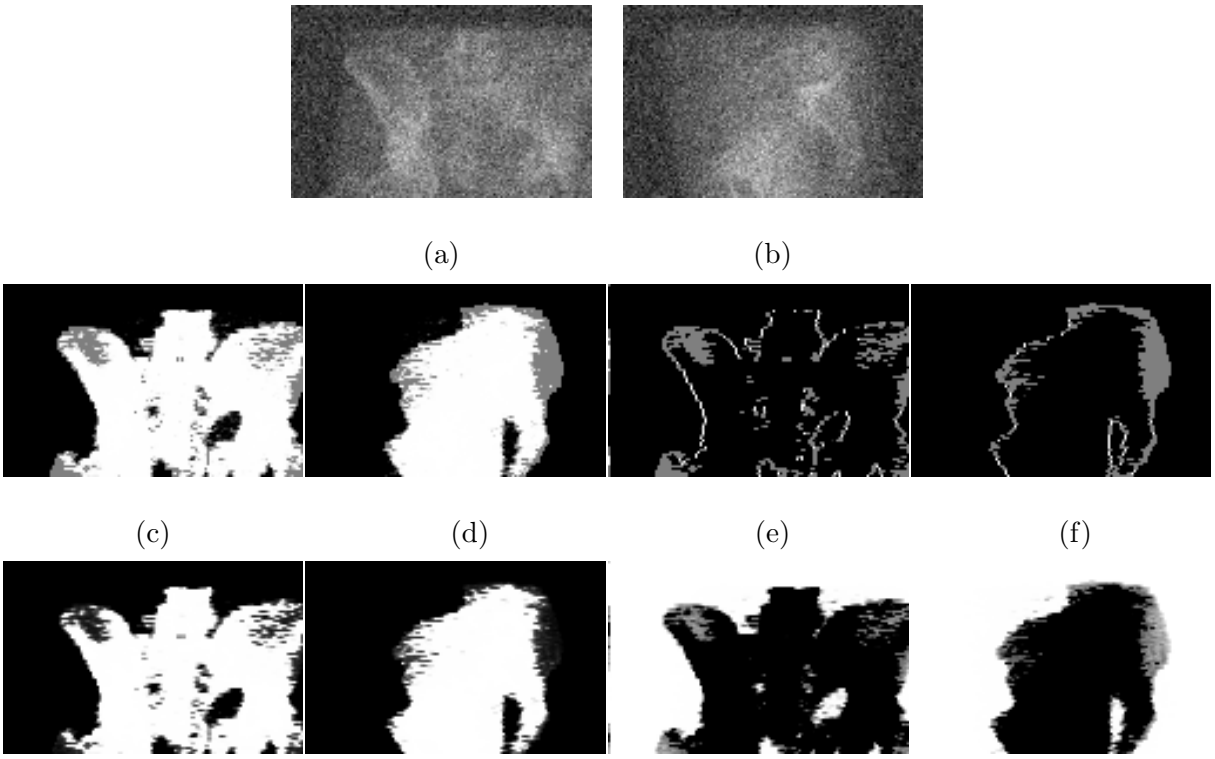
8.3 2D Portal to 3DCT Registration

This section compares the performance of the MI and MIME–based algorithms using dual simulated portal images. To obtain the simulated dual portal images, the diagnostic energy 3DCT voxel values are first mapped to the values at treatment energy. Digitally reconstructed radiographs (DRRs) are then rendered, after transforming the 3DCT data set by known transformation parameters, in the *anterior–posterior* (AP) and *left–lateral* (LL) directions. Varying amounts of noise and blur are then added to the DRRs to obtain the simulated portal images. Since the true transformation parameters are known, this dataset can be used to compare the performance of the two registration algorithms.

8.3.1 3D Analysis Using Dual Simulated Portal Images

The simulated AP and LL portal images, shown in figures 8.11 (a) and (b), are obtained by rendering the 3DCT dataset in the AP and LL directions (first transformed by $t_x = 15$ voxels, $t_y = 15$ voxels, $\theta_{XZ} = 10^o$) and then adding i.i.d. Gaussian noise of $\sigma = 20.0$ pixels. Note that the rotation angle θ_{XZ} is an out–of–plane angle for both the portal images.

	t_x (vox)	t_y (vox)	t_z (vox)	θ_{YZ} (deg)	θ_{XZ} (deg)	θ_{XY} (deg)
True	15.0	15.0	0	0	10.0	0
MI	15.36	15.49	0.66	0.58	8.01	-0.13
MIME	15.18	15.03	0.32	-0.133	10.88	0.41



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Figure 8.11: (a), (b) simulated AP and LL portal images, respectively. The simulated portal images are obtained by adding i.i.d. Gaussian noise of standard deviation (σ) 20.0 pixels. The dynamic range of pixel intensities of the portal images is 255. (c), (d) show the estimated segmentations. White is the estimated bone, black si the estimated no–bone and the gray are the pixels misclassified by the algorithm. (e), (f) are the estimated edges. (g), (h) are the estimated bone–bone probability distributions. (i), (j) are the estimated nobone–nobone distributions. The table at the top of the figure shows the true and the parameters estimated by the MI and MIME algorithms.

Figures 8.11 (c) and (d) show the estimated segmentations of the portal images. The estimated edges are shown in figures 8.11 (e) and (f). The table at the top of the figure shows that the MIME algorithm (incorporating 1D MRF) leads to more accurate estimate of the transformation parameters for this dataset as compared to the MI algorithm.

The portal images in figures 8.12 (a) and (b) are obtained by blurring the DRRs, rendered in the AP and LL directions, by applying a blurring kernel of width 11 pixels. The 3DCT data set was transformed by $t_x = 15$ voxels, $t_y = 15$ voxels, $\theta_{XZ} = 10^o$ to obtain the DRRs. The table at the top of the figure shows that for this dataset, the MIME algorithm is able to estimate the out–of–plane angle, θ_{XZ}, within 1° and the X, Y translations with sub–pixel accuracy.

To obtain the portal images in figures 8.13 (a), (b) and figures 8.14 (a), (b), the 3DCT dataset was first transformed by $t_x = 15$ voxels, $t_y = 15$ voxels, $\theta_{XY} = 10^o$. Then the DRRs, in the AP and LL directions, were rendered. For simulated portal images in figures 8.13 (a), (b), i.i.d. Gaussian noise of $\sigma = 20.0$ was added to the DRRs and to obtain the simulated portal images in figures 8.14 (a), (b), the DRRs were blurred by 11 pixels.

The table at the top of the figure 8.13 shows the parameters estimated by the MI and MIME algorithms for this dataset. Note again that the MIME algorithm estimates the transformation parameters with sub–pixel accuracy for this dataset. For the blurred simulated portal images also, see table at the top of figure 8.14, the MIME algorithm estimates the X, Y translation parameters with sub–pixel accuracy and the rotation angle θ_{XY} within 0.5^o of the true rotation.

These examples show that the MIME algorithm is robust to both noise and blur, for this dataset, and is more accurate in estimating the transformation parameters as

	t_x (vox)	t_y (vox)	t_z (vox)	θ_{YZ} (deg)	θ_{XZ} (deg)	θ_{XY} (deg)
True	15.0	15.0	0	0	10.0	0
MI	14.95	15.45	0.739	0.528	7.187	0.018
MIME	15.27	14.93	0.118	0.55	9.18	0.408

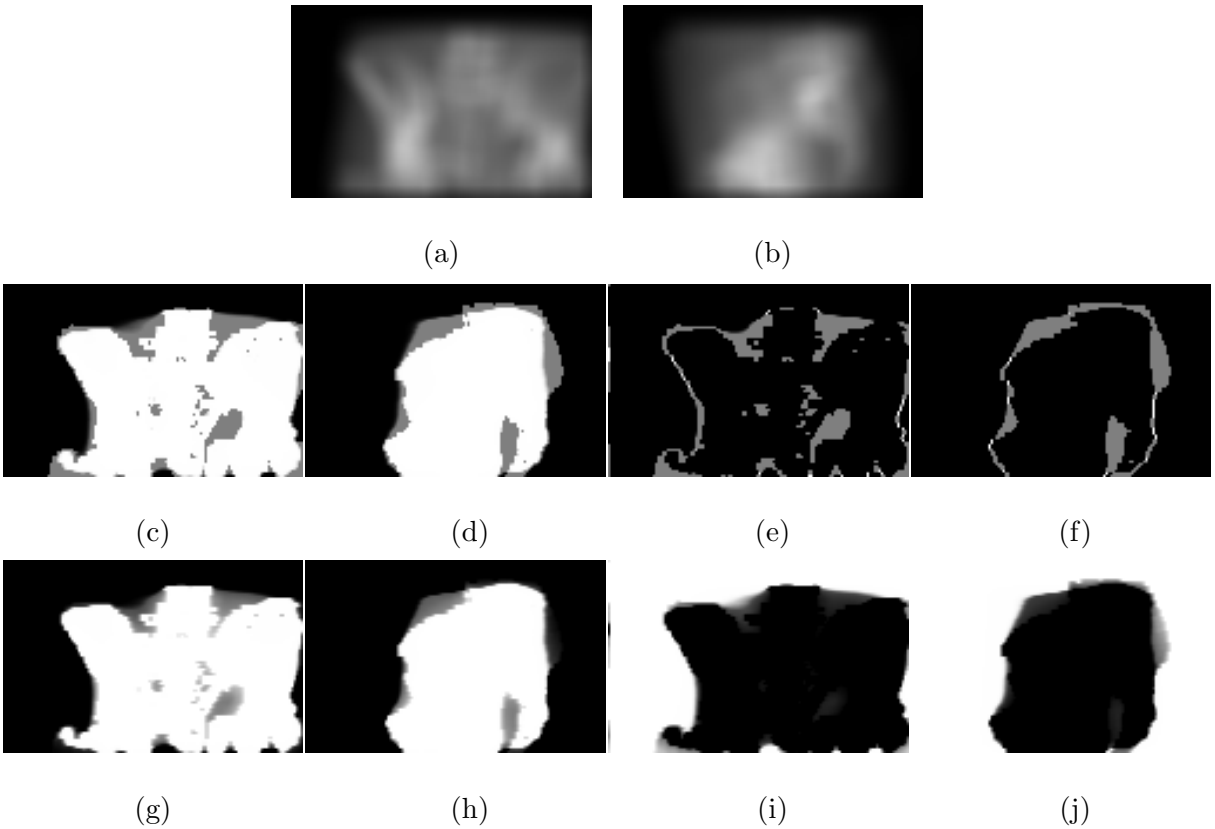


Figure 8.12: (a), (b) simulated AP and LL portal images, respectively. The simulated portal images are blurred by 11 pixels. (c), (d) show the estimated segmentations. (e), (f) are the estimated edges. (g), (h) are the estimated bone–bone probability distributions. (i), (j) are the estimated nobone–nobone distributions. The table at the top of the figure shows the true and the parameters estimated by the MI and MIME algorithms.

	t_x (vox)	t_y (vox)	t_z (vox)	θ_{YZ} (deg)	θ_{XZ} (deg)	θ_{XY} (deg)
True	15.0	15.0	0	0	0	10.0
MI	15.28	15.436	0.68	0.59	-0.05	6.23
MIME	15.20	14.95	0.026	-0.76	-0.03	10.39

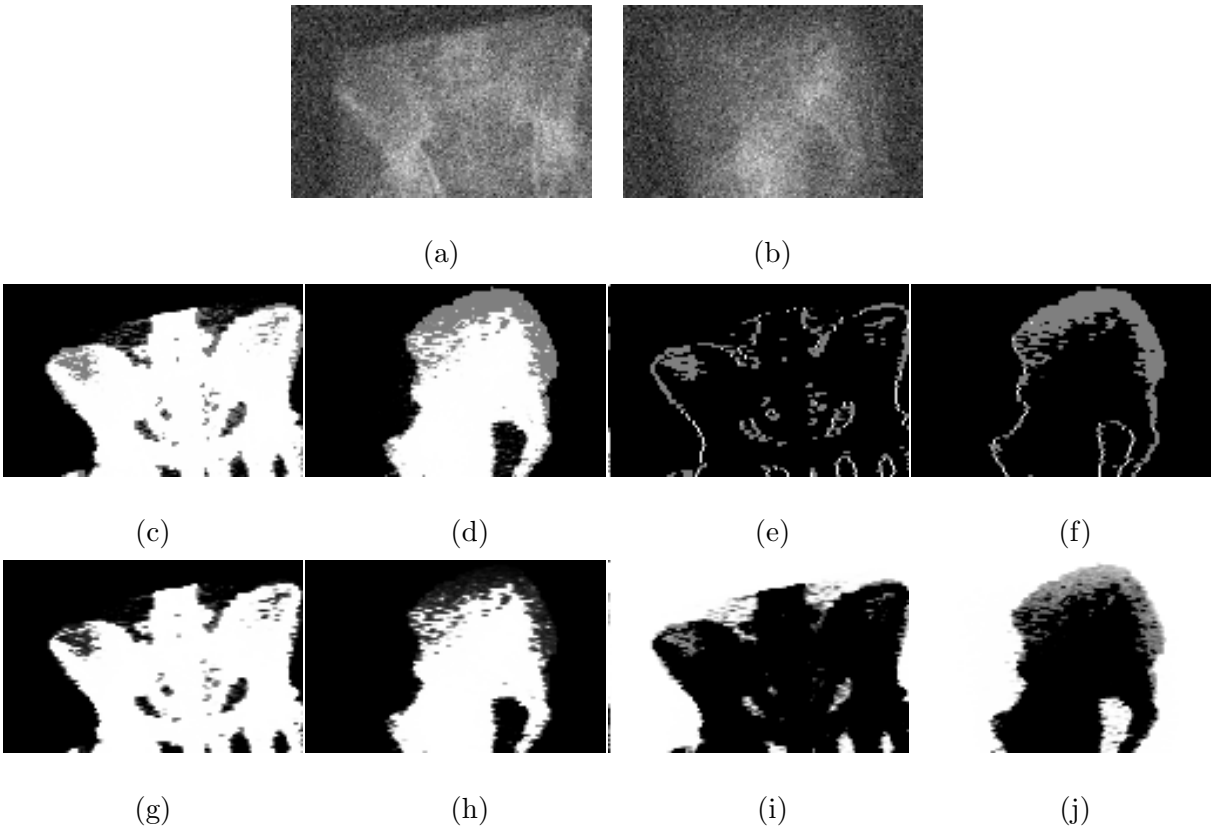


Figure 8.13: (a), (b) simulated AP and LL portal images, respectively. The simulated portal images are obtained by adding i.i.d. **Gaussian noise of standard deviation (σ) 20.0**. The dynamic range of pixel intensities of the portal images is 255. (c), (d) show the estimated segmentations. (e), (f) are the estimated edges. (g), (h) are the estimated bone–bone probability distributions. (i), (j) are the estimated nobone–nobone distributions. The table at the top of the figure shows the true and the parameters estimated by the MI and MIME algorithms.

	t_x (vox)	t_y (vox)	t_z (vox)	θ_{YZ} (deg)	θ_{XZ} (deg)	θ_{XY} (deg)
True	15.0	15.0	0	0	0	10.0
MI	15.40	15.80	0.72	1.67	-0.075	9.107
MIME	15.27	15.07	0.078	0.65	1.09	10.43

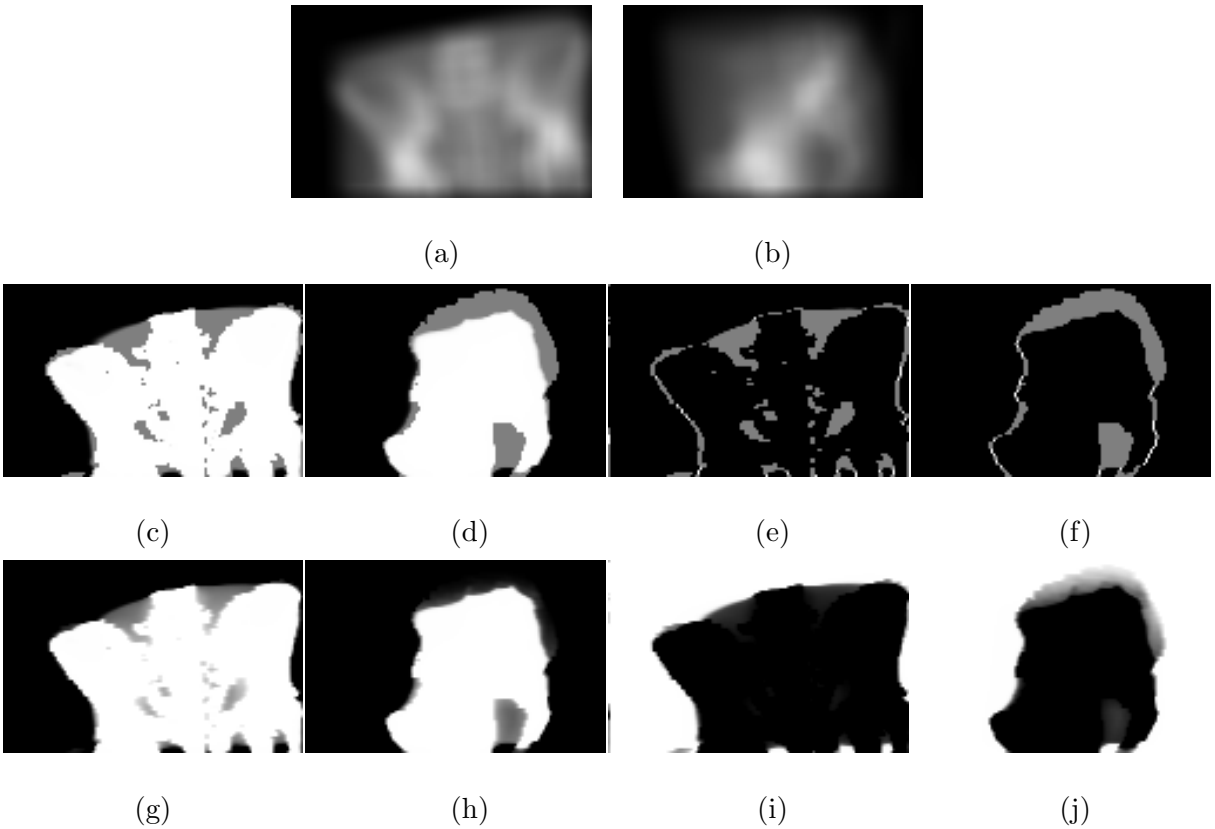
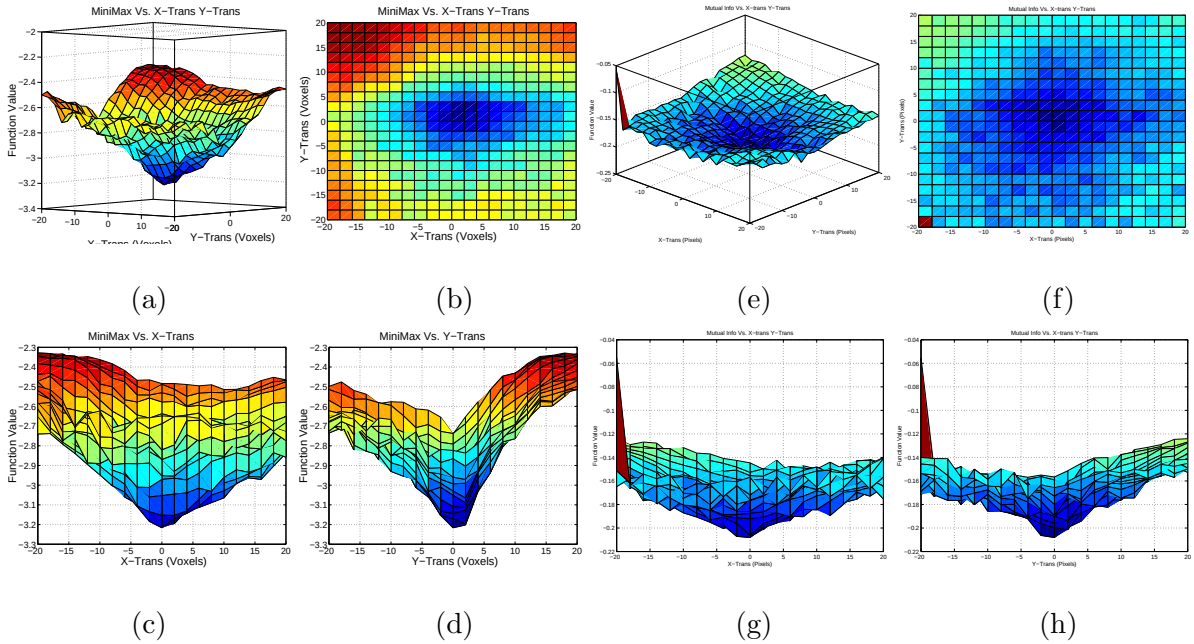


Figure 8.14: (a), (b) simulated AP and LL portal images, respectively. The simulated portal images are **blurred by 11 pixels**. (c), (d) show the estimated segmentations. (e), (f) are the estimated edges. (g), (h) are the estimated bone–bone probability distributions. (i), (j) are the estimated nobone–nobone distributions. The table at the top of the figure shows the true and the parameters estimated by the MI and MIME algorithms.



(a) (b) (e) (f)

(c) (d) (g) (h)

Figure 8.15: Energy function plots using noisy ($\sigma = 20.0$) simulated portal images (figures 8.11 (a), (b)). The true registration parameters are $t_x = 15$ voxels, $t_y = 15$ voxels and $\theta_{XZ} = 10^o$. (a), (b), (c) and (d) show the surface plots of the MIME energy functions. (e), (f), (g) and (h) are the surface plots of the MI energy functions. The MIME algorithm, in addition, uses the various distributions shown in figures 8.11.

compared to the MI–based registration algorithm. To further compare the MI and MIME–based registration algorithms, we plotted the energy functions of the two algorithms for these simulated portal image datasets.

Figures 8.15, 8.16, 8.17 and 8.18 show the surface plots of the energy function of the MIME and MI–based registration algorithms. In these plots, (0,0) is the optimum registration position. Simulated portal images shown in figures 8.11 8.12, 8.13 and 8.14 are used to plot the energy functions shown in the figures 8.15, 8.16, 8.17 and 8.18,

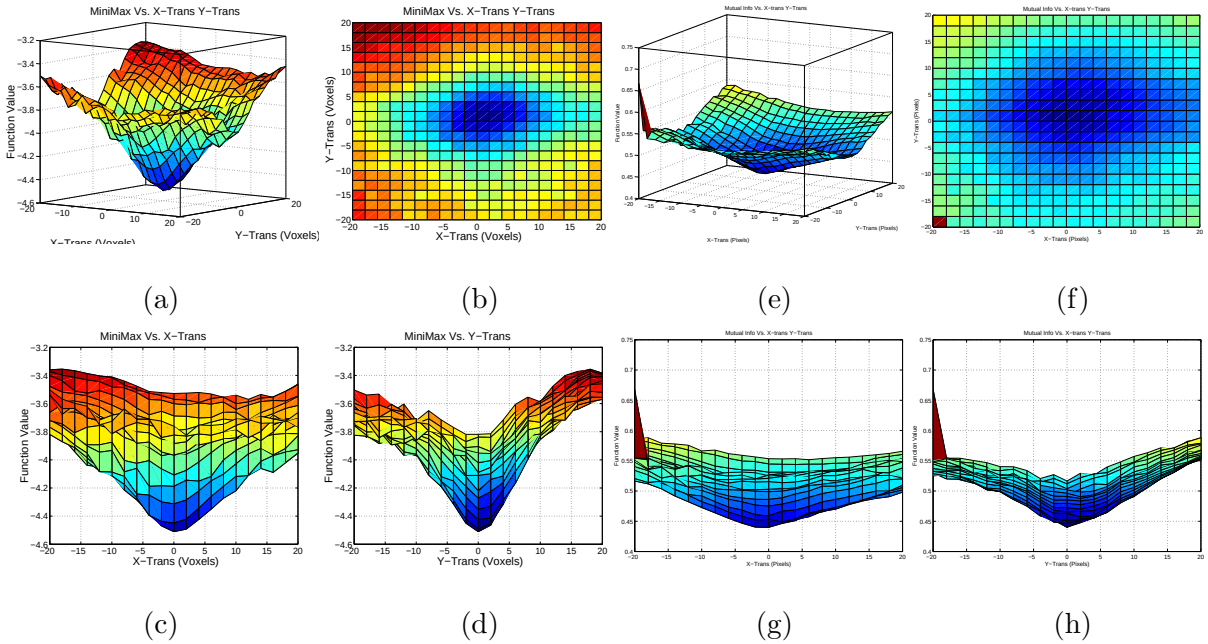


Figure 8.16: Energy function plots using blurred (11 pixels) simulated portal images (figures 8.12 (a), (b)). The true registration parameters are $t_x = 15$ voxels, $t_y = 15$ voxels and $\theta_{XZ} = 10^o$. (a), (b), (c) and (d) show the surface plots of the MIME energy functions. (e), (f), (g) and (h) are the surface plots of the MI energy functions. The MIME algorithm, in addition, uses the various distributions shown in figures 8.12.

respectively. The MIME registration framework in–addition utilizes the various estimated distribution shown in the figures. From these energy function surface plots it is clear that the energy function of the MI–based registration algorithm has many local optima for noisy simulated portal images and that while using the blurred simulated portal images, the MI–based energy function is flat near the global optimum. On the other hand, the energy functions for the MIME algorithm has a global optimum at (or at most one voxel away) from the global optimum for both the blurred and noisy simulated portal images.

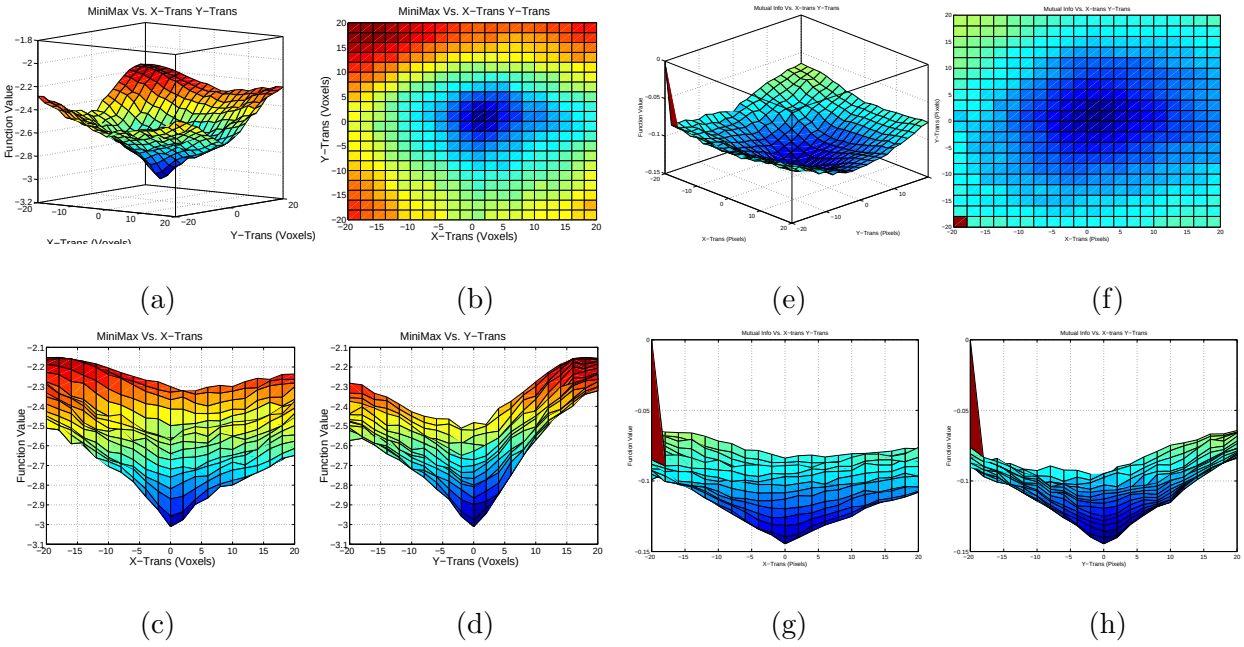


Figure 8.17: Energy function plots using noisy ($\sigma = 20.0$) simulated portal images (figures 8.13 (a), (b)). The true registration parameters are $t_x = 15$ voxels, $t_y = 15$ voxels and $\theta_{XY} = 10^o$. (a), (b), (c) and (d) show the surface plots of the MIME energy functions. (e), (f), (g) and (h) are the surface plots of the MI energy functions. The MIME algorithm, in addition, uses the various distributions shown in figures 8.13.

The surface plots of the energy functions also show that the MIME algorithm is both robust to noise and leads to accurate estimates of the transformation parameters as compared to the MI–based registration algorithm. To further compare the performance of the MI and MIME algorithms, we plotted the error in the estimated parameters for both MI and MIME algorithms, for increasing levels of noise and blur in the simulated portal images.

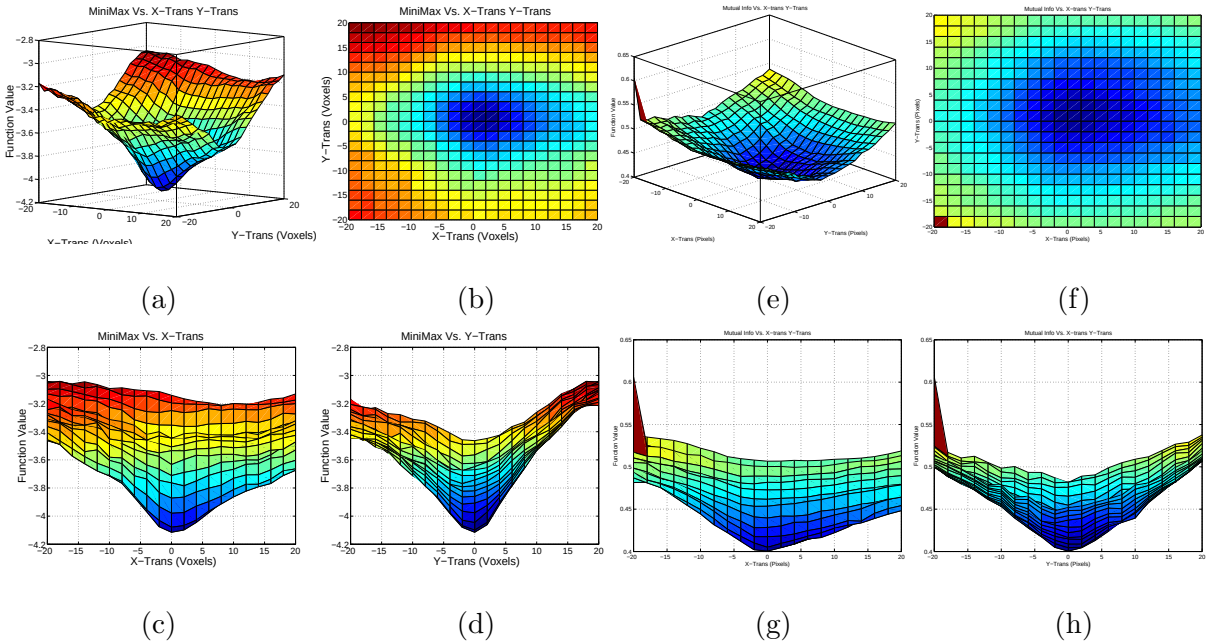


Figure 8.18: Energy function plots using blurred (11 pixels) simulated portal images (figures 8.14 (a), (b)). The true registration parameters are $t_x = 15$ voxels, $t_y = 15$ voxels and $\theta_{XY} = 10^o$. (a), (b), (c) and (d) show the surface plots of the MIME energy functions. (e), (f), (g) and (h) are the surface plots of the MI energy functions. The MIME algorithm, in addition, uses the various distributions shown in figures 8.14.

8.3.2 Performance Under Varying Noise and Blur

In this section, we compare the performance of the MIME and MI–based algorithms for increasing noise and blur in the simulated portal images.

Figures 8.19 (a), (b), (c) show the graphs of the error in the estimated parameters with increasing noise in the portal images. Figures 8.19 (d), (e), (f) show the graphs of error with increasing blur. These graphs show that the MIME algorithm is robust to both noise and blur in the portal image and leads to more accurate estimation of the

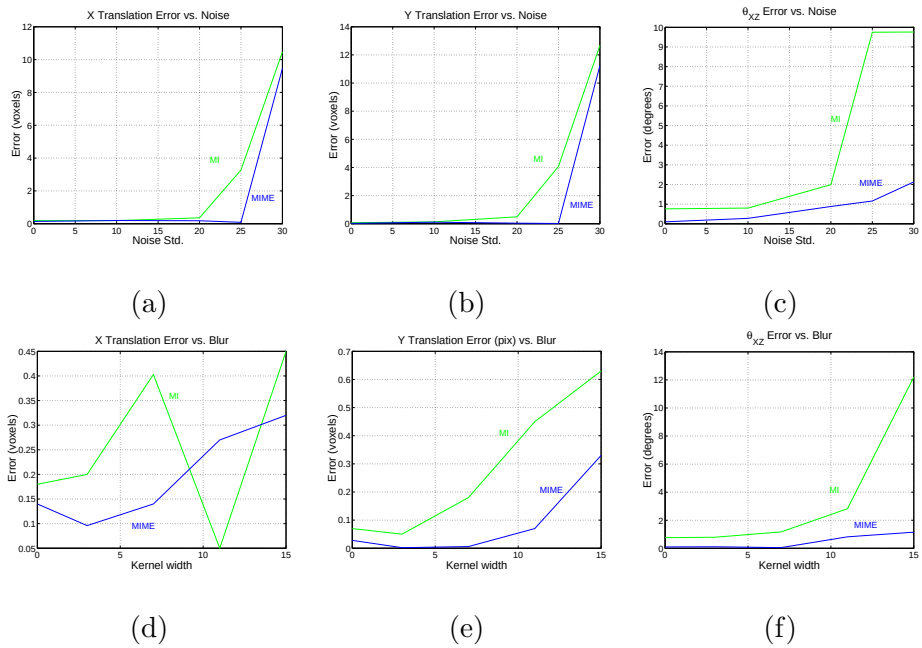


Figure 8.19: (a), (b), (c) Graphs of error in estimated transformation parameters with increasing noise. (d), (e), (f) show the graphs of errors in estimated parameters with increasing blur in the simulated portal images. The graph for MI is shown in green and the graph for MIME is shown in purple. The true registration parameters are $t_x = 15$ voxels, $t_y = 15$ voxels, $t_z = 0$ voxels, $\theta_{XY} = 0.0^o$, $\theta_{XZ} = 10^o$ and $\theta_{YZ} = 0^o$.

transformation parameters. Note also that the rotation angle, θ_{XZ}, being estimated is an out–of–plane angle for both the AP and LL portal images. However, using dual portal images, the MIME algorithm is able to estimate this angle within 1° for noise up to a standard deviation of 25.0. Figure 8.19 (d) shows that for blur 11 in the portal image, the error in the estimated X–translation reduces suddenly for the MI–based algorithm. This can only be explained by the presence of a local optimum close to the true optimum for this dataset.

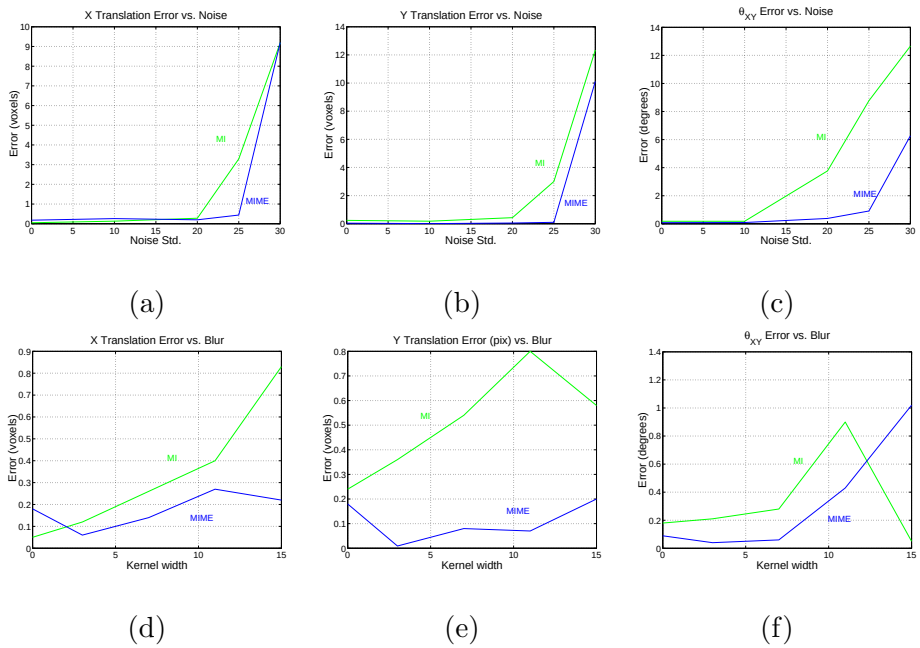


Figure 8.20: (a), (b), (c) Graphs of error in estimated transformation parameters with increasing noise. (d), (e), (f) show the graphs of errors in estimated parameters with increasing blur in the simulated portal images. The graph for MI is plotted in green and the graph for MIME is plotted in purple. The true registration parameters are $t_x = 15$ voxels, $t_y = 15$ voxels, $t_z = 0$ voxels, $\theta_{XY} = 10.0^o$, $\theta_{XZ} = 0^o$ and $\theta_{YZ} = 0^o$.

Figure 8.20 plots the error in the estimated transformation parameters where the true parameters to be estimated are $t_x = 15$ voxels, $t_y = 15$ voxels, $t_z = 0$ voxels, $\theta_{XY} = 10.0^o$, $\theta_{XZ} = 0^o$ and $\theta_{YZ} = 0^o$. Note that the angle θ_{XY} is an in–plane rotation angle for the AP portal image. These graphs also show that, for this dataset, the MIME algorithm estimates the transformation parameters robustly and accurately under both increasing noise and blur. However, figure 8.20 (f) shows that at a blur of 15 pixels, the error in the estimated θ_{XY} is very small. This again can only be explained by presence of a local

optimum close to the true optimum for this dataset.

8.4 Performance on Real Patient Data Set

This section presents the performance of the MIME registration framework on real patient data. The AP and LL portal images were obtained by moving the patient into the treatment room and actually imaging the 2D portal images, using the treatment energy X–rays, onto the X–ray films. The X–ray films were then digitized using the 12–bit CCD digitizer. However, only 8–bit per pixel data were available for estimating the registration parameters. For the purposes of registration, the digitized portal images were then contrast enhanced using adaptive histogram equalization technique. The treatment planning 3DCT dataset of the patient was obtained earlier. After the dual portal images were contrast enhanced, registration of the dual AP and LL portal images to 3DCT dataset was carried out completely automatically, with no human interaction.

Figures 8.21 (a), (b) show the contrast enhanced AP and LL portal images, respectively. The DRRs obtained from the 3DCT dataset in the AP and LL directions, in its initial pose, are shown in figures 8.21 (c) and (d). The DRRs rendered from the 3DCT dataset, in its final pose estimated by the MIME algorithm, in the AP and LL directions are shown in the figures 8.21 (e), (f). The segmentations of the portal images, iteratively and simultaneously estimated in the MIME framework are shown in the figures 8.21 (g), (h). Since the true transformation parameters are not known, to assess the accuracy of the estimated pose by the proposed framework, contours are hand drawn on the portal images along the prominent anatomical features. These contours are shown in red in figures 8.21 (a), (b). The contours are then mapped onto the DRRs obtained from the 3DCT image in its initial and final pose undeformed, as shown in the figures 8.21 (c), (d),

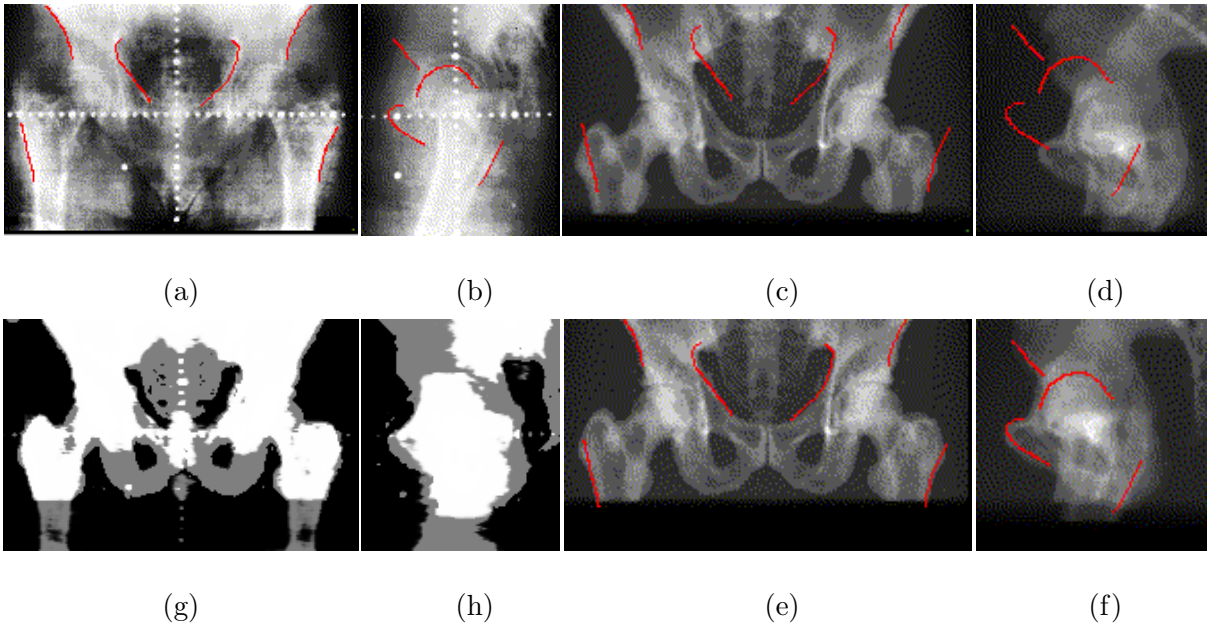
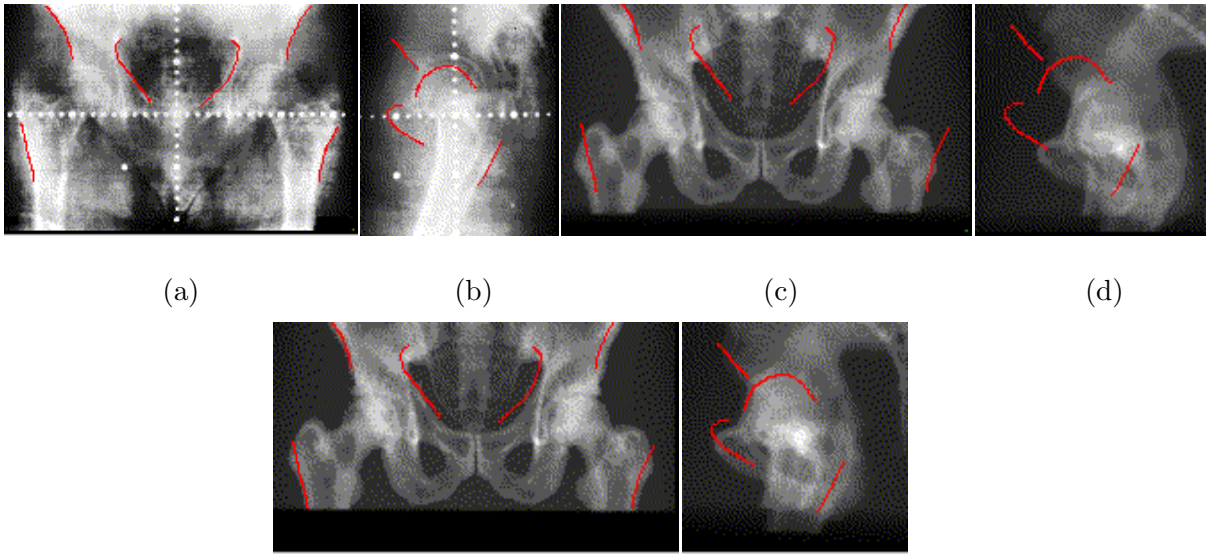


Figure 8.21: Recovery of setup variation using *actual patient data* and the MIME registration framework proposed in the previous chapter. (a,b) Portal images, (c,d) DRRs of 3DCT in original pose, (e,f) DRRs in corrected pose, (g,h) implicit segmentation of portals. See more detailed explanation in the text.

(e) and (f). Note the goodness of match of the contours to the anatomical features in the DRRs in figures 8.21 (e), (f) and thus verifying the accuracy of the estimated parameters. However, as can been seen from figure 8.21 (f), the final pose is still off from the correct registration by at–least 2 pixels.

The difference between the initial and the final parameters estimated by the MIME algorithm are $\theta_{XY} = 0.0738^o$, $\theta_{YZ} = 0.72259^o$, $\theta_{XZ} = 0.0177^o$, $t_x = 2.416$ voxels, $t_y = 12.3701$ voxels and $t_z = 11.7002$ voxels.

As a comparison of the MIME algorithm to the MI–based registration algorithm



(a) (b) (c) (d)

(e) (f)

Figure 8.22: Recovery of setup variation using *actual patient data* and the MI–based registration algorithm. (a,b) Portal images, (c,d) DRRs of 3DCT in original pose, (e,f) DRRs in corrected pose using MI–based algorithm.

on real patient data, figure 8.22 shows the performance of the MI–based algorithm on the same patient data. Note that the Y–translation and the Z–translation parameters as estimated by the MI–based algorithms are further from the true parameters compared to the parameters estimated by the MIME–based algorithm. Also, the MI–based algorithm seems to have introduced erroneous estimates of rotation angles. The difference between the initial and the final parameters estimated by the MI–based algorithm are $\theta_{XY} = 0.351614^o$, $\theta_{YZ} = 2.4984^o$, $\theta_{XZ} = 0.7289^o$, $t_x = 2.888$ voxels, $t_y = 9.7512$ voxels and $t_z = 9.43122$ voxels.

8.4.1 Performance Using Phantom Data

This section evaluates the performance of the minimax entropy registration framework and compares it with the performance of the mutual information based registration framework using two sets of anteriori–posteriori (AP) and left–lateral (LL) phantom portal images. After the first set of dual portal images is acquired, the phantom is moved by a known amount of rotation and translations before the second set of dual portal images is acquired. Figures 8.23 (a), (b) show the histogram equalized first set of AP and LL portal images acquired. The second set of AP and LL portal images are shown in the figures 8.24 (a) and (b).

The top row in table 8.1 shows the parameters by which the phantom was transformed before taking the second set of portal images. The size of each pixel is 2.73 mm. The portal images shown in figures 8.23 (a) and (b) are obtained from the PMMA phantom using 6MV energy X–rays. The X–ray films were then digitized using a 12–bit laser scanner. The resulting digitized portal images were then histogram equalized. Figures 8.23 (a), (b) show the adaptive histogram equalized AP and LL portal images. To obtain the second set of AP and LL portal images shown in figures 8.24 (a) and (b), the phantom was rotated by 1° about the Z–axis and then translated by 1 cm along both the X and Y axes. Then the AP and LL portal images were acquired using the 6MV X–rays. Figures 8.24 (a) and (b) show the adaptive histogram equalized portal images obtained. Since the size of each pixel is 2.73 mm, a 1 cm shift is equal to a shift by 3.66 pixels along the axes. The rotation about the X–axis, θ_{YZ}, is roughly 10°. Thus, for both algorithms, θ_{YZ}, was initialized *close* to 10°. It was initialized to 7° for both the MIME and MI algorithm.

Figures 8.23 (c) and (d) show the DRRs obtained from the 3DCT in the final

	t_x (vox)	t_y (vox)	t_z (vox)	θ_{YZ} (deg)	θ_{XZ} (deg)	θ_{XY} (deg)
MIME	0.55	-9.09	8.89	-11.89	1.81	2.75
MI	1.02	-5.2	10.86	-8.78	0.94	4.65

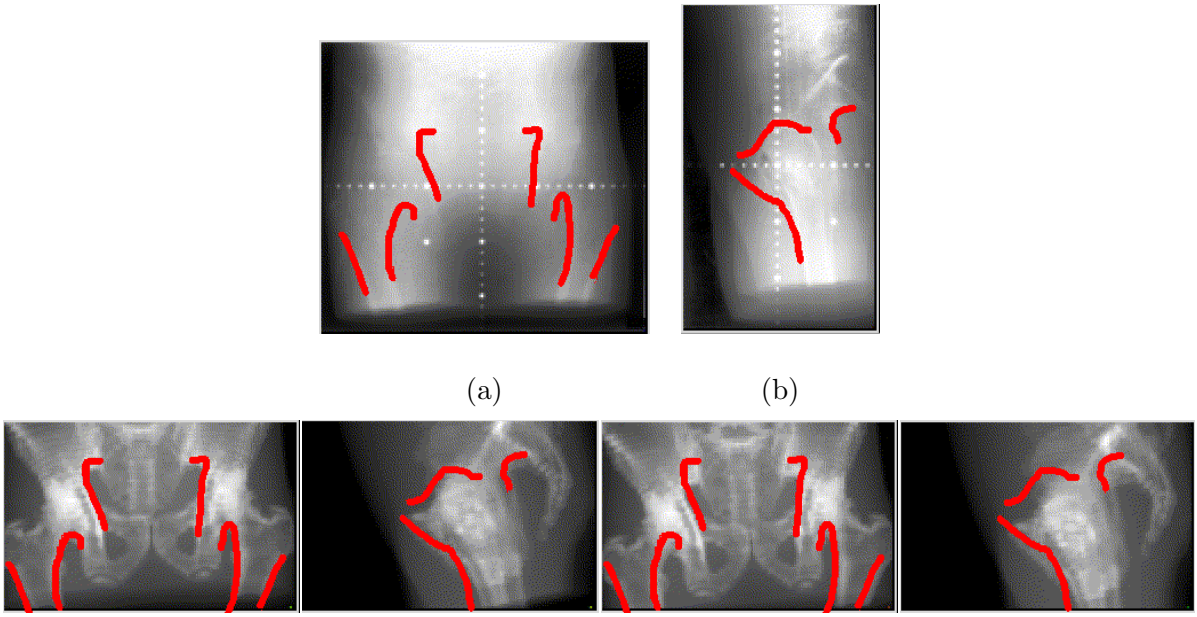


Figure 8.23: (a), (b) show AP and LL portal images obtained from phantom using 6MV X-rays. (c), (d) show the AP and the LL DRRs, respectively, obtained from the CT at the final estimated position using the MIME algorithm. The MIME algorithm models the images a 1D Markov random process. (e), (f) are the AP and the LL DRRs, respectively, obtained from the CT at the final estimated position using the MI–based registration algorithm. The table above shows the parameters estimated by the two algorithms.

position as estimated by the MIME algorithm and the figures 8.23 (e) and (f) show the DRRs obtained from the 3DCT in the final position as estimated by the MI–based algorithm. The table at the top of the figure shows the parameters estimated by the two algorithm. To visualize the accuracy of the estimated parameters, contours were hand drawn along features in the portal images and were then mapped over to the DRRs. From these figures, it can be seen that the MIME algorithm is roughly off by 3 to 4 voxels in estimating the t_y whereas the MI estimates are even poorer.

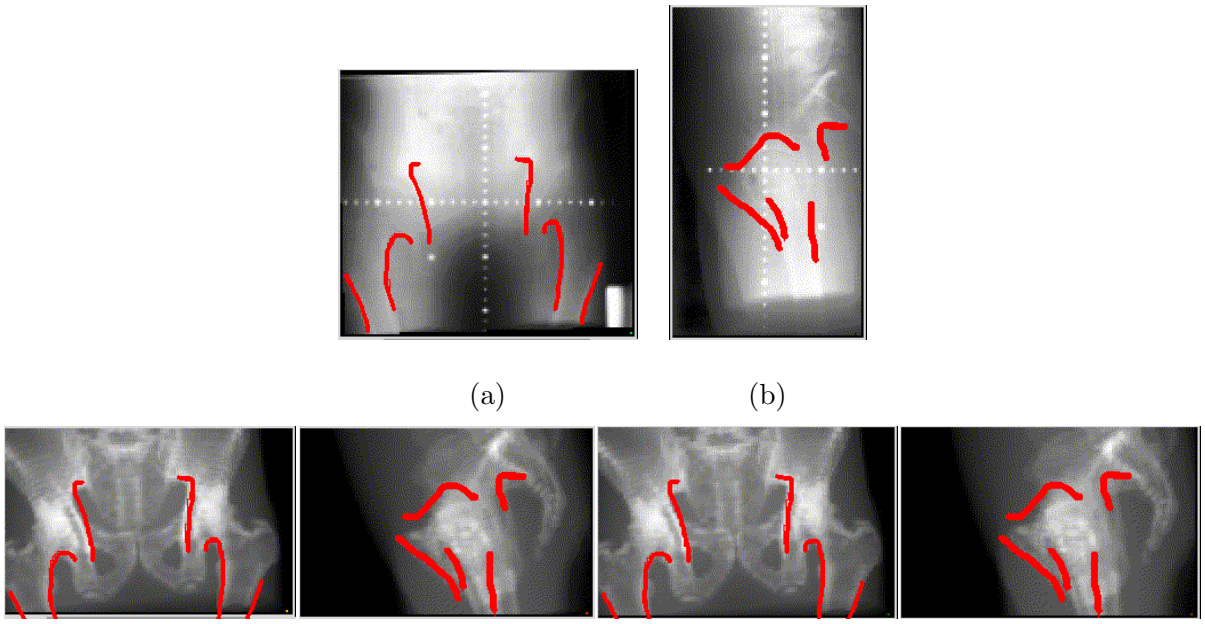
Similarly, figures 8.24 (c) and (d) show the DRRs obtained from the 3DCT in the final position as estimated by the MIME algorithm and the figures 8.24 (e) and (f) show the DRRs obtained from the 3DCT in the final position as estimated by the MI–based algorithm. Again, the contours are hand–drawn and then mapped over to the DRRs to visualize the accuracy of the estimated parameters. It can be noted that for this dataset also, except for t_y, the MIME–based algorithm has estimated the transformation parameters accurately.

The top row of the table 8.1 shows the parameters by which the phantom was rotated to obtain the second set of portal images. The second row of the table list the estimated difference in the position of the phantom as estimated by the MIME algorithm and the third row is the estimate obtained using the MI–based registration algorithm. Thus, for this problem, this table shows that the proposed MIME algorithm does a much better job in estimating the phantom position.

Sources of Errors Some of the sources of errors which might lead to inaccurate estimation of the transformation parameters are:

- After digitizing the portal images and before doing registration, the center of the field should be the center of the portal images. Thus, the digitized portal images

	t_x (vox)	t_y (vox)	t_z (vox)	θ_{YZ} (deg)	θ_{XZ} (deg)	θ_{XY} (deg)
MIME	-3.98	-5.10	8.67	-11.65	1.89	4.38
MI	-3.46	-5.37	10.82	-7.41	1.10	4.64



(a) (b)

(c) (d) (e) (f)

Figure 8.24: (a), (b) show AP and LL portal images obtained from phantom using 6MV X-rays. To obtain the portal images, the phantom was first rotated and translated by a known (from the position in which the portal images shown in figures 8.23 were obtained) amount on the treatment table and then the portals were obtained. The phantom was first rotated by 1° about the Z–axis, then translated by 1 cm along both X and Y axes. (c), (d) show the AP and the LL DRRs, respectively, obtained from the CT at the final estimated position using the MIME algorithm. The MIME algorithm models the images a 1D Markov random process. (e), (f) are the AP and the LL DRRs, respectively, obtained from the CT at the final estimated position using the MI–based registration algorithm. The table above shows the parameters estimated by the two algorithms.

	t_x (vox)	t_y (vox)	t_z (vox)	θ_{YZ} (deg)	θ_{XZ} (deg)	θ_{XY} (deg)
True	3.66	3.66	0	0	0	1
MIME	4.53	3.99	0.22	0.24	0.08	1.63
MI	4.48	0.17	0.04	1.37	0.16	0.01

Table 8.1: The top row of the table shows the parameters by which the phantom was transformed before taking the second set of portal images. The second row shows estimates of these parameters as estimated by using the proposed MIME algorithm. The third row shows the parameters estimated using the MI based registration algorithm.

are cropped to make the center of the field to be the center of the cropped portal images. Such a cropping can introduce an error of up–to a pixel along either axes in centering the field. This is important while utilizing dual portal images for patient setup verification as now the two portal images might not be consistent (i.e. there might not be a unique position of the 3DCT image such that it is registered to both the portal images).

- The field lines, including the cross hairs, should be parallel to the boundaries of the cropped portal images. Thus, it might be necessary to rotate the portal images to make the field lines parallel which might lead to an error of up–to 1° in–plane rotation for each portal image.

These two sources of errors are important and need to be carefully taken care of before moving the algorithm to a clinical setting.

Estimation of Y–translation From this example, it can be seen that while the MIME–based algorithm was quite accurate in estimating t_x, the error in the estimation of t_y was of the order of 2 to 3 voxels. Similar results were observed while registering image data set obtained from a real patient. Note that in our minimax entropy registration framework, we have modeled 2D images as being 1D random processes by concatenating the rows. In this model only the vertical edges were being modeled and all the horizontal edges will be lost. Since using this model we are getting good estimates on t_x, it will be interesting to see whether modeling the images as 2D Markov random fields improve the estimation on both t_x and t_y and in particular the estimation of t_y.

8.5 Summary

Performance analysis using simulated portal images datasets verifies the accuracy and the robustness of the minimax entropy registration algorithm in the presence of both noise and blur in the portal images. In general, the minimax entropy registration framework leads to more accurate estimates of the parameters compared to the mutual information–based registration algorithm, especially in the presence of blur in the simulated portal images. The minimax entropy algorithm even estimated the the out–of–plane angle, θ_{XZ}, within 1° of the true value for a large range of noise and blur. Performance analysis on the real patient portal images shows that while the proposed algorithm, in general, estimated the alignment correctly, it is off by about 2 to 3 voxels along the Y–axis. The use of better portal images, possibly scanned using 12–bit laser scanner, is expected lead to accurate registration of the portal images to the 3DCT image data.

Chapter 9

Future Work

9.1 Introduction

Comparisons of the minimax entropy registration framework to other competing methods, especially the ridge–based registration method [44], in chapter 6 showed that the feature–based registration methods can lead to accurate estimation of the parameters, if the features have been accurately estimated. A number of feature–based registration algorithms have been proposed in the literature, with the algorithm proposed by Lavallee *et al.* [66] being of special interest for this chapter. The downfall of the feature–based methods is that in the presence of noise and blur in the images, accurate feature detection and localization requires human intervention.

The minimax entropy–based registration method (MIME) developed in chapter 7 incorporates the information that the neighboring pixel intensities in the images are correlated, except at the boundaries of regions. A simple 1D MRF, an Ising chain, model was used to capture neighborhood correlations. *Line processes* [43] were incorporated to model boundaries, or the pixel intensity discontinuities, between regions. Assuming that

the pixel intensity distribution can be modeled using a MRF leads to regularization of the estimated segmentation of the portal images, which in turn leads to better localization and detection of edges.

As the edge information has already been estimated during the *max* step of the minimax entropy algorithm, we propose to incorporate this information into the parameter estimation step, the *min* step, also. Incorporation of accurately–localized edges is expected to further improve the accuracy of the proposed minimax entropy algorithm. This chapter develops the underlying mathematics, but leaves out the verification of the accuracy of the strategy as future work.

9.2 Formulation Incorporating Edge Information

As mentioned in the introduction, the edge information has already been estimated in the *max* step as formulated in chapter 7. This section only develops the *min* step while incorporating the estimated edge information. Let $\mathbf{G}_e$ be the $N \times N \times N$ random variable denoting edges obtained from $\mathbf{G}$. The transformation parameters are estimated by minimizing the joint conditional entropy as follows:

Min step:

$$\begin{aligned}
\hat{T} &= \arg\min_T H(\mathcal{E}, M, X|G, G_e, T) \\
&= \arg\min_T \left[H(M, X|G, G_e, T, \mathcal{E}) + H(\mathcal{E}|G, G_e, T)\right] \\
&\leq \arg\min_T \left[H(M, X|G, T) + H(\mathcal{E}|G_e, T)\right] \\
&= \arg\min_T \left[H(M, X|Y(T)) + H(\mathcal{E}|G_e, T)\right] \\
&= \arg\min_T \left[H(M, X, Y) - H(Y) + H(\mathcal{E}|G_e, T)\right]
\end{aligned}$$

$$
\begin{aligned}
&= \arg\min_{T} \left[H(M, X, Y) - H(Y) + H(\mathcal{E}, G_e, T) \right] \\
&= \arg\min_{T} \left[H(X, Y|M) + H(M) - H(Y) + H(\mathcal{E}, G_e, T) \right] \\
&= \arg\min_{T} \left[H(X, Y|M) - H(Y) + H(\mathcal{E}, G_e, T) \right]
\end{aligned}
$$

We have dropped the terms $H(M),\ H(G_e, T)$ as they are constant. The inequality follows from the fact that conditioning reduces entropy. The first term is similar to the conditional entropy which we minimize in [4]. However, a subtle point to note is that this formulation incorporates correlation between the neighboring pixels. The second term minimizes the conditional entropy of the edges in the portal image and the edges from the 3D CT space.

To formulate the problem in the entropy framework, we shall assume that $\mathbf{G}_e$ is the signed distance transformation of the edges in three space. To obtain $\mathbf{G}_e$, first, the 3D surface extracted from $\mathbf{G}$ is distance transformed. Then all distances in the distance transformed image are set to negative values. Further, a positive sign is spread from an initial exterior point. This recursive spreading of the positive sign from points to their neighbors is terminated at points close to the 3D surface. Note that after the recursive spreading has stopped, distances outside the surface are positive–valued and distances within the surface are negative valued.

The distance of an edge, in 2D portal image, to the surface of pelvic bone, in a 3D CT image, is defined as follows. A ray is back–projected from the edge feature to the radiation source. Then the minimum distance along the ray to the 3D pelvic surface is defined as the distance of the edge from the pelvic surface. Using a signed distance transform of the 3D pelvic bone surface uniformly handles the case where the back–projected ray cuts through the bone surface. Once a signed distance transformed

3D image $\mathbf{G}_e$ is obtained, a 2D distance projection image, $\mathbf{S}(\mathbf{T}) = \mathbf{S}'$, at the current estimate of transformation parameters, is obtained in the following manner. The ith pixel value, in $\mathbf{S}(\mathbf{T}) = \mathbf{S}'$, is the minimum distance in the signed distance transformed 3D CT along the projection ray from the ith pixel to the radiation source. When the 3D CT is aligned to the 2D portal images, the projection rays from the edges in 2D portal images will be tangential to the 3D surface and hence the minimum distance along these projection rays will be zero. In order to estimate the registration parameters, the entropy of the minimum distances along the projection rays from the edges in portal images is minimized. Therefore, the joint entropy term $H(\mathcal{E}, G_e, T)$, which is defined as $H(\mathcal{E}, S')$, is minimized to estimate the transformation parameters. M. van Herk *et al.* [114] showed that, for portal images, image artifacts, organ motion and poor segmentation may lead to detection of false edges in portal images which might decrease the accuracy of the registration algorithm. This observation requires a match method which is robust to these outliers. Collignon *et al.* [27] also proposed to minimize the entropy of distances between corresponding points, while registering two point sets. We feel that the minimization of the entropy of distances will lead to robustness of the algorithm in the presence of outliers.

The joint entropy term $H(\mathcal{E}, S')$ can be further simplified to be (see appendix D.5.3 for details):

$$H(\mathcal{E}, S') = N \cdot < \overline{e} > \; H_1(s')$$

where,

$$< \overline{e} > \equiv \frac{1}{N} \sum_{i=1}^{N} < e_i > = \frac{1}{N} \sum_{i=1}^{N} P(\mathbf{e}_i = 1)$$

and appendix D.6.3 evaluates the Parzen window estimates and derivatives of the entropy term $H_1(s')$.

Thus, the *min* step can be summarized as:

<u>Min Step:</u>

$$\begin{aligned} \hat{T} &\leq \arg\min_T \left[H(X,Y|M) - H(Y) + H(\mathcal{E}, G_e, T) \right] \\ &= \arg\min_T \left[H(X,Y|M) - H(Y) + H(\mathcal{E}, S') \right] \end{aligned}$$

with the various entropy terms being evaluated as:

$$\begin{aligned} H(X,Y|M) &\approx (N-1) \left[\sum_{\{\alpha,\beta\}} < \overline{z_\alpha z_\beta} > H_{\alpha\beta}(x_2, y_2, x_1, y_1) - \sum_{\alpha \in A} < \overline{z_\alpha} > H_\alpha(x,y) \right] \\ H(Y) &\approx N \cdot H(y) \\ H(\mathcal{E}, S') &= N \cdot < \overline{e} > \; H_1(s') \end{aligned}$$

9.2.1 Incorporation of Whole Boundary Information

The *min* step formulation above incorporated edge information into the minimax entropy framework. However, using individual detected edges may not be robust, especially in the presence of noise. Thus, it is felt that possible grouping of edges into whole object boundaries, extracted from portal images, may further constrain the registration. Therefore, incorporation of whole boundary information into the registration framework remains a

direction for our future research efforts.

9.2.2 Using higher order MRFs

The minimax entropy framework, developed in chapter 7, modeled images as 1D Markov random fields (MRFs), thus incorporating only the correlation among the nearest neighbor pixels in the same row. Such a 1D MRF model can capture only the vertical edges in the portal images. 2D MRF models with 1st or higher order neighborhoods can be employed to capture horizontal edges. Thus, in future, we would like to study the effects of higher order Markov random field models of portal images on the estimated segmentation and registration parameters.

9.3 Organ Motion and Using Ultrasound Data

The minimax entropy registration framework was developed mainly for patient setup verification in prostate cancer treatment. Patient setup verification was achieved largely by using the bony anatomy visible in the portal images and the 3D CT data set. However, for the problem of prostate cancer treatment using radiotherapy, the real goal is to identify and localize soft tissue position. The prostate may move, on a day–to–day basis, to a large extent with respect to the bony anatomy, depending upon the bladder and rectum filling [1, 31]. Thus, there are inherent errors in using the bony anatomy to localize the prostate and so the gains in patient positioning accuracy can be superseded by the uncertainty introduced in linking the bony anatomy to the prostate. Thus, there is a need to incorporate organ motion compensation into the prostate cancer radiotherapy. As future work, we would like to integrate intra–treatment ultrasound for measuring prostate motion. If the prostate can be segmented from several ultrasound views taken

during a particular treatment fraction and then registered to the internal anatomy at both day 1 of the treatment and the current treatment, estimated prostate motion can be used to more accurately adjust the treatment to the actual position of the prostate.

9.4 Summary

This chapter develops the minimax registration framework which incorporates the edge information estimated in the *max* step. Incorporation of the edge information leads to a *three–way* integrated registration framework. The *min* step, using the estimated edge information integrates the sparse–field based and the dense–field based registration methods. Accurate detection and localization of the features used in registration necessitates human intervention in most feature based registration methods. Whereas the dense–field registration methods are robust to noise and blur in the images. Thus, in the initial iterations of the minimax entropy algorithm, the registration is guided by the dense information, the pixel intensities. During these iterations the edges are detected and localized automatically in the *max* step. These accurately–detected and localized edge information can then guide the registration in the later iterations, leading to an accurate estimate of the registration parameters. An important point to keep in mind is that in the minimax entropy registration framework registration is guided by the bony anatomy as only the bony structures are visible in the portal images. However, for the external beam radiotherapy (EBRT) of the prostate cancer it is necessary to localize the prostate which might move relative to the bony anatomy.

Chapter 10

Summary

This thesis presented an integrated information theoretic framework where the segmentation and the registration of the dual portal images to the 3D CT dataset are carried out simultaneously and iteratively. The goal of this registration framework is an iterative strategy where a rough estimate of segmentation of a low resolution image is used to better estimate the registration parameters and an estimate of the registration parameters is used to better segment the low resolution image by making use of information from the high resolution image.

The proposed registration framework, called the *minimax* entropy framework, has two steps, the *max* step and the *min* step. The *max* step and the *min* step evaluate appropriate entropies to estimate the probability distribution on the segmentation labels and the registration parameters which bring the 3D CT dataset into alignment with the dual portal images. The registration framework does not assume anything about the underlying dataset being registered and estimates various probability density functions from the given dataset itself. Thus, the proposed registration framework can be easily extended for the registration of any low resolution image to a high resolution image.

Experimental results show that the incorporation of segmentation information into the registration framework leads to a registration algorithm which is robust to both noise and blur in the portal images. The robustness of the algorithm to noise and blur is evaluated using simulated portal images. The segmentation of the portal image as estimated by the proposed algorithm was compared to the segmentation obtained using other strategies. These comparisons showed that the proposed registration framework leads to better segmentation of the portal images. The reason why the MIME algorithm is expected to better segment portal image is because of the fact that the algorithm is using information from the high resolution 3D CT image.

Incorporation of the correlation information into the registration framework led to better estimate of the segmentation and accurate registration. However, this increased the computational complexity of the algorithm.

Experiments with a ridge–based method [44] showed that the feature based registration algorithms can lead to very accurate estimation of the registration parameters if the features have been accurately detected and localized. Thus, for the future work, registration framework has been formulated to make use of the edge information, estimated in the *max* step, in the *min* step to increase the accuracy of the estimated registration parameters.

Incorporation of the edge information into the *min* step will lead to a *three–way* integrated registration framework. On one hand, the framework has successfully integrated the dual problem of segmentation and registration of low resolution, low contrast image to a high resolution, high contrast image. On the other hand, incorporation of the edge information will lead to integration of dense–field and sparse–field based registration methods. Sparse–field based methods are plagued by the inaccuracies in detection and

localization of the features to be used for registering images. However, if the features are accurately detected and localized, these methods can lead to accurate estimation of the registration parameters. Dense–field methods are robust to noise and blur and so the proposed framework uses the gray–scale pixel intensities to estimate the registration parameters in the initial iterations of the algorithm. Better estimates of the segmentation of the portal images leads to better estimates of the edges. The estimated edge information can then be used in the latter iterations to further improve the image registration.

In short, this thesis presents a registration framework which is robust to both noise and blur in the portal images and leads to an accurate estimation of the registration parameters. However, the incorporation of edges and whole boundary information into the registration framework may further improve the accuracy of the estimated parameters. For patient setup verification during EBRT, one should also take into consideration the movement of the prostate with respect to the bony anatomy.

Appendix A

Relation between Correlation Coefficient and Mutual Information

Recently in the literature [117, 104, 27] mutual information has been used to register multi–modality images. For the same modality images correlation might still be the right metric to use. This section shows, under some assumptions, that the mutual information is more sensitive to mis–registration compared to the correlation coefficient.

Let the pixel intensities of the reference image be denoted by a random variable $\mathbf{x}$ which is assumed to be Gaussian distributed with mean μ_x and standard deviation σ_x. Thus, its density function can be written as:

$$p_{\mathbf{x}}(x) = \frac{1}{\sigma_x\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{x-\mu_x}{\sigma_x}\right)^2\right]$$

Similarly, let the pixel intensities of the transformed image be denoted by a random

variable $\mathbf{y}$ which is also assumed to be Gaussian distributed with mean μ_T and standard deviation σ_T. Note that the mean and variance of $\mathbf{y}$ are a function of transformation parameters T.

$$p_{\mathbf{y}}(y) = \frac{1}{\sigma_T\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{y-\mu_T}{\sigma_T}\right)^2\right]$$

Assume further that the joint density function, $p_{\mathbf{x},\mathbf{y}}(x,y)$, is also a jointly Gaussian density function specified as:

$$p_{\mathbf{x},\mathbf{y}}(x,y) = \frac{1}{2\pi|\Psi|^{1/2}} \exp\left[-\frac{1}{2}(x-\mu_x \;\; y-\mu_T)\Psi^{-1}\begin{pmatrix} x-\mu_x \\ y-\mu_T \end{pmatrix}\right]$$

where Ψ is the covariance matrix given as

$$\Psi = \begin{pmatrix} \sigma_x^2 & r_T\,\sigma_x\sigma_T \\ r_T\,\sigma_x\sigma_T & \sigma_T^2 \end{pmatrix}$$

We shall assume that as the images are registered, the form of their joint densities and the marginal remain Gaussian, while possibly the mean, variance and the correlation coefficients change. The reference image stays unchanged, and hence $p_{\mathbf{x}}(x)$ is independent of T, transformation parameters.

The mutual information between the random variables $\mathbf{x}$ and $\mathbf{y}$ is defined to be:

$$I(x,y) = H(x) + H(y) - H(x,y)$$

Differentiating the equation with respect to the transformation parameters T, we have

$$\frac{\partial}{\partial T} I(x,y) \quad = \quad \frac{\partial}{\partial T} H(y) - \frac{\partial}{\partial T} H(x,y) \tag{A.1}$$

Let us evaluate the two terms on the right side of the equation separately.

A.0.1 Evaluation of $\frac{\partial}{\partial T} H(y)$:

Since $\mathbf{y}$ is assumed to be Gaussian distributed, the entropy of the distribution $p_{\mathbf{y}}(y)$ can easily be shown to be:

$$H(y) = \ln\left(\sigma_T \sqrt{2\pi e}\right)$$

Differentiating both side by T, we have

$$\frac{\partial}{\partial T} H(y) = \left(\frac{1}{\sigma_T}\right) \frac{\partial}{\partial T} \sigma_T \tag{A.2}$$

A.0.2 Evaluation of $\frac{\partial}{\partial T} H(x,y)$:

The inverse of the covariance matrix, Ψ, is given as:

$$\Psi^{-1} = \frac{1}{\sigma_x^2 \sigma_T^2 (1 - r_T^2)} \begin{pmatrix} \sigma_T^2 & -r_T \sigma_x \sigma_T \\ -r_T \sigma_x \sigma_T & \sigma_x^2 \end{pmatrix}$$

Thus, the joint entropy can be evaluated as:

$$H(x,y) = -\int\int p(x,y) \ \ln p(x,y) \ dx \ dy$$

Since $p(x, y)$ is jointly Gaussian, it can easily be shown,

$$\begin{aligned} H(x,y) &= \ln\left(2\pi\sigma_x\sigma_T\sqrt{1-r_T^2}\right) \\ &\quad + \int\int p(x,y)\,\frac{1}{2}\left\{[x-\mu_x \;\; y-\mu_T]\Psi^{-1}\begin{pmatrix} x-\mu_x \\ y-\mu_T \end{pmatrix}\right\} dx\,dy \end{aligned}$$

which can be further simplified to:

$$\begin{aligned} &H(x,y) - \ln\left(2\pi\sigma_x\sigma_T\sqrt{1-r_T^2}\right) \\ &= \iint \frac{p(x,y)}{2\sigma_x^2\sigma_T^2(1-r_T^2)}\left[\sigma_T^2(x-\mu_x)^2 - 2r_T\sigma_x\sigma_T(x-\mu_x)(y-\mu_T) + \sigma_x^2(y-\mu_y)^2\right] dx\,dy \end{aligned}$$

The last equation can be further simplified as follows:

$$\begin{aligned} &2\sigma_x^2\sigma_T^2(1-r_T^2)\left[H(x,y) - \ln\left(2\pi\sigma_x\sigma_T\sqrt{1-r_T^2}\right)\right] \\ &\quad = \quad 2\sigma_T^2\sigma_x^2 - 2r_T\sigma_T\sigma_x\iint p(x,y)(x-\mu_x)(y-\mu_T)\,dx\,dy \\ &\quad = \quad 2\sigma_T^2\sigma_x^2 - 2r_T^2\sigma_T^2\sigma_x^2 \end{aligned}$$

Therefore, we can write

$$\begin{aligned} H(x,y) &= \ln\left(2\pi\sigma_x\sigma_T\sqrt{1-r_T^2}\right) + 1 \\ &= \ln\left(2\pi e\sigma_x\sigma_T\sqrt{1-r_T^2}\right) \end{aligned}$$

Differentiating the last equation with respect to T, we have,

$$\frac{\partial}{\partial T}H(x,y) = \frac{1}{\sigma_T}\frac{\partial}{\partial T}\sigma_T - \frac{r_T}{1-r_T^2}\frac{\partial}{\partial T}r_T \tag{A.3}$$

Thus, from equations (A.1), (A.2) and (A.3), we can now write:

$$\frac{\partial r_T}{\partial I(x,y)} = \left(\frac{r_T}{(1-r_T^2)}\right)\frac{\partial r_T}{\partial T} \tag{A.4}$$

From equation (A.4), it is clear that a small change in transformation parameters, close to the true registration, will lead to a large change in the mutual information between the random variables. Thus, it is expected that the mutual information–based registration method will be more sensitive to the mis–registration between same modality images. It is also easy to show, from equation (A.4), that the mutual information is more sensitive to mis–registration when $r_T \geq \frac{-1+\sqrt{5}}{2}$, the *golden ratio*, $\left(\frac{r_T}{(1-r_T^2)}\right) \geq 1$.

Figure A.1 plots the graph of $\left(\frac{r_T}{(1-r_T^2)}\right)$ as a function of r_T. Note that when the images are reasonably correlated, that is when $r_T \geq \frac{-1+\sqrt{5}}{2}$, a small change in the correlation coefficient will lead to a large change in the mutual information.

A.1 Extensions to Mixture of Gaussian Density Functions

This result can be easily extended to a more general case where the joint density between the pixel intensities of the two images to be registered can be modeled as a weighted sum of Gaussians, that is, if the joint density function can be written in the form:

$$p(x,y) = \sum_{a\in A}\alpha_a \frac{1}{2\pi|\Psi_a|^{1/2}}\exp\left[-\frac{1}{2}w_a(x)\Psi_a^{-1}w_a'(x)\right]$$

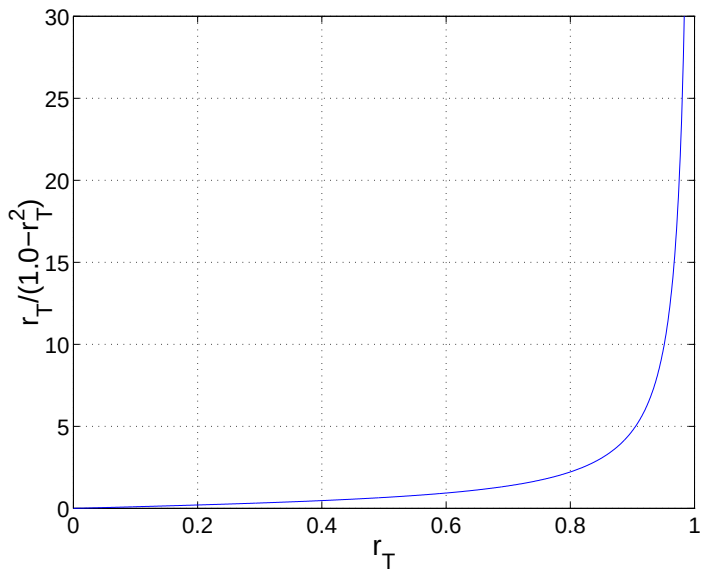


Figure A.1: The graph plots $\left(\frac{r_T}{(1-r_T^2)}\right) \geq 1$ as a function of correlation coefficient r_T.

where, $w_a(x) = (x - \mu_x \; y - \mu_T)$ and Ψ_a is the covariance matrix given as

$$\Psi_a = \begin{pmatrix} \sigma_{ax}^2 & r_{aT}\ \sigma_{ax}\sigma_{aT} \\ r_{aT}\ \sigma_{ax}\sigma_{aT} & \sigma_{aT}^2 \end{pmatrix}$$

Then it can be easily shown that:

$$\begin{aligned} \frac{\partial}{\partial T} I(x,y) &= \frac{\partial}{\partial T} H(y) - \frac{\partial}{\partial T} H(x,y) \\ &= \sum_{a \in A} \alpha_a \left(\frac{r_{aT}}{(1 - r_{aT}^2)} \right) \frac{\partial r_{aT}}{\partial T} \end{aligned} \tag{A.5}$$

Appendix B

Derivation of the EM steps

This section develops the E–step and the M–step of the EM algorithm for our problem. Specifically, we will show the development of the equation (4.10):

$$<z_{ai}>^k = \left(\frac{<z_{ai}>^{k-1} \; p_a(x_i, y_i)}{\sum_{b \in A} <z_{bi}>^{k-1} \; p_b(x_i, y_i)} \right)$$

where, $y_i = y(i, T^{(k-1)})$.

Usually within the EM framework spatially stationary priors on the probability of the tissue labels are used which remain constant throughout the iterations (these priors are also called *occupancy probabilities* of a pixel belonging to a different tissue classes). If these priors are not known, then they can also be estimated in the EM algorithm. However, in our experience on the segmentation of the portal image, such spatially stationary priors (even when the true values were supplied to the algorithm), invariably lead to poor estimates on the segmentation. Therefore, we do not assume spatially stationary priors but let the priors change with iterations. The estimated probability density function from the previous iteration is used as the prior information in the current iteration. This is

justified due to the fact that in each iteration we are estimating the probability density function in the *max* step and also this leads to much better segmentation of the portal image. Spatially varying priors on the tissue class for segmentation has also been proposed and used by Kapur *et al.* [61].

B.0.1 Development of the E–step

The conditional density function of the ith pixel intensity is given as:

$$\begin{aligned} p(x_i|y_i) &= \sum_{b \in A} P(\mathbf{m}_i = b)\, p(x_i|y_i, \mathbf{m}_i = b) \\ &= \sum_{b \in A} P_i(b)\, p_b(x_i|y_i) \\ &= \prod_{a \in A} \left[\sum_{b \in A} P_i(b)\, p_b(x_i|y_i) \right]^{z_{ai}} \end{aligned}$$

The joint density function of the pixel label and the intensity at the ith pixel, given the pixel intensity at the ith pixel in the DRR can be written as:

$$\begin{aligned} p(\mathbf{m}_i = c, x_i|y_i) &= P(\mathbf{m}_i = c)\, p(x_i|y_i, \mathbf{m}_i = c) \\ &= P_i(c)\, p_c(x_i|y_i) \\ &= \prod_{a \in A} [P_i(c)\, p_c(x_i|y_i)]^{z_{ai}} \end{aligned}$$

Thus, we have

$$
\begin{aligned}
P(m_i|x_i, y_i) &= \frac{p(m_i, x_i|y_i)}{p(x_i|y_i)} \\
&= \frac{p(m_i, x_i|y_i)}{\sum_{b\in A} p(\mathbf{m}_i = b, x_i|y_i)} \\
&= \frac{P_i(c)\ p_c(x_i|y_i)}{\sum_{b\in A} P_i(b)\ p_b(x_i|y_i)} \\
&= \prod_{a\in A} \left[\frac{P_i(c)\ p_c(x_i|y_i)}{\sum_{b\in A} P_i(b)\ p_b(x_i|y_i)} \right]^{z_{ai}}
\end{aligned}
$$

Therefore, assuming that each pixel is independent, the posterior distribution can be written in the form:

$$
\begin{aligned}
P(M|X, Y) &= \prod_i p(m_i|x_i, y_i) \\
&= \prod_i \prod_{a\in A} \left[\frac{P_i(c)\ p_c(x_i|y_i)}{\sum_{b\in A} P_i(b)\ p_b(x_i|y_i)} \right]^{z_{ai}} \\
&\equiv \prod_{ai} \left[\frac{P_i(c)\ p_c(x_i|y_i)}{\sum_{b\in A} P_i(b)\ p_b(x_i|y_i)} \right]^{z_{ai}}
\end{aligned}
$$

Thus, the conditional and the log conditional density function $p(X|Y)$ can be written as:

$$
\begin{aligned}
p(X|Y) &= \frac{p(M, X|Y)}{p(M|X, Y)} \\
\ln p(X|Y) &= \ln p(M, X|Y) - \ln p(M|X, Y) \qquad \text{(B.1)}
\end{aligned}
$$

Taking expectation on the both sides of the equation (B.1) with respect to $P(M|X,Y)$, the *E–Step* of the EM algorithm can be written as:

$$\begin{aligned}\ln p(X|Y) &= E_M[\ln p(M,X|Y)|X,Y] - E_M[\ln P(M|X,Y)|X,Y] \\ &\geq E_M[\ln p(M,X|Y)|X,Y] - E_M[\ln P(M|X,Y)|X,Y] \qquad \text{(B.2)}\end{aligned}$$

where, $E_M[\cdot]$ is defined to be the expected value with respect to the probability distribution $P(M|X,Y)$ and written as:

$$E_M[\ln p(M,X|Y)|X,Y] \equiv \sum_M P(M|X,Y)\ \ln p(M,X|Y)$$

and the inequality in equation (B.2) follows from the Kullback–Leibler (KL) inequality.

B.0.2 Development of the M–step

In the M–step, the first term on the right side of the equation (B.2) is maximized to estimate the transformation parameters which is simplified as follows:

$$\begin{aligned}\arg\max_T \ln p(X|Y) &= \arg\max_T E_M[\ln p(M,X|Y)|X,Y] \\ &= \arg\max_T E_M\left[\sum_{ai} z_{ai} \ln\left(P_i(c)\ p_c(x_i|y_i)\right)\right] \\ &= \arg\max_T \sum_M \left[P(M|X,Y) \sum_{ai} z_{ai} \ln\left(P_i(c)\ p_c(x_i|y_i)\right)\right] \\ &= \arg\max_T \sum_M \left[\prod_{di}\left[\frac{P_i(c)\ p_c(x_i|y_i)}{\sum_{b\in A} P_i(b)\ p_b(x_i|y_i)}\right]^{z_{di}} \sum_{ai} z_{ai} \ln\left(P_i(c)\ p_c(x_i|y_i)\right)\right] \\ &= \arg\max_T \sum_M \left[\sum_{ai} z_{ai} \prod_{di}\left[\frac{P_i(c)\ p_c(x_i|y_i)}{\sum_{b\in A} p_i(b)\ p_b(x_i|y_i)}\right]^{z_{di}} \ln\left(P_i(c)\ p_c(x_i|y_i)\right)\right]\end{aligned}$$

$$
\begin{aligned}
&= \arg\max_{T} \sum_{ai} \left[\left(\frac{P_i(a)\ p_a(x_i|y_i)}{\sum_{b\in A} P_i(b)\ p_b(x_i|y_i)} \right) \ln\left(P_i(a)\ p_a(x_i|y_i)\right) \right] \\
&= \arg\max_{T} \sum_{ai} \left[<z_{ai}> \ln(P_i(a) p_a(x_i|y_i)) \right] \\
&= \arg\max_{T} \sum_{ai} \left[<z_{ai}> \left(\ln P_i(a) + \ln p_a(x_i|y_i)\right) \right] \\
&= \arg\max_{T} \sum_{ai} \left[<z_{ai}> \ln p_a(x_i|y_i) \right]
\end{aligned}
$$

where,

$$
\begin{aligned}
<z_{ai}> &= \left(\frac{P_i(a)\ p_a(x_i|y_i)}{\sum_{b\in A} P_i(b)\ p_b(x_i|y_i)} \right) \\
&= \left(\frac{P_i(a)\ p_a(x_i, y_i)}{\sum_{b\in A} P_i(b)\ p_b(x_i, y_i)} \right)
\end{aligned}
$$

where we have assumed that pixel intensities in the DRR, y_i, are independent of the portal image segmentation labels. This will be the case especially when the two images are not registered.

Priors on the segmentation labels are assumed to be spatially varying, with current estimates on the density of the segmentation labels being used as the priors in the next iteration. Thus, we have,

$$
<z_{ai}>^k = \left(\frac{<z_{ai}>^{(k-1)}\ \ p_a(x_i, y_i)}{\sum_{b\in \mathbf{A}} <z_{bi}>^{(k-1)}\ \ p_b(x_i, y_i)} \right)
$$

Appendix C

Developing the Minimax Entropy Registration Framework

This appendix develops the various results used in the development of the minimax entropy algorithm as described in the chapter 5.

C.1 Why Evaluate Expected Values

Suppose the problem is to estimate the transformation parameters **T** which will bring the two images, **X** and **Y** into alignment. If the joint density function between the pixel intensities of the two images, say $p_t(X, Y)$, is known, the transformation parameters which optimize the log likelihood of the two images can then be estimated as:

$$\begin{aligned}
\hat{T} &= \arg\max_T \ln p_t(X|Y, T) \\
&= \arg\max_T \ln p_t(X|Y') \\
&= \arg\max_T \ln \frac{p_t(X, Y')}{p_t(Y')}
\end{aligned}$$

$$= \quad \arg\max_{T} \left[\ln p_t(X, Y') - \ln p_t(Y')\right] \tag{C.1}$$

Suppose now that the joint density function, $p_t(X, Y)$, is not known, which is usually the case when registering multi–modal images. Now if we estimate the joint density function, say $p(X, Y')$, from the given data at the current estimate of the transformation parameters, say T^k, and use this estimated density in equation (C.1), then the optimal parameters, $\hat{T}$, estimated will be T^k itself (as the joint density was estimated at this transformation parameters and thus this estimate of the parameters will explain the density best).

However, now assume that the intensities in the two images are such that, when properly registered, the intensities in one image explain the intensities in the other image, that is, there is a relation between the two images, which need not be linear. Thus, the optimal estimated joint density function, $p(X, Y')$, should be one which best explains all the pixel intensities in the two images. Since $p_t(X, Y)$ denotes the true joint density function, the expected value of the log–likelihood with respect to the true density function is given as:

$$\begin{aligned}
E_t[l(T)] &= \int\int p_t(X, Y') \ln p(X|Y')\, dX\, dY' \qquad \text{(C.2)} \\
&= \int\int p_t(X, Y') \ln p(X, Y')\, dX\, dY' - \int\int p_t(X, Y') \ln p(Y')\, dX\, dY' \\
&= \int\int p_t(X, Y') \ln p(X, Y')\, dX\, dY' - \int p_t(Y') \ln p(Y')\, dY' \\
&\leq \int\int p_t(X, Y') \ln p_t(X, Y')\, dX\, dY' - \int p_t(Y') \ln p_t(Y')\, dY' \\
&= H_t(Y') - H_t(X, Y') \\
&= H_t(X) + H_t(Y') - H_t(X, Y')
\end{aligned}$$

$$= \quad I_t(X, Y')$$

Thus, the mutual information between the two random variables, when the true transformation parameters are known, forms the upper bound of the expected value of the log likelihood of the data. Note that if we instead take the expectation of the log–likelihood with respect to $p(X, Y')$, we get,

$$\begin{aligned}
E[l(T)] &= \int\int p(X, Y') \ln p(X|Y') \; dX \; dY' \\
&= \int\int p(X, Y') \ln p(X, Y') \; dX \; dY' - \int\int p(X, Y') \ln p(Y') \; dX \; dY' \\
&= \int\int p(X, Y') \ln p(X, Y') \; dX \; dY' - \int p(Y') \ln p(Y') \; dY' \\
&= -H(X, Y') + H(Y') \\
&= H(X) + H(Y') - H(X, Y') \\
&= I(X, Y')
\end{aligned}$$

Thus, maximization of the expected value of the log–likelihood is equivalent to the maximization of the mutual information.

C.2 Relation Between the EM and the Min Step

In this section, starting from equation (4.8) we formally develop the equation (5.4), explicitly stating the assumptions and our motivation for the final formulation of the *min* step.

Equation (4.8) can be rewritten as:

$$\begin{aligned} Q(T, T^{(k-1)}) &= \sum_i \left(\sum_{a \in A} <z_{ai}>^k \ \ln p_a(x_i, y_i) - \ln p(y_i) \right) \\ &= \sum_i Q_i(T, T^{(k-1)}) \end{aligned}$$

Taking expected value of $Q_i(T, T^{(k-1)})$ with respect the the joint density function $p(x_i, y_i)$, we have,

$$\begin{aligned} E[Q_i(T, T^{(k-1)})] &= \iint p(x_i, y_i) Q_i(T, T^{(k-1)}) \ dx_i \ dy_i \\ &= \iint p(x_i, y_i) \left[\sum_{a \in A} <z_{ai}>^k \ln p_a(x_i, y_i) - \ln p(y_i) \right] \ dx_i \ dy_i \\ &= \sum_{a \in A} <z_{ai}>^k \iint p(x_i, y_i) \ln p_a(x_i, y_i) \ dx_i \ dy_i - \iint p(x_i, y_i) \ln p(y_i) \ dx_i \ dy_i \\ &= \sum_{a \in A} <z_{ai}>^k \iint p(x_i, y_i) \ln p_a(x_i, y_i) \ dx_i \ dy_i - \int p(y_i) \ln p(y_i) \ dy_i \\ &= \sum_{a \in A} <z_{ai}>^k \iint p(x_i, y_i) \ln p_a(x_i, y_i) \ dx_i \ dy_i + H(y_i) \end{aligned} \tag{C.3}$$

- **Case I:** $<z_{ai}> = 0.5, \forall a$ or $<z_{ai}> = \{0, 1\}$. For this case, it is easy to see that

$$\begin{aligned} \iint p(x_i, y_i) \ln p_a(x_i, y_i) \ dx_i \ dy_i &= \iint \left(\sum_{b \in A} <z_{bi}> \ p_b(x_i, y_i) \right) \ln p_a(x_i, y_i) \ dx_i \ dy_i \\ &= \iint p_a(x_i, y_i) \ln p_a(x_i, y_i) \ dx_i \ dy_i \end{aligned} \tag{C.4}$$

Thus, from equations (C.3) and (C.4), we have,

$$\begin{aligned}
E[Q_i(T,T^{(k-1)})] &= \sum_{a\in A} <z_{ai}>^k \int\int p(x_i,y_i)\ln p_a(x_i,y_i)\ dx_i\ dy_i + H(y_i) \\
&= \sum_{a\in A} <z_{ai}>^k \int\int p_a(x_i,y_i)\ln p_a(x_i,y_i)\ dx_i\ dy_i + H(y_i) \\
&= -\sum_{a\in A} <z_{ai}>^k\ H_a(x_i,y_i) + H(y_i) \\
&= -\left[\sum_{a\in A} <z_{ai}>^k\ H_a(x_i,y_i) - H(y_i)\right]
\end{aligned}$$

Thus, we have,

$$\begin{aligned}
E[Q(T,T^{(k-1)})] &= \sum_i E[Q_i(T,T^{(k-1)})] \\
&= -\sum_i \left[\sum_{a\in A} <z_{ai}>\ H_a(x_i,y_i) - H(y_i)\right] \\
&= -\sum_{a\in A}\sum_i <z_{ai}>\ H_a(x_i,y_i) + \sum_i H(y_i) \\
&= -\sum_{a\in A} H_a(x,y)\sum_i <z_{ai}> + \sum_i H(y_i) \\
&= -N\left[\sum_{a\in A} H_a(x,y)\sum_i \frac{<z_{ai}>}{N} - H(y)\right] \\
&= -N\left[\sum_{a\in A} <\overline{z_a}>\ H_a(x,y) - H(y)\right]
\end{aligned}$$

- **Case II:** For other values of $<z_{ai}>$.

$$\begin{aligned}
&\iint p(x_i,y_i)\ln p_a(x_i,y_i)\ dx_i\ dy_i \\
&= \iint \left(\sum_{b\in A} <z_{bi}>\ p_b(x_i,y_i)\right)\ln p_a(x_i,y_i)\ dx_i\ dy_i \\
&= <z_{1i}> \iint p_1(x_i,y_i)\ln p_a(x_i,y_i)\ dx_i\ dy_i + <z_{2i}> \iint p_2(x_i,y_i)\ln p_a(x_i,y_i)\ dx_i\ dy_i \\
&= \iint p_1(x_i,y_i)\ln p_a(x_i,y_i)^{<z_{1i}>}\ dx_i\ dy_i + \iint p_2(x_i,y_i)\ln p_a(x_i,y_i)^{<z_{2i}>}\ dx_i\ dy_i \qquad \text{(C.5)}
\end{aligned}$$

Thus, from equations (C.3) and (C.5), we have,

$$\begin{aligned} E[Q_i(T,T^{(k-1)})] &= <z_{1i}> \iint p_1(x_i,y_i) \ln \left(p_1(x_i,y_i)^{<z_{1i}>} p_2(x_i,y_i)^{<z_{2i}>}\right)\, dx_i\, dy_i \\ &\quad + <z_{2i}> \iint p_2(x_i,y_i) \ln \left(p_1(x_i,y_i)^{<z_{1i}>} p_2(x_i,y_i)^{<z_{2i}>}\right)\, dx_i\, dy_i \\ &\quad + H(y_i) \end{aligned} \tag{C.6}$$

Using Kullback–Leibler (KL) divergence, the equation (C.6) can be simplified to:

$$\begin{aligned} E[Q_i(T,T^{(k-1)})] &\leq <z_{1i}> \iint p_1(x_i,y_i) \ln p_1(x_i,y_i)\, dx_i\, dy_i \\ &\quad + <z_{2i}> \iint p_2(x_i,y_i) \ln p_2(x_i,y_i)\, dx_i\, dy_i + H(y_i) \end{aligned} \tag{C.7}$$

It is easy to see that the equality, in equation (C.7), holds for the values of $<z_{ai}>$ discussed in case I. Equation (C.7) can also be written as:

$$E[Q_i(T,T^{(k-1)})] \leq -<z_{1i}> H_1(x_i,y_i) - <z_{2i}> H_2(x_i,y_i) + H(y_i) \tag{C.8}$$

$$= -\sum_{a\in A} <z_{ai}> H_a(x_i,y_i) + H(y_i) \tag{C.9}$$

Thus, proceeding as for previous case, we have,

$$\begin{aligned} E[Q(T,T^{(k-1)})] &= \sum_i E[Q_i(T,T^{(k-1)})] \\ &\leq -N\left[\sum_{a\in A} <\overline{z_a}> H_a(x,y) - H(y)\right] \end{aligned}$$

Thus, instead of estimating the transformation parameters according to

$$\hat{T} = \arg\max_T Q(T,T^{(k-1)})$$

we propose to chose optimal transformation parameters as follows:

$$\hat{T} = \arg\max_T -N\left[\sum_{a\in A} <\overline{z_a}> H_a(x,y) - H(y)\right]$$

$$
\begin{aligned}
&= \arg\min_{T} N \left[\sum_{a \in A} <\overline{z_a}> H_a(x,y) - H(y) \right] \\
&= \arg\min_{T} \left[\sum_{a \in A} <\overline{z_a}> H_a(x,y) - H(y) \right] \qquad \text{(C.10)}
\end{aligned}
$$

We chose to optimize (C.10) for the following reasons.

1. Equation (C.10) has a nice interpretation of minimizing joint conditional entropy, $H(M, X|Y)$, for estimating transformation parameters. That is, starting with the joint conditional entropy $H(M, X|Y)$, one can easily derive the equation (C.10). However, a subtle point to note is that in equation (C.10), the expected values, $<z_{ai}>$ are evaluated with respect to conditional probabilities. However, note that when writing $H(M, X|Y)$, as in equation (5.4), the expected values are with respect to the true segmentation density functions.

2. Starting from the joint conditional entropy interpretation, additional information, for example edge information, can be systematically introduced in the problem, as will be done in our future work.

C.3 Deriving the Min Step from Joint Conditional Entropy

This section shows that the *min* step, as developed in equation (C.10), can also be derived by starting from the joint conditional entropy $H(M, X|Y)$.

$$
\begin{aligned}
T^k &= \arg\min_T H(M, X|Y) \\
&= \arg\min_T \ (H(M, X, Y) - H(Y)) \\
&= \arg\min_T \ (H(M) + H(X, Y|M) - H(Y)) && \text{(C.11)} \\
&= \arg\min_T \ (H(X, Y|M) - H(Y)) && \text{(C.12)} \\
&= \arg\min_T \ \left(\sum_i (H(x_i, y_i|m_i) - H(y_i))\right) && \text{(C.13)} \\
&= \arg\min_T \left(\sum_i \left(\sum_{a\in A} P_i(a) H(x_i, y_i|\mathbf{m}_i = a) - H(y_i)\right)\right) \\
&= \arg\min_T \left(\sum_i \left(\sum_{a\in A} P_i(a) H_a(x_i, y_i) - H(y_i)\right)\right) \\
&= \arg\min_T \left(\sum_i \left(\sum_{a\in A} P_i(a) H_a(x, y) - H(y)\right)\right) && \text{(C.14)} \\
&= \arg\min_T \left(N \left(\sum_{a\in A} H_a(x, y) \sum_i \{\frac{P_i(a)}{N}\} - H(y)\right)\right) \\
&= \arg\min_T \left(\sum_{a\in A} H_a(x, y) \sum_i \{\frac{<z_{ai}>^k}{N}\} - H(y)\right) \\
&= \arg\min_T \left(\sum_{a\in A} <\overline{z_a}>^k \ H_a(x, y) - H(y)\right)
\end{aligned}
$$

where $P(M)$ is the joint density function on the segmentation of the portal image and $<\overline{z_a}> \equiv \frac{<z_{ai}>^k}{N}$. Note that we assume all pixels to be statistically independent. Equation (C.11) follows from the consistency requirement of the entropy measure [58]. Equation (C.12) follows from the fact that in min step $P(M)$ is independent of T. Equa-

tion (C.13) follows from the assumption that the pixels are statistically independent.

C.4 Gradient Evaluation

This section evaluates derivatives of the various terms in the derivative of the energy function for stochastic gradient descent approach for estimating the transformation parameters. We define a column vector $w = (x, y)$ to simplify the notations in this section. From equation (5.6), we have,

$$\begin{aligned} H_a(x,y) &= H_a(w) \\ &\approx -\frac{1}{\sum_{w_j \in \mathbf{J}} P_j(a)} \sum_{w_j \in \mathbf{J}} P_j(a) \ \ln \frac{1}{\sum_{w_i \in \mathbf{I}} P_i(a)} \sum_{w_i \in \mathbf{I}} P_i(a)\, G_{\Psi_a}(w_j - w_i) \end{aligned}$$

where the covariance matrices of the Gaussian kernels in the Parzen window density estimates are assumed to be diagonal, that is, $\Psi_a \equiv DIAG(\psi_{axx}, \psi_{ayy})$ (However, note that this assumption does not mean that the joint density function can be written in a factored form). The Gaussian kernels are defined as:

$$G_\Psi(x) \equiv (2\pi)^{-\frac{n}{2}} |\Psi|^{\frac{-1}{2}} \exp(\frac{-1}{2} x^T \Psi^{-1} x)$$

Hence, derivative of $G_{\Psi_a}(w_j - w_i)$ w.r.t the transformation parameters is given as:

$$\begin{aligned} G_{\Psi_a}(w_j - w_i) &\equiv (2\pi)^{-1} |\Psi_a|^{\frac{-1}{2}} \exp(\frac{-1}{2}(w_j - w_i)^T \Psi_a^{-1}(w_j - w_i)) \\ \frac{d}{dT} G_{\Psi_a}(w_j - w_i) &= G_{\Psi_a}(w_j - w_i)(\frac{-1}{2}) \frac{d}{dT} \left[(w_j - w_i)^T \Psi_a^{-1}(w_j - w_i) \right] \\ &= G_{\Psi_a}(w_j - w_i)(\frac{-1}{2}) 2(y_j - y_i) \psi_{ayy}^{-1} \frac{d}{dT}(y_j - y_i) \\ &= -G_{\Psi_a}(w_j - w_i)(y_j - y_i) \psi_{ayy}^{-1} \frac{d}{dT}(y_j - y_i) \end{aligned}$$

Now the derivative of the joint entropy with respect to the transformation parameters can written as

$$
\begin{aligned}
\frac{d}{dT}H_a(x,y) &= \frac{d}{dT}H_a(w) \\
&= -\left(\frac{1}{\sum_{w_j\in\mathbf{J}} P_j(a)}\right)\sum_{w_j\in\mathbf{J}} P_j(a)\frac{\sum_{w_i\in\mathbf{I}} P_i(a)\frac{d}{dT}G_{\Psi_a}(w_j-w_i)}{\sum_{w_k\in\mathbf{I}} P_k(a)G_{\Psi_a}(w_j-w_k)} \\
&= \left(\frac{1}{\sum_{w_j\in\mathbf{J}} P_j(a)}\right)\sum_{w_j\in\mathbf{J}} P_j(a)\frac{\sum_{w_i\in\mathbf{I}} P_i(a)G_{\Psi_a}(w_j-w_i)(y_j-y_i)\psi_{ayy}^{-1}\frac{d}{dT}(y_j-y_i)}{\sum_{w_k\in\mathbf{I}} P_k(a)G_{\Psi_a}(w_j-w_k)} \\
&= \left(\frac{1}{\sum_{w_j\in\mathbf{J}} P_j(a)}\right)\sum_{w_j\in\mathbf{J}} P_j(a)\sum_{w_i\in\mathbf{I}}\left(\frac{P_i(a)G_{\Psi_a}(w_j-w_i)}{\sum_{w_k\in\mathbf{I}} P_k(a)G_{\Psi_a}(w_j-w_k)}\right)(y_j-y_i)\psi_{ayy}^{-1}\frac{d}{dT}(y_j-y_i) \\
&= \left(\frac{1}{\sum_{w_j\in\mathbf{J}} P_j(a)}\right)\sum_{w_j\in\mathbf{J}} P_j(a)\sum_{w_i\in\mathbf{I}} W_{\Psi_a}(w_j,w_i)(y_j-y_i)\psi_{ayy}^{-1}\frac{d}{dT}(y_j-y_i) \qquad \text{(C.15)}
\end{aligned}
$$

where, we have defined,

$$
W_{\Psi_a}(w_j,w_i) \equiv \left(\frac{P_i(a)G_{\Psi_a}(w_j-w_i)}{\sum_{w_k\in\mathbf{I}} P_k(a)G_{\Psi_a}(w_j-w_k)}\right)
$$

The marginal density is also estimated as a statistical mean as follows:

$$
H(y) \approx -\frac{1}{N_J}\sum_{y_j\in\mathbf{J}}\ln\frac{1}{N_I}\sum_{y_i\in\mathbf{I}} G_\Psi(y_j-y_i)
$$

Thus, the derivatives of the entropy, $H(y)$, of the marginal density w.r.t. T are evaluated as:

$$
\frac{d}{dT}H(y) = -\frac{1}{N_J}\sum_{y_j\in\mathbf{J}}\frac{\sum_{y_i\in I}\frac{d}{dT}G_\Psi(y_j-y_i)}{\sum_{y_k\in\mathbf{I}} G_\Psi(y_j-y_k)}
$$

$$
\begin{aligned}
&= \frac{1}{N_J} \sum_{y_j \in \mathbf{J}} \frac{\sum_{y_i \in I} G_\Psi(y_j - y_i)(y_j - y_i)\psi_y^{-1} \frac{d}{dT}(y_j - y_i)}{\sum_{y_k \in \mathbf{I}} G_\Psi(y_j - y_k)} \\
&= \frac{1}{N_J} \sum_{y_j \in \mathbf{J}} \sum_{y_i \in I} \left(\frac{G_\Psi(y_j - y_i)}{\sum_{y_k \in \mathbf{I}} G_\Psi(y_j - y_k)} \right) (y_j - y_i)\psi_y^{-1} \frac{d}{dT}(y_j - y_i) \\
&= \frac{1}{N_J} \sum_{y_j \in \mathbf{J}} \sum_{y_i \in I} W_\Psi(y_j, y_i)(y_j - y_i)\psi_y^{-1} \frac{d}{dT}(y_j - y_i) \qquad \text{(C.16)}
\end{aligned}
$$

where we have defined:

$$
W_\Psi(y_j, y_i) \equiv \left(\frac{G_\Psi(y_j - y_i)}{\sum_{y_k \in \mathbf{I}} G_\Psi(y_j - y_k)} \right)
$$

where y_i is the pixel intensity at the ith pixel in the 2D DRR, Y, obtained by summing the voxel intensities in the 3D CT data set, G, along a projection ray, $proj_i$, that is,

$$
y_i = \sum_{(r,s,t) \in proj_i} G(r, s, t)
$$

where $proj_i$ denotes the set of voxel coordinates in 3D CT data which lie along the projection ray from the radiation source to the ith pixel. It should be noted that x_i, y_i denote ith pixel intensities of the 2D portal and DRR respectively, whereas, r, s, t denote 3D CT voxel coordinates.

Thus, the derivative of y_i with respect a transformation parameter, say t_x, is evaluated as follows:

$$
\begin{aligned}
\frac{dy_i}{dt_x} &= \frac{d}{dt_x} \sum_{(r,s,t) \in proj_i} G(r, s, t) \\
&= \sum_{(r,s,t) \in proj_i} \frac{d}{dt_x} G(r, s, t)
\end{aligned}
$$

$$= \sum_{(r,s,t)\in proj_i} \left[\frac{\partial}{\partial r} G(r,s,t) \frac{dr}{dt_x} + \frac{\partial}{\partial s} G(r,s,t) \frac{ds}{dt_x} + \frac{\partial}{\partial t} G(r,s,t) \frac{dt}{dt_x} \right] \tag{C.17}$$

C.5 Coordinate Descent Interpretation of Minimax

In this section we provide a coordinate descent interpretation of the proposed *minimax* entropy strategy. This interpretation, based on the work of Neal *et al.* [79] highlights the iterative nature of the algorithm. Also, this interpretation helps develop the intuition as to why the proposed minimax entropy registration framework should converge to a local optimum.

Neal *et al.* [79] showed that the EM algorithm is equivalent to co–ordinate descent approach while optimizing the function $F(\tilde{P}, \theta)$ defined to be:

$$F(\tilde{P}, \theta) = E_{\tilde{P}}[\ln p(y, z|\theta)] + H(\tilde{P})$$

where θ is the set of parameters to be estimated, $\mathbf{y}$ denotes the missing data and $\mathbf{z}$ denotes the observed data. Note that $H(\tilde{P})$ represents the entropy of the distribution $\tilde{P}(y)$, which is estimated under the constraint $\int \tilde{P}(y)\, dy = 1$. Optimizing the function $F(\tilde{P}, \theta)$ using the co–ordinate descent strategy leads to the following two steps:

Step 1: Descend along the $\tilde{P}$ coordinates

$$\tilde{P}^k = \arg\max_{\tilde{P}} F(\tilde{P}, \theta^{k-1})$$

Step 2: Keep $\tilde{P}$ fixed at the values from the previous iteration and estimate θ.

$$\theta^k = \arg\max_{\theta} F(\tilde{P}^k, \theta)$$

It can be easily seen that the two steps as formulated above represent the E–step and the

M–step respectively of the EM algorithm. In more details, the two steps can be written as

E–Step: Descend along the $\tilde{P}$ coordinates

$$\tilde{P}^k = \arg\max_{\tilde{P}} \Big(E_{\tilde{P}}[\ln p(y, z|\theta)] + H(\tilde{P}) \Big)$$

under the constraint $\int \tilde{P}(y)\, dy = 1$. Thus, this step is estimating the maximum entropy distribution of the missing data.

M–Step: Keep $\tilde{P}$ fixed at the values from the previous iteration and estimate θ.

$$\theta^k = \arg\max_{\theta} \Big(E_{\tilde{P}^k}[\ln p(y, z|\theta)] \Big)$$

as the entropy $H(\tilde{P}^k)$ is independent of the parameters θ.

Note that relation to our minimax entropy strategy is already getting clear, which shows our strategy to be co–ordinate descent approach in estimation of the transformation parameters.

For estimating transformation parameters, $\mathbf{T}$, the corresponding energy function to be optimized can be written as:

$$F(\tilde{P}, T) = E_{\tilde{P}}[\ln p(M, X|Y)] + H(\tilde{P})$$

where T represents the transformation parameters being estimated. Note that Y depends on the current estimate of the transformation parameters. However, instead of optimizing $F(\tilde{P}, T)$, as defined above, we proposed to optimize the *expected* value of $F(\tilde{P}, T)$ (as the probability density function are being estimated from the data itself at the current

estimates of the transformation parameters).

Let us define an energy function $F_H(\tilde{P}, T)$ as:

$$F_H(\tilde{P}, T) = -H(M, X|Y) + H(M)$$

and the probability distribution and the transformation parameters are estimated by optimizing it in a coordinate descent method as follows. Define two steps to estimate the transformation parameters

Step 1:

$$\tilde{P}^k = \arg\max_{\tilde{P}} F_H(\tilde{P}, T^{k-1})$$

under the constraint $\int \tilde{P}^k(M)\, dM = 1$.

Step 2:

$$T^k = \arg\max_{T} F_H(\tilde{P}^k, T)$$

To see the relation between the above two steps and the minimax entropy registration framework, we write $F_H(\tilde{P}, T)$ in mode details as:

$$\begin{aligned} F_H(\tilde{P}, T) &= -H(M, X|Y) + H(M) \\ &= \iiint dM\ dX\ dY\ p(X, Y|M)\ \tilde{P}(M)\ \ln p(M, X|Y) - \int dM\ \tilde{P}(M)\ \ln \tilde{P}(M) \end{aligned}$$

with the constraint to be satisfied $\int \tilde{P}^k(M)\, dM = 1$. Using *Lagrange* multiplier technique

to solve for the desired density function, we expand the energy function as:

$$
\begin{aligned}
F'_H(\tilde{P},T) &= -H(M,X|Y) + H(M) - \lambda\left(\int dM\ \tilde{P}(M)\ \ln\tilde{P}(M) - 1\right) \\
&= \iiint dM\ dX\ dY\ p(X,Y|M)\ \tilde{P}(M)\ \ln p(M,X|Y) - \int dM\ \tilde{P}(M)\ \ln\tilde{P}(M) \\
&\quad -\lambda\left(\int dM\ \tilde{P}(M)\ \ln\tilde{P}(M) - 1\right) \\
&= \iint dX\ dY\ \int dM\ p(X,Y|M)\left[\tilde{P}(M)\ \ln p(M,X|Y) - \tilde{P}(M)\ \ln\tilde{P}(M)\right. \\
&\quad \left. -\lambda\left(\tilde{P}(M)\ \ln\tilde{P}(M) - 1\right)\right]
\end{aligned}
$$

Thus, differentiating the integrand in the inner most integral with respect to $\tilde{P}$, we get,

$$
\begin{aligned}
\frac{d}{d\tilde{P}}p(X,Y|M)\left[\tilde{P}(M)\ \ln p(M,X|Y) - \tilde{P}(M)\ \ln\tilde{P}(M) - \lambda\left(\tilde{P}(M)\ \ln\tilde{P}(M) - 1\right)\right] &= 0 \\
p(X,Y|M)\ \frac{d}{d\tilde{P}}\left[\tilde{P}(M)\ \ln p(M,X|Y) - \tilde{P}(M)\ \ln\tilde{P}(M) - \lambda\left(\tilde{P}(M)\ \ln\tilde{P}(M) - 1\right)\right] &= 0
\end{aligned}
$$

which implies, assuming that $p(X,Y|M) \neq 0$,

$$
\ln p(M,X|Y) - \ln\tilde{P}(M) - 1 - \lambda = 0
$$

i.e.,

$$
\begin{aligned}
\ln\tilde{P}(M) &= \ln p(M,X|Y) - 1 - \lambda \\
&= \ln\frac{p(M,X|Y)}{e\cdot e^{\lambda}} \\
\tilde{P}(M) &= \frac{p(M,X|Y)}{e\cdot e^{\lambda}}
\end{aligned}
$$

where λ is determined by the integratibility constraint on the estimated probability distribution, i.e.,

$$\begin{aligned}
\int dM\ \tilde{P}(M) &= 1 \\
\int dM\ \frac{p(M,X|Y)}{e \cdot e^{\lambda}} &= 1 \\
\frac{1}{e \cdot e^{\lambda}} \int dM\ p(M,X|Y) &= 1 \\
\frac{1}{e \cdot e^{\lambda}}\ p(X|Y) &= 1 \\
e \cdot e^{\lambda} &= p(X|Y)
\end{aligned}$$

Thus, the density function $\tilde{P}(M)$ is estimated to be:

$$\begin{aligned}
\tilde{P}(M) &= \frac{p(M,X|Y)}{e \cdot e^{\lambda}} \\
&= \frac{p(M|X,Y)\ p(X|Y)}{p(X|Y)} \\
&= p(M|X,Y)
\end{aligned}$$

Thus, in the first step of the minimax, the probability density function, $\tilde{P}(M)$ is being estimated, using the maximum entropy equation. Once the probability on the segmentation of the portal image has been estimated, it is plugged into the second step to estimate the transformation parameters.

Thus, the two steps in the minimax entropy can now be written as:

<u>Max Step:</u>

$$\tilde{P}^k = \arg\max_{\tilde{P}} \left[\iiint dM\ dX\ dY\ p(X,Y|M)\ \tilde{P}(M)\ \ln p(M,X|Y) - \int dM\ \tilde{P}(M)\ \ln \tilde{P}(M) \right]$$

under the constraint $\int dM\ \tilde{P}(M)\ \ln \tilde{P}(M) = 1$ and once the segmentation distribution has been estimated, the *min step* is

<u>Min Step:</u>

$$T^k = \arg\min_T - \iiint dM\ dX\ dY\ p(X,Y|M)\ \tilde{P}^k(M)\ \ln p(M,X|Y)$$

C.6 Convergence Analysis of Stochastic Approximation Algorithms

In this appendix we follow the analysis from [53]. Consider a stochastic approximation algorithm where the parameters $\mathbf{w}$ are being updated as:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \lambda(n)\, h(\mathbf{w}(n), \mathbf{x}(n)) \tag{C.18}$$

where n is the number of iterations, $\mathbf{w}(\cdot)$ is a sequence of vectors that are the object of interest and $\mathbf{x}(n)$ is an input vector received at time n using which the vector $\mathbf{w}(n)$ is updated. Note that in general the vectors $\mathbf{w}(n)$ and $\mathbf{x}(n)$ can be of different dimensions. The sequence $\lambda(n)$ is a sequence of positive numbers. The *update function* $h(\cdot,\cdot)$ is a deterministic function with some regularity conditions.

To study the convergence of the stochastic update rule, a deterministic nonlinear differential equation is associated with the stochastic nonlinear difference equation (C.18). The stability properties of the differential equation are then tied to the convergence properties of the stochastic update rule. This procedure was developed by [72] and [64]. In this procedure, it is assumed that the stochastic update rule given by the equation (C.18) satisfies the following conditions:

1. The $\lambda(n)$ is a decreasing sequence of positive real numbers which satisfy the conditions:

   (a) $$\sum_{n=1}^{\infty} \lambda(n) = \infty$$

   (b) $$\sum_{n=1}^{\infty} \lambda^p(n) < \infty \qquad \text{for } p > 1$$

(c) $\lambda(n) \to 0 \qquad n \to \infty$

2. The sequence of the parameter vectors $\mathbf{w}(n)$ is bounded with probability 1.

3. The update function $h(\mathbf{w}(n), \mathbf{x}(n))$ is continuously differentiable with respect to $\mathbf{w}$ and $\mathbf{x}$, and its derivatives are bounded.

4. The limit

$$\overline{h}(\mathbf{w}) = \lim_{n \to \infty} E[h(\mathbf{w}, \mathbf{x})] \tag{C.19}$$

exists for each $\mathbf{w}$; the statistical expectation operator E is over $\mathbf{x}$.

5. There is locally asymptotically stable (in the sense of Liapunov) solution to the ordinary differential equation

$$\frac{d}{dt}\mathbf{w}(t) = \overline{h}(\mathbf{w}(t)) \tag{C.20}$$

where t denotes a continuous variable.

6. Let, $\mathbf{q}_0$ denote the solution to equation (C.20) with the basin of attraction $\mathcal{B}(\mathbf{q}_\prime)$. Then the parameter vector $\mathbf{w}(n)$ enters the compact subset $\mathcal{A}$ of the basin of attraction $\mathcal{B}(\mathbf{q}_\prime)$ infinitely often, with probability 1.

The condition 1(a) ensures that the algorithm moves the estimate to the desired limit, regardless of the initial conditions. Condition 1(b) gives a condition how fast $\lambda(n)$ must tend to zero. However, in most of the applications, it is not feasible to require the condition 1(c). This difficulty by assigning some small positive value to λ, the size of which usually depends on the application of interest. For our application, the value of λ was set to 0.02 which worked very well for most of the image data set. However, it

will be most interesting to see the effect of decreasing λ with iterations on the estimated parameters.

Appendix D

Incorporating Correlation in Registration Framework

Chapter 7 assumes that the neighboring pixel intensities in an image are correlated. Such a correlation among the pixels is modeled using a 1D MRF. In this appendix we derive the various results used in developing the minimax entropy framework in chapter 7 while modeling and incorporating the correlation information in both the *min* step and the *max* step.

D.1 Maximum Entropy Estimation

This appendix develops the intermediate steps involved in the estimation of the optimal parameters $\varepsilon^*_{\alpha i}$. The development in the appendix starts from equation (7.11) in chapter 7, given below for completeness,

$$\frac{\partial}{\partial \varepsilon_{vi}} \left[F_I(\beta) + < E(\mathbf{Z}) - E_I(\mathbf{Z}) > \right] = 0 \tag{7.11}$$

Starting with equation (7.11), we derive the optimal estimates $\varepsilon^*_{\alpha i}$ to be:

$$\varepsilon^*_{\alpha i} = \left[\left\langle \frac{z_{\alpha i-1} + z_{\alpha i+1}}{2} + h_{\alpha i} - \frac{1}{\beta} \ \ln(l_i l_{i-1}) \right\rangle \right] \tag{7.12}$$

We first evaluate the three derivatives on the left side of equation (7.11) as follows:

D.1.1 Evaluation of the term $\frac{\partial}{\partial \varepsilon_{vi}} F_I(\beta)$

$$\begin{aligned} F_I(\beta) &= \frac{-1}{\beta} \ \ln Z_I(\beta) \\ &= \frac{-1}{\beta} \ \ln \left(\sum_{\mathbf{Z}} \exp\left[-\beta E_I(\mathbf{Z})\right] \right) \\ &= \frac{-1}{\beta} \ \ln \left(\sum_{\mathbf{Z}} \exp \left[\beta \sum_j \sum_{\alpha \in A} \varepsilon_{\alpha j} z_{\alpha j} \right] \right) \end{aligned}$$

Therefore,

$$\begin{aligned} \frac{\partial}{\partial \varepsilon_{vi}} F_I(\beta) &= \frac{-1}{\beta} \ \frac{\sum_{\mathbf{Z}} \exp\left[-\beta E_I(\mathbf{Z})\right] \ \beta \ z_{vi}}{\sum_{\mathbf{Z}'} \exp\left[-\beta E_I(\mathbf{Z})\right]} \\ &= -\frac{1}{Z_I(\beta)} \sum_{\mathbf{Z}} z_{vi} \ \exp\left[-\beta E_I(\mathbf{Z})\right] \\ &= -\sum_{\mathbf{Z}} z_{vi} \ P^{Gb}(E_I(\mathbf{Z})) \\ &= - < z_{vi} > \end{aligned}$$

D.1.2 Evaluation of the term $\frac{\partial}{\partial \varepsilon_{vi}} < E_I(\mathbf{Z}) >$

$$< E_I(\mathbf{Z}) > \ = \ \sum_{\mathbf{Z}} E_I(\mathbf{Z}) \ \ P^{Gb}(E_I(\mathbf{Z}))$$

$$
\begin{aligned}
&= \sum_{\mathbf{Z}} \left(-\sum_{j} \sum_{\alpha \in A} \varepsilon_{\alpha j} z_{\alpha j} \right) \; P^{Gb}(E_I(\mathbf{Z})) \\
&= -\sum_{j} \sum_{\alpha \in A} \varepsilon_{\alpha j} \left(\sum_{\mathbf{Z}} z_{\alpha j} \; P^{Gb}(E_I(\mathbf{Z})) \right) \\
&= -\sum_{j} \sum_{\alpha \in A} \varepsilon_{\alpha j} < z_{\alpha j} >
\end{aligned}
$$

Thus, taking the partial derivative of the above expression with respect to ε_{vi}, we get,

$$
\frac{\partial}{\partial \varepsilon_{vi}} < E_I(\mathbf{Z}) > \quad = \quad - < z_{vi} > - \sum_{j} \sum_{\alpha \in A} \varepsilon_{\alpha j} \; \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} >
$$

The second term on the right hand side can be simplified further.

$$
\begin{aligned}
\sum_{j} \sum_{\alpha \in A} \varepsilon_{\alpha j} \; \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} > \quad &= \quad \sum_{j} \sum_{\alpha \in A} \varepsilon_{\alpha j} \; \frac{\partial}{\partial \varepsilon_{vi}} \left(\sum_{\mathbf{Z}} z_{\alpha j} \; P^{Gb}(E_I(\mathbf{Z})) \right) \\
&= \quad \sum_{j} \sum_{\alpha \in A} \varepsilon_{\alpha j} \; \sum_{\mathbf{Z}} z_{\alpha j} \; \frac{\partial}{\partial \varepsilon_{vi}} P^{Gb}(E_I(\mathbf{Z}))
\end{aligned}
$$

Let us now consider the term $\frac{\partial}{\partial \varepsilon_{vi}} P^{Gb}(E_I(\mathbf{Z}))$.

$$
\begin{aligned}
\frac{\partial}{\partial \varepsilon_{vi}} P^{Gb}(E_I(\mathbf{Z})) \quad &= \quad \frac{\partial}{\partial \varepsilon_{vi}} \left(\frac{1}{Z_I(\beta)} \exp(-\beta \; E_I(\mathbf{Z})) \right) \\
&= \quad \frac{\partial}{\partial \varepsilon_{vi}} \left(\frac{1}{Z_I(\beta)} \exp(\beta \; \sum_{j} \sum_{\alpha \in A} \varepsilon_{\alpha j} \; z_{\alpha j}) \right) \\
&= \quad \left(\frac{1}{Z_I(\beta)} \right)^2 [Z_I(\beta) \; \beta z_{vi} \; \exp(-\beta E_I(\mathbf{Z})) \\
&\qquad - \exp(-\beta E_I(\mathbf{Z})) \; \sum_{\mathbf{Z}'} \beta z'_{vi} \; \exp(-\beta E_I(\mathbf{Z}')) \Big] \\
&= \quad \beta \left[z_{vi} \; P^{Gb}(E_I(\mathbf{Z})) - P^{Gb}(E_I(\mathbf{Z})) \sum_{\mathbf{Z}'} z'_{vi} \; P^{Gb}(E_I(\mathbf{Z}')) \right]
\end{aligned}
$$

$$
\begin{aligned}
&= \beta \left[z_{vi}\ P^{Gb}(E_I(\mathbf{Z})) - < z_{vi} >\ P^{Gb}(E_I(\mathbf{Z})) \right] \\
&= \beta \left[z_{vi} - < z_{vi} > \right]\ P^{Gb}(E_I(\mathbf{Z}))
\end{aligned}
$$

Thus,

$$
\begin{aligned}
\sum_j \sum_{\alpha \in A} \varepsilon_{\alpha j}\ \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} > &= \sum_j \sum_{\alpha \in A} \varepsilon_{\alpha j}\ \sum_{\mathbf{Z}} z_{\alpha j}\ \beta \left[z_{vi} - < z_{vi} > \right]\ P^{Gb}(E_I(\mathbf{Z})) \\
&= \beta \sum_j \sum_{\alpha \in A} \varepsilon_{\alpha j}\ \sum_{\mathbf{Z}} \left[z_{\alpha j} z_{vi} - z_{\alpha j} < z_{vi} > \right]\ P^{Gb}(E_I(\mathbf{Z})) \\
&= \beta \sum_j \sum_{\alpha \in A} \varepsilon_{\alpha j}\ \left[< z_{\alpha j} z_{vi} > - < z_{\alpha j} >< z_{vi} > \right]
\end{aligned}
$$

Note that if $j \neq i$, then $< z_{\alpha j} z_{vi} > \ = \ < z_{\alpha j} >< z_{vi} >$ (Note that this is true as the expected value is being taken with respect to the probability density function $P^{Gb}(E_I(\mathbf{Z}))$, which can be written in the factored form). Thus, we can reduce the above to

$$
\begin{aligned}
\sum_j \sum_{\alpha \in A} \varepsilon_{\alpha j}\ \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} > &= \beta \sum_{\alpha \in A} \varepsilon_{\alpha j}\ \left[< z_{\alpha i} z_{vi} > - < z_{\alpha i} >< z_{vi} > \right] \\
&= \sum_{\alpha \in A} \varepsilon_{\alpha j}\ \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} >
\end{aligned}
$$

Thus, we have,

$$
\begin{aligned}
\frac{\partial}{\partial \varepsilon_{vi}} < E_I(\mathbf{Z}) > &= - < z_{vi} > - \sum_j \sum_{\alpha \in A} \varepsilon_{\alpha j}\ \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} > \\
&= - < z_{vi} > - \sum_{\alpha \in A} \varepsilon_{\alpha j}\ \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} >
\end{aligned}
$$

Thus, we can summarize the results as:

$$\begin{aligned}
\frac{\partial}{\partial \varepsilon_{vi}} F_I(\beta) &= -< z_{vi} > \\
\frac{\partial}{\partial \varepsilon_{vi}} < E(\mathbf{Z}) > &= \frac{\partial}{\partial \varepsilon_{vi}} < E(\mathbf{Z}) > \\
\frac{\partial}{\partial \varepsilon_{vi}} < E_I(\mathbf{Z}) > &= -< z_{vi} > + \sum_j \sum_{\alpha \in A} \varepsilon_{\alpha j} \ \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} > \\
&= -< z_{vi} > + \sum_{\alpha \in A} \varepsilon_{\alpha i} \ \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} > \qquad \text{(D.1)}
\end{aligned}$$

Thus, from equations (7.11) and (D.1)

$$\frac{\partial}{\partial \varepsilon_{vi}} < E(\mathbf{Z}) > + \sum_{\alpha \in A} \varepsilon_{\alpha i} \ \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} >= 0$$

Let us define a variable l_i as follows:

$$l_i \equiv \left(\frac{1}{1 + \exp[-\beta(\frac{1}{2} \sum_{\alpha \in A} z_{\alpha i} \ z_{\alpha i+1} - \lambda)]} \right)$$

which can be shown to be the conditional probability that $\mathbf{e}_i = 0$ (See appendix D.4 for the proof).

Taking logarithm, we get,

$$\ln l_i = -\ln \left[1 + \exp[-\beta(\frac{1}{2} \sum_{\alpha \in A} z_{\alpha i} \ z_{\alpha i+1} - \lambda)] \right]$$

Thus, we can write $E(\mathbf{Z})$ as

$$E(\mathbf{Z}) = -\sum_j \left[\sum_{\alpha \in A} h_{\alpha j} z_{\alpha j} + \frac{1}{2} \sum_{\alpha \in A} z_{\alpha j} z_{\alpha j+1} - \frac{1}{\beta} \ln l_j \right]$$

Thus, the expected value of $E(\mathbf{Z})$ is evaluated as:

$$< E(\mathbf{Z}) > \;=\; -\sum_j \left[\sum_{\alpha \in A} h_{\alpha j} < z_{\alpha j} > + \frac{1}{2} \sum_{\alpha \in A} < z_{\alpha j} z_{\alpha j+1} > - \frac{1}{\beta} < \ln l_j > \right]$$

Thus, differentiating with respect to ε_{vi}, we get,

$$\begin{aligned}
&\frac{\partial}{\partial \varepsilon_{vi}} < E(\mathbf{Z}) > \\
&= -\sum_j \left[\sum_{\alpha \in A} h_{\alpha j} \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} > + \frac{1}{2} \sum_{\alpha \in A} \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} z_{\alpha j+1} > - \frac{1}{\beta} \frac{\partial}{\partial \varepsilon_{vi}} < \ln l_j > \right] \\
&= -\left[\sum_{\alpha \in A} h_{\alpha i} \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} > + \frac{1}{2} \sum_j \sum_{\alpha \in A} \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} z_{\alpha j+1} > - \frac{1}{\beta} \sum_j \frac{\partial}{\partial \varepsilon_{vi}} < \ln l_j > \right]
\end{aligned}$$

Thus, for estimating variables ε_{vi}, we need to solve the following set of equations

$$\sum_{\alpha \in A} (h_{\alpha i} - \varepsilon_{\alpha i}) \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} > + \frac{1}{2} \sum_j \sum_{\alpha \in A} \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} z_{\alpha j+1} > - \frac{1}{\beta} \sum_j \frac{\partial}{\partial \varepsilon_{vi}} < \ln l_j > = 0 \quad \text{(D.2)}$$

Let us now evaluate the partial derivatives in equation (D.2).

D.1.3 Evaluating the term $\sum_j \sum_{\alpha \in A} \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} z_{\alpha j+1} >$

$$\sum_j \sum_{\alpha \in A} \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} z_{\alpha j+1} > \;=\; \sum_j \sum_{\alpha \in A} \left[\frac{\partial}{\partial \varepsilon_{vi}} \sum_{\mathbf{Z}} z_{\alpha j} z_{\alpha j+1} \;\; P^{Gb}(E_I(\mathbf{Z})) \right]$$

$$
\begin{aligned}
&= \sum_j \sum_{\alpha \in A} \left[\sum_{\mathbf{Z}} z_{\alpha j} z_{\alpha j+1} \ \frac{\partial}{\partial \varepsilon_{vi}} P^{Gb}(E_I(\mathbf{Z})) \right] \\
&= \beta \sum_j \sum_{\alpha \in A} \left[\sum_{\mathbf{Z}} z_{\alpha j} z_{\alpha j+1} \ (z_{iv} - < z_{vi} >) \ P^{Gb}(E_I(\mathbf{Z})) \right] \\
&= \beta \sum_j \sum_{\alpha \in A} [< z_{\alpha j} z_{\alpha j+1} z_{vi} > \ - \ < z_{\alpha j} z_{\alpha j+1} > < z_{vi} >]
\end{aligned}
$$

If $j \neq i$ or $j \neq i-1$, then we have $< z_{\alpha j} z_{\alpha j+1} z_{vi} > = < z_{\alpha j} z_{\alpha j+1} > < z_{vi} >$. Thus, we can reduce the above expression to

$$
\begin{aligned}
&\sum_j \sum_{\alpha \in A} \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} z_{\alpha j+1} > \\
&\qquad = \beta \sum_{j=(i-1)}^{i} \sum_{\alpha \in A} [< z_{\alpha j} z_{\alpha j+1} z_{vi} > \ - \ < z_{\alpha j} z_{\alpha j+1} > < z_{vi} >] \\
&\qquad = \beta \sum_{\alpha \in A} [< z_{\alpha i-1} z_{\alpha i} z_{vi} > \ - \ < z_{\alpha i-1} z_{\alpha i} > < z_{vi} >] + \\
&\qquad\quad \beta \sum_{\alpha \in A} [< z_{\alpha i} z_{\alpha i+1} z_{vi} > \ - \ < z_{\alpha i} z_{\alpha i+1} > < z_{vi} >] \\
&\qquad = \beta \sum_{\alpha \in A} [< z_{\alpha i-1} > < z_{\alpha i} z_{vi} > \ - \ < z_{\alpha i-1} > < z_{\alpha i} > < z_{vi} >] + \\
&\qquad\quad \beta \sum_{\alpha \in A} [< z_{\alpha i+1} > < z_{\alpha i} z_{vi} > \ - \ < z_{\alpha i+1} > < z_{\alpha i} > < z_{vi} >] \\
&\qquad = \beta \sum_{\alpha \in A} < z_{\alpha i-1} > \ [< z_{\alpha i} z_{vi} > \ - \ < z_{\alpha i} > < z_{vi} >] + \\
&\qquad\quad \beta \sum_{\alpha \in A} < z_{\alpha i+1} > \ [< z_{\alpha i} z_{vi} > \ - \ < z_{\alpha i} > < z_{vi} >] \\
&\qquad = \beta \sum_{\alpha \in A} [< z_{\alpha i-1} > + < z_{\alpha i+1} >] \ [< z_{\alpha i} z_{vi} > \ - \ < z_{\alpha i} > < z_{vi} >] \\
&\qquad = \beta \sum_{\alpha \in A} [< z_{\alpha i-1} > + < z_{\alpha i+1} >] \ \frac{\partial}{\partial \varepsilon_{vi}} < z_{vi} >
\end{aligned}
$$

D.1.4 Evaluating the term $\frac{1}{\beta}\sum_j \frac{\partial}{\partial \varepsilon_{vi}} < \ln l_j >$

First consider the term $\frac{\partial}{\partial \varepsilon_{vi}} < \ln l_i >$.

$$\begin{aligned}
\frac{\partial}{\partial \varepsilon_{vi}} < \ln l_i > &= \frac{\partial}{\partial \varepsilon_{vi}} \sum_{\mathbf{Z}} \ln l_i \cdot P^{Gb}(E_I(\mathbf{Z})) \\
&= \sum_{\mathbf{Z}} \ln l_i \cdot \frac{\partial}{\partial \varepsilon_{vi}} P^{Gb}(E_I(\mathbf{Z})) \\
&= \ln l_i \sum_{\mathbf{Z}} \frac{\partial}{\partial \varepsilon_{vi}} P^{Gb}(E_I(\mathbf{Z}))
\end{aligned}$$

The last equation follows as l_i is independent of z_{vi} as the values from the previous iterations are used. Thus,

$$\begin{aligned}
\frac{\partial}{\partial \varepsilon_{vi}} < \ln l_i > &= \ln l_i \sum_{\mathbf{Z}} \left(\sum_{\alpha \in A} z_{\alpha i} \right) \frac{\partial}{\partial \varepsilon_{vi}} P^{Gb}(E_I(\mathbf{Z})) \\
&= \ln l_i \sum_{\mathbf{Z}} \left[\sum_{\alpha \in A} z_{\alpha i} \frac{\partial}{\partial \varepsilon_{vi}} P^{Gb}(E_I(\mathbf{Z})) \right] \\
&= \ln l_i \sum_{\alpha \in A} \left[\sum_{\mathbf{Z}} z_{\alpha i} \frac{\partial}{\partial \varepsilon_{vi}} P^{Gb}(E_I(\mathbf{Z})) \right] \\
&= \ln l_i \sum_{\alpha \in A} \frac{\partial}{\partial \varepsilon_{vi}} \left[\sum_{\mathbf{Z}} z_{\alpha i} \, P^{Gb}(E_I(\mathbf{Z})) \right] \\
&= \ln l_i \sum_{\alpha \in A} \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} > \\
&= \sum_{\alpha \in A} \ln l_i \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} >
\end{aligned}$$

Therefore,

$$\frac{\partial}{\partial \varepsilon_{vi}} < \ln l_{i-1} > \;=\; \sum_{\alpha \in A} \ln l_{i-1} \, \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} >$$

So,

$$\begin{aligned}
\frac{1}{\beta} \sum_{j} \frac{\partial}{\partial \varepsilon_{vi}} < \ln l_j > &= \frac{1}{\beta} \left[\frac{\partial}{\partial \varepsilon_{vi}} < \ln l_{i-1} > + \frac{\partial}{\partial \varepsilon_{vi}} < \ln l_i > \right] \\
&= \frac{1}{\beta} \sum_{\alpha \in A} [\ln l_{i-1} + \ln l_i] \, \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} > \\
&= \frac{1}{\beta} \sum_{\alpha \in A} \ln(l_{i-1} \, l_i) \, \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} >
\end{aligned}$$

Summarizing the results of the partial derivatives, we have,

$$\begin{aligned}
\sum_{j} \sum_{\alpha \in A} \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha j} z_{\alpha j+1} > &= \sum_{\alpha \in A} (< z_{\alpha i-1} > + < z_{\alpha i+1} >) \, \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} > \\
&= \sum_{\alpha \in A} < z_{\alpha i-1} + z_{\alpha i+1} > \; \frac{\partial}{\partial \varepsilon_{vi}} < z_{i\alpha} > \\
\frac{1}{\beta} \sum_{j} \frac{\partial}{\partial \varepsilon_{vi}} < \ln l_j > &= \frac{1}{\beta} \left[\frac{\partial}{\partial \varepsilon_{vi}} < \ln l_{i-1} > + \frac{\partial}{\partial \varepsilon_{vi}} < \ln l_i > \right] \\
&= -\frac{1}{\beta} \sum_{\alpha \in A} \left[\ln(l_i l_{i-1}) \; \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} > \right] \qquad \text{(D.3)}
\end{aligned}$$

Thus, from equations (D.2) and (D.3), for an estimate on the variables ε_{vi}, we need to solve the following set of equations

$$\sum_{\alpha \in A} \left[\varepsilon_{\alpha i} - \left\langle \frac{z_{\alpha i-1} + z_{\alpha i+1}}{2} + h_{\alpha i} \right\rangle + \frac{1}{\beta} \;\; \ln(l_i l_{i-1}) \right] \frac{\partial}{\partial \varepsilon_{vi}} < z_{\alpha i} >= 0$$

or

$$\sum_{\alpha\in A}\left[\varepsilon_{\alpha i}-\left(\left\langle\frac{z_{\alpha i-1}+z_{\alpha i+1}}{2}+h_{\alpha i}\right\rangle-\frac{1}{\beta}\ \ \ln(l_i l_{i-1})\right)\right]\frac{\partial}{\partial\varepsilon_{vi}}<z_{\alpha i}>=0$$

Define $\tilde{\varepsilon}_{\alpha i}$ to be:

$$\tilde{\varepsilon}_{\alpha i}\equiv\left[\frac{z_{\alpha i-1}+z_{\alpha i+1}}{2}+h_{\alpha i}-\frac{1}{\beta}\ \ \ln(l_i l_{i-1})\right]$$

Thus, the above equation reduces to

$$\sum_{\alpha\in A}[\varepsilon_{\alpha i}-<\tilde{\varepsilon}_{\alpha i}>]\ \ \frac{\partial}{\partial\varepsilon_{vi}}<z_{\alpha i}>=0$$

Since

$$\frac{\partial}{\partial\varepsilon_{vi}}<z_{\alpha i}>\ =\ \ \beta[<z_{vi}z_{\alpha i}>-<z_{vi}><z_{\alpha i}>]$$

we have

$$\begin{aligned}
& \textstyle\sum_{\alpha\in A}(\varepsilon_{\alpha i}-<\tilde{\varepsilon}_{\alpha i}>)\ \ \frac{\partial}{\partial\varepsilon_{vi}}<z_{\alpha i}> &=& \ 0\\
\Rightarrow\quad & \beta\textstyle\sum_{\alpha\in A}(\varepsilon_{\alpha i}-<\tilde{\varepsilon}_{\alpha i}>)(<z_{vi}z_{\alpha i}>-<z_{vi}><z_{\alpha i}>) &=& \ 0\\
\Rightarrow\quad & \textstyle\sum_{\alpha\in A}[<z_{vi}z_{\alpha i}>(\varepsilon_{\alpha i}-<\tilde{\varepsilon}_{\alpha i}>)-<z_{vi}><z_{\alpha i}>(\varepsilon_{\alpha i}-<\tilde{\varepsilon}_{\alpha i}>)] &=& \ 0\\
\Rightarrow\quad & \textstyle\sum_{\alpha\in A}<z_{vi}z_{\alpha i}>(\varepsilon_{\alpha i}-<\tilde{\varepsilon}_{\alpha i}>)-<z_{vi}>\sum_{\alpha\in A}<z_{\alpha i}>(\varepsilon_{\alpha i}-<\tilde{\varepsilon}_{\alpha i}>) &=& \ 0\\
\Rightarrow\quad & <z_{vi}>(\varepsilon_{vi}-<\tilde{\varepsilon}_{vi}>)-<z_{vi}>\textstyle\sum_{\alpha\in A}<z_{\alpha i}>(\varepsilon_{\alpha i}-<\tilde{\varepsilon}_{\alpha i}>) &=& \ 0\\
\Rightarrow\quad & <z_{vi}>[(\varepsilon_{vi}-<\tilde{\varepsilon}_{vi}>)-\textstyle\sum_{\alpha\in A}<z_{\alpha i}>(\varepsilon_{\alpha i}-<\tilde{\varepsilon}_{\alpha i}>)] &=& \ 0
\end{aligned}$$

These set of equations are satisfied only for the case when

$$\varepsilon^*_{vi} = < \tilde{\varepsilon}_{vi} > + c_i; \;\; \forall v \in A$$

where c_i are arbitrary constants. In particular, if $c_i = 0$; $\forall i$, we have

$$\varepsilon^*_{vi} = < \tilde{\varepsilon}_{vi} >$$

D.2 Evaluating Correlation Between Pixels

Correlation between random variables z_{vi} and $z_{\alpha j}$ is defined as the expected value $< z_{vi}z_{\alpha j} >_{E(\mathbf{Z})}$, with respect to the probability density function $P^{Gb}(E(\mathbf{Z}))$. The expected value is evaluated as follows:

$$\begin{aligned} < z_{vi}z_{\alpha j} >_{E(\mathbf{Z})} &= \sum_{\mathbf{Z}} z_{vi}z_{\alpha j}P^{Gb}(E(\mathbf{Z})) \\ &= \frac{\sum_{\mathbf{Z}} z_{vi}z_{\alpha j}\exp(-\beta E(\mathbf{Z}))}{\sum_{\mathbf{Z}'}\exp(-\beta E(\mathbf{Z}'))} \\ &= \frac{1}{\sum_{\mathbf{Z}'}\exp(-\beta E(\mathbf{Z}'))}\sum_{\mathbf{Z}}\exp(-\beta E(\mathbf{Z}))\frac{\sum_{\{Z_i,Z_j\}} z_{vi}z_{\alpha j}\exp(-\beta E(\mathbf{Z}))}{\sum_{\{Z_i,Z_j\}}\exp(-\beta E(\mathbf{Z}))} \end{aligned} \tag{D.4}$$

where $\{Z_i, Z_j\}$ denotes all possible combinations of the values of the ith and the jth row of the matrix $\mathbf{Z}$, that is, $\{(0,1,0,1),(0,1,1,0),(1,0,0,1),(1,0,1,0)\}$, if $i \neq j$.

Depending upon the values of i and j, we have the following four cases.

1. **Case I:** $i = j$, that is, i and j are the same pixel. Then the possible combinations of $\{Z_i, Z_i\}$ are $\{(0,1,0,1),(1,0,1,0)\}$. Thus, it is easy to see the following two cases:

   - $\alpha \neq v$: $< z_{vi}z_{\alpha j} >_{E(\mathbf{Z})} = 0$
   - $\alpha = v$: $< z_{vi}z_{\alpha j} >_{E(\mathbf{Z})} = < z_{vi} >_{E(\mathbf{Z})}$

2. **Case II:** $i \neq j+1$ or $j \neq i+1$. For this case write energy $E(\mathbf{Z})$ as:

$$\begin{aligned} E(\mathbf{Z}) &= -\sum_j \left[\frac{1}{2}\sum_{\alpha\in A} z_{\alpha j}z_{\alpha j+1} + \sum_{\alpha\in A} h_{\alpha j}z_{\alpha j} - \frac{1}{\beta}\ln l_j\right] \\ &= -\Bigg[E_t(\mathbf{Z}) + \frac{1}{2}\sum_{\alpha\in A} z_{\alpha i}z_{\alpha i+1} + \frac{1}{2}\sum_{\alpha\in A} z_{\alpha i}z_{\alpha i-1} + \sum_{\alpha\in A} h_{\alpha i}z_{\alpha i} - \frac{1}{\beta}\ln l_i - \frac{1}{\beta}\ln l_{i-1} + \end{aligned}$$

$$
\begin{aligned}
& \frac{1}{2}\sum_{\alpha\in A} z_{\alpha j} z_{\alpha j+1} + \frac{1}{2}\sum_{\alpha\in A} z_{\alpha j} z_{\alpha j-1} + \sum_{\alpha\in A} h_{\alpha j} z_{\alpha j} - \frac{1}{\beta}\ln l_j - \frac{1}{\beta}\ln l_{j-1}\Bigg] \\
= \; & -\Bigg[E_t(\mathbf{Z}) + \sum_{\alpha\in A} z_{\alpha i}\left(\frac{z_{\alpha i+1}+z_{\alpha i-1}}{2} + h_{\alpha i} - \frac{1}{\beta}\ln(l_i l_{i-1})\right) + \\
& \sum_{\alpha\in A} z_{\alpha j}\left(\frac{z_{\alpha j+1}+z_{\alpha j-1}}{2} + h_{\alpha j} - \frac{1}{\beta}\ln(l_j l_{j-1})\right)\Bigg] \\
= \; & -\left[E_t(\mathbf{Z}) + \sum_{\alpha\in A} z_{\alpha i}\tilde{\varepsilon}_{\alpha i} + \sum_{\alpha\in A} z_{\alpha j}\tilde{\varepsilon}_{\alpha j}\right]
\end{aligned}
$$

where the term $E_t(\mathbf{Z})$ clubs all the terms without either $z_{\alpha i}$ or $z_{\alpha j}$.

Thus, the third term in the equation (D.4) can be written as:

$$
\begin{aligned}
& \frac{\sum_{\{Z_i,Z_j\}} z_{vi} z_{\alpha j} \exp\left[-\beta E(\mathbf{Z})\right]}{\sum_{\{Z_i,Z_j\}} \exp\left[-\beta E(\mathbf{Z})\right]} \\
& \qquad = \frac{\sum_{\{Z_i,Z_j\}} z_{vi} z_{\alpha j} \exp\left[\beta\left(\sum_{\alpha\in A} z_{\alpha i}\tilde{\varepsilon}_{\alpha i} + \sum_{\alpha\in A} z_{\alpha j}\tilde{\varepsilon}_{\alpha j}\right)\right]}{\sum_{\{Z_i,Z_j\}} \exp\left[\beta\left(\sum_{\alpha\in A} z_{\alpha i}\tilde{\varepsilon}_{\alpha i} + \sum_{\alpha\in A} z_{\alpha j}\tilde{\varepsilon}_{\alpha j}\right)\right]} \\
& \qquad = \frac{\exp\left[\beta\left(\tilde{\varepsilon}_{vi} + \tilde{\varepsilon}_{\alpha j}\right)\right]}{\sum_{\{Z_i,Z_j\}} \exp\left[\beta\left(\sum_{\alpha\in A} z_{\alpha i}\tilde{\varepsilon}_{\alpha i} + \sum_{\alpha\in A} z_{\alpha j}\tilde{\varepsilon}_{\alpha j}\right)\right]}
\end{aligned}
$$

Therefore, we have,

$$
\begin{aligned}
< z_{vi} z_{\alpha j} >_{E(\mathbf{Z})} &= \left\langle \frac{\exp\left[\beta\left(\tilde{\varepsilon}_{vi} + \tilde{\varepsilon}_{\alpha j}\right)\right]}{\sum_{\{Z_i,Z_j\}} \exp\left[\beta\left(\sum_{\alpha\in A} z_{\alpha i}\tilde{\varepsilon}_{\alpha i} + \sum_{\alpha\in A} z_{\alpha j}\tilde{\varepsilon}_{\alpha j}\right)\right]} \right\rangle_{E(\mathbf{Z})} \\
&\approx \frac{\exp\left[\beta\left(< \tilde{\varepsilon}_{vi} > + < \tilde{\varepsilon}_{\alpha j} >\right)\right]}{\sum_{\{Z_i,Z_j\}} \exp\left[\beta\left(\sum_{\alpha\in A} z_{\alpha i} < \tilde{\varepsilon}_{\alpha i} > + \sum_{\alpha\in A} z_{\alpha j} < \tilde{\varepsilon}_{\alpha j} >\right)\right]} \\
&= \frac{\exp\left[\beta\left(< \tilde{\varepsilon}_{vi} > + < \tilde{\varepsilon}_{\alpha j} >\right)\right]}{\left(\sum_{\alpha\in A} \exp\left[\beta < \tilde{\varepsilon}_{\alpha i} >\right]\right)\left(\sum_{\alpha\in A} \exp\left[\beta < \tilde{\varepsilon}_{\alpha j} >\right]\right)} \\
&= \left(\frac{\exp\left(\beta\varepsilon^*_{vi}\right)}{\sum_{\alpha\in A} \exp\left(\beta\varepsilon^*_{i\alpha}\right)}\right)\left(\frac{\exp\left(\beta\varepsilon^*_{\alpha j}\right)}{\sum_{\alpha\in A} \exp\left(\beta\varepsilon^*_{\alpha j}\right)}\right) \\
&= < z_{vi} >_{E(\mathbf{Z})} \cdot < z_{\alpha j} >_{E(\mathbf{Z})}
\end{aligned}
$$

3. **Case III:** $i = j+1$. This is the most interesting case as we are evaluating correlation

between the neighboring pixels. Again, write the energy function $E(\mathbf{Z})$ as:

$$
\begin{aligned}
E(\mathbf{Z}) &= -\sum_j \left[\frac{1}{2}\sum_{\alpha\in A} z_{\alpha j} z_{\alpha j+1} + \sum_{\alpha\in A} h_{\alpha j} z_{\alpha j} - \frac{1}{\beta}\ln l_j\right] \\
&= -\Bigg[E_t(\mathbf{Z}) + \frac{1}{2}\sum_{\alpha\in A} z_{\alpha j} z_{\alpha j+1} + \frac{1}{2}\sum_{\alpha\in A} z_{\alpha j} z_{\alpha j-1} + \frac{1}{2}\sum_{\alpha\in A} z_{\alpha i} z_{\alpha i+1} \\
&\quad + \sum_{\alpha\in A} h_{\alpha j} z_{\alpha j} + \sum_{\alpha\in A} h_{\alpha i} z_{\alpha i} - \frac{1}{\beta}\ln l_j - \frac{1}{\beta}\ln l_i - \frac{1}{\beta}\ln l_{j-1}\Bigg] \\
&= -\Bigg[E_t(\mathbf{Z}) + \frac{1}{2}\sum_{\alpha\in A} z_{\alpha j} z_{\alpha i} + \sum_{\alpha\in A}\left(\frac{z_{\alpha j-1}}{2} + h_{\alpha j}\right) z_{\alpha j} + \sum_{\alpha\in A}\left(\frac{z_{\alpha i+1}}{2} + h_{\alpha i}\right) z_{\alpha i} \\
&\quad -\frac{1}{\beta}\ln(l_j l_i l_{j-1})\Bigg] \\
&= -\left[E_t'(\mathbf{Z}) + E_{ij}(\mathbf{Z})\right]
\end{aligned}
$$

where the term $E_t'(\mathbf{Z})$ clubs all the terms without either $z_{\alpha i}$ or $z_{\alpha j}$.

Thus, the third term in the equation (D.4) for this case can be written as:

$$
\begin{aligned}
\frac{\sum_{\{Z_i,Z_j\}} z_{vi} z_{\alpha j} \exp\left[-\beta E(\mathbf{Z})\right]}{\sum_{\{Z_i,Z_j\}} \exp\left[-\beta E(\mathbf{Z})\right]} &= \frac{\sum_{\{Z_i,Z_j\}} z_{vi} z_{\alpha j} \exp\left[\beta E_{ij}(\mathbf{Z})\right]}{\sum_{\{Z_i,Z_j\}} \exp\left[\beta E_{ij}(\mathbf{Z})\right]} \\
&= \frac{Numerator}{Deno_1 + Deno_2 + Deno_3 + Deno_4}
\end{aligned} \tag{D.5}
$$

The numerator can be written as

$$
\begin{aligned}
&Numerator \\
&= \exp\Bigg[\beta\Bigg\{\frac{1}{2}\delta(v-\alpha) + \left(\frac{z_{\alpha j-1}}{2} + h_{\alpha j}\right) + \left(\frac{z_{vi+1}}{2} + h_{vi}\right) \\
&\quad + \frac{1}{\beta}\ \ln\left[1 + \exp\left(-\beta\left(\frac{z_{vi+1}}{2} - \lambda\right)\right)\right] + \frac{1}{\beta}\ \ln\left[1 + \exp\left(-\beta\left(\frac{\delta(v-\alpha)}{2} - \lambda\right)\right)\right] \\
&\quad + \frac{1}{\beta}\ \ln\left[1 + \exp\left(-\beta\left(\frac{z_{\alpha j-1}}{2} - \lambda\right)\right)\right]\Bigg\}\Bigg]
\end{aligned}
$$

$Deno_1$, for the combination $\{Z_i, Z_j\} = \{(0,1,0,1)\}$, can be written as

$$
Deno_1 = \exp\left[\beta\left\{\frac{1}{2} + \left(\frac{z_{2\,j-1}}{2} + h_{2j}\right) + \left(\frac{z_{2\,i+1}}{2} + h_{2i}\right)\right.\right.
$$

$$+\frac{1}{\beta}\ \ln\left[1+\exp\left(-\beta\left(\frac{z_{2\ i+1}}{2}-\lambda\right)\right)\right]+\frac{1}{\beta}\ \ln\left[1+\exp\left(-\beta\left(\frac{1}{2}-\lambda\right)\right)\right]$$
$$+\frac{1}{\beta}\ \ln\left[1+\exp\left(-\beta\left(\frac{z_{2\ j-1}}{2}-\lambda\right)\right)\right]\Bigg\}\Bigg]$$

$Deno_2$, for the combination $\{Z_i, Z_j\} = \{(0,1,1,0)\}$, can be written as

$$\begin{aligned} Deno_2 &= \exp\Bigg[\beta\Bigg\{\left(\frac{z_{1\ j-1}}{2}+h_{1j}\right)+\left(\frac{z_{2\ i+1}}{2}+h_{2i}\right) \\ &\quad +\frac{1}{\beta}\ \ln\left[1+\exp\left(-\beta\left(\frac{z_{2\ i+1}}{2}-\lambda\right)\right)\right]+\frac{1}{\beta}\ \ln\left[1+\exp\left(\beta\lambda\right)\right] \\ &\quad +\frac{1}{\beta}\ \ln\left[1+\exp\left(-\beta\left(\frac{z_{1\ j-1}}{2}-\lambda\right)\right)\right]\Bigg\}\Bigg] \end{aligned}$$

$Deno_3$, for the combination $\{Z_i, Z_j\} = \{(1,0,0,1)\}$, can be written as

$$\begin{aligned} Deno_3 &= \exp\Bigg[\beta\Bigg\{\left(\frac{z_{2\ j-1}}{2}+h_{2j}\right)+\left(\frac{z_{1\ i+1}}{2}+h_{1i}\right) \\ &\quad +\frac{1}{\beta}\ \ln\left[1+\exp\left(-\beta\left(\frac{z_{1\ i+1}}{2}-\lambda\right)\right)\right]+\frac{1}{\beta}\ \ln\left[1+\exp\left(\beta\lambda\right)\right] \\ &\quad +\frac{1}{\beta}\ \ln\left[1+\exp\left(-\beta\left(\frac{z_{2\ j-1}}{2}-\lambda\right)\right)\right]\Bigg\}\Bigg] \end{aligned}$$

$Deno_4$, for the combination $\{Z_i, Z_j\} = \{(1,0,1,0)\}$, can be written as

$$\begin{aligned} Deno_4 &= \exp\Bigg[\beta\Bigg\{\frac{1}{2}+\left(\frac{z_{1\ j-1}}{2}+h_{1j}\right)+\left(\frac{z_{1\ i+1}}{2}+h_{1i}\right) \\ &\quad +\frac{1}{\beta}\ \ln\left[1+\exp\left(-\beta\left(\frac{z_{1\ i+1}}{2}-\lambda\right)\right)\right]+\frac{1}{\beta}\ \ln\left[1+\exp\left(-\beta\left(\frac{1}{2}-\lambda\right)\right)\right] \\ &\quad +\frac{1}{\beta}\ \ln\left[1+\exp\left(-\beta\left(\frac{z_{1\ j-1}}{2}-\lambda\right)\right)\right]\Bigg\}\Bigg] \end{aligned}$$

Let us define the following:

$$\begin{aligned} \alpha_1 &\equiv \ln\left[1+\exp\left(-\beta\left(\frac{z_{1\ i+1}}{2}-\lambda\right)\right)\right] \\ \alpha_2 &\equiv \ln\left[1+\exp\left(-\beta\left(\frac{z_{2\ i+1}}{2}-\lambda\right)\right)\right] \\ \alpha_3 &\equiv \ln\left[1+\exp\left(-\beta\left(\frac{z_{1\ j-1}}{2}-\lambda\right)\right)\right] \end{aligned}$$

$$\begin{aligned}
\alpha_4 &\equiv \ln\left[1+\exp\left(-\beta\left(\frac{z_{2\,j-1}}{2}-\lambda\right)\right)\right] \\
\alpha_5 &\equiv \ln\left[1+\exp\left(-\beta\left(\frac{1}{2}-\lambda\right)\right)\right] \\
\alpha_6 &\equiv \ln\left[1+\exp\left(\beta\lambda\right)\right]
\end{aligned}$$

Thus, we can write the four terms in the denominator of the equation (D.5) as:

$$\begin{aligned}
Deno_1 &\equiv \exp\left[\beta\left(0.5+\frac{z_{2\,j-1}+z_{2\,i+1}}{2}+(h_{2j}+h_{2i})\right)+\alpha_2+\alpha_4+\alpha_5\right] \\
Deno_2 &\equiv \exp\left[\beta\left(\frac{z_{1\,j-1}+z_{2\,i+1}}{2}+(h_{1j}+h_{2i})\right)+\alpha_2+\alpha_3+\alpha_6\right] \\
Deno_3 &\equiv \exp\left[\beta\left(\frac{z_{2\,j-1}+z_{1\,i+1}}{2}+(h_{2j}+h_{1i})\right)+\alpha_1+\alpha_4+\alpha_6\right] \\
Deno_4 &\equiv \exp\left[\beta\left(0.5+\frac{z_{1\,j-1}+z_{1\,i+1}}{2}+(h_{1j}+h_{1i})\right)+\alpha_1+\alpha_3+\alpha_5\right]
\end{aligned}$$

Thus, we can write the correlation values as,

$$\begin{aligned}
< z_{1i}z_{1j} > &= \left\langle\frac{Deno_4}{Deno_1+Deno_2+Deno_3+Deno_4}\right\rangle \\
< z_{2i}z_{2j} > &= \left\langle\frac{Deno_1}{Deno_1+Deno_2+Deno_3+Deno_4}\right\rangle
\end{aligned}$$

4. **Case IV:** $j = i + 1$. Same as the previous case.

D.3 Approximated Expected Value

In this section we show that the expected value of the random variable, z_{vi}, evaluated with respect to a density function which can be written in a factored form is equivalent to approximated expected value, $< z_{vi} >_{E(\mathbf{Z})}$ using mean field approximations and ignoring second and higher order derivatives. In chapter 7, the expected value is approximated to be:

$$< z_{vi} >_{E(\mathbf{Z})} \approx \frac{\exp\left(\beta < \tilde{\varepsilon}_{vi} >\right)}{\sum_{\alpha \in A} \exp\left(\beta < \tilde{\varepsilon}_{\alpha i} >\right)}$$

The approximation follows from the assumption that $\delta\tilde{\varepsilon}_{vi} = \tilde{\varepsilon}_{vi} - < \tilde{\varepsilon}_{vi} >$ is small. Note that expected value can be written as

$$\begin{aligned}
<z_{ai}> &= \langle f(< \tilde{\varepsilon}_{vi} > + \delta\tilde{\varepsilon}_{vi}) \rangle \\
&= \left\langle f(< \tilde{\varepsilon}_{vi} >) + \sum_v \frac{df}{d\tilde{\varepsilon}_{vi}}\, \delta\tilde{\varepsilon}_{vi} + \frac{1}{2} \sum_v \sum_\alpha \frac{d^2 f}{d\tilde{\varepsilon}_{vi}\, d\tilde{\varepsilon}_{\alpha i}} \delta\tilde{\varepsilon}_{vi}\, \delta\tilde{\varepsilon}_{\alpha i} + o(\delta\tilde{\varepsilon}_{vi}^3) \right\rangle \\
&= f(< \tilde{\varepsilon}_{vi} >) + \sum_v \frac{df}{d\tilde{\varepsilon}_{vi}} \; < \delta\tilde{\varepsilon}_{vi} > + \frac{1}{2} \sum_v \sum_\alpha \frac{d^2 f}{d\tilde{\varepsilon}_{vi}\, d\tilde{\varepsilon}_{\alpha i}} < \delta\tilde{\varepsilon}_{vi}\, \delta\tilde{\varepsilon}_{\alpha i} > + o(\delta < \tilde{\varepsilon}_{vi}^3 >) \\
&= f(< \tilde{\varepsilon}_{vi} >) + \frac{1}{2} \sum_v \sum_\alpha \frac{d^2 f}{d\tilde{\varepsilon}_{vi}\, d\tilde{\varepsilon}_{\alpha i}} < \delta\tilde{\varepsilon}_{vi}\, \delta\tilde{\varepsilon}_{\alpha i} > + o(\delta < \tilde{\varepsilon}_{vi}^3 >)
\end{aligned}$$

Ignoring the 2nd and higher order terms and defining the function $f(\tilde{\varepsilon}_{vi})$ to be:

$$f(\tilde{\varepsilon}_{vi}) = \frac{\exp\left(\beta\tilde{\varepsilon}_{vi}\right)}{\sum_{\alpha \in A} \exp\left(\beta\tilde{\varepsilon}_{\alpha i}\right)}$$

we get the approximated value of the expected value of z_{vi}.

D.4 Conditional Probability of the ith edge

The variable l_i is defined in the chapter 7 to be:

$$l_i \equiv \left(\frac{1}{1 + \exp\left[-\beta\left(\frac{1}{2}\sum_{\alpha \in A} z_{\alpha i}\, z_{\alpha i+1} - \lambda\right)\right]} \right)$$

In this section, we show that l_i is the conditional probability that there is no edge at the ith location, that is $\mathbf{e}_i = 0$, given the segmentation label information, that is, $P(\mathbf{e}_i = 0|\mathbf{Z})$. Note that the expression for $P^{Gb}(E(\mathbf{Z}, \mathcal{E}))$ is known. From this probability density it is easy to see that $\mathbf{e}_i$ and $\mathbf{e}_j$ are independent, for $i \neq j$. Thus, the density function $P(\mathcal{E})$ could be written in the factored form. The marginal density function $P(\mathcal{E})$ can be obtained from the joint density function as follows:

$$P(\mathcal{E}) \equiv \sum_{\mathbf{Z}} P^{Gb}(E(\mathbf{Z}, \mathcal{E}))$$

Note that since the segmentation labels are correlated, sum over all possible set of labels, $\{\mathbf{Z}\}$, is an unsurmountable problem. Instead we will evaluate the conditional probability which is analytically tractable.

We have

$$\begin{aligned} P^{Gb}(E(\mathbf{Z}, \mathcal{E})) &= \frac{1}{Z(\beta)} \exp\left[-\beta E(\mathbf{Z}, \mathcal{E})\right] \\ &= \frac{1}{Z(\beta)} \exp\left[-\beta(E_1(\mathbf{Z}) + E_2(\mathbf{Z}, \mathcal{E}))\right] \end{aligned}$$

Integrating out the edges, $P^{Gb}(E(\mathbf{Z}))$ is written as:

$$\begin{aligned} P^{Gb}(E(\mathbf{Z})) &= \sum_{\mathbf{E}} P^{Gb}(E(\mathbf{Z},\mathcal{E})) \\ &= \frac{1}{Z(\beta)} \exp\left[-\beta(E_1(\mathbf{Z}) + E_3(\mathbf{Z}) + E_4(\mathbf{Z}))\right] \end{aligned}$$

Thus, the conditional density function, $P(\mathcal{E}|Z)$, is given as

$$\begin{aligned} P(\mathcal{E}|\mathbf{Z}) &\equiv \left(\frac{P^{Gb}(E(\mathbf{Z},\mathcal{E}))}{P^{Gb}(E(\mathbf{Z}))}\right) \\ &= \frac{\exp\left[-\beta\ (E_1(\mathbf{Z}) + E_2(\mathbf{Z},\mathcal{E}))\right]}{\exp\left[-\beta\ (E_1(\mathbf{Z}) + E_3(\mathbf{Z}) + E_4(\mathbf{Z}))\right]} \\ &= \frac{\exp\left[-\beta\ E_2(\mathbf{Z},\mathcal{E})\right]}{\exp\left[-\beta\ (E_3(\mathbf{Z}) + E_4(\mathbf{Z}))\right]} \\ &= \frac{\prod_i \exp\left[\beta\ \left(\frac{1}{2}(\sum_{v\in A} z_{vi}\ z_{vi+1})(1-e_i) + \lambda e_i\right)\right]}{\prod_i \exp\left[\beta\ \left(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} + \frac{1}{\beta}\ \ln\left\{1 + \exp(-\beta(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda))\right\}\right)\right]} \\ &= \prod_i \left[\frac{\exp\left[\beta e_i\ \left(\lambda - \frac{1}{2}(\sum_{v\in A} z_{vi}\ z_{vi+1})\right)\right]}{\exp\left[\beta\ \left(\frac{1}{\beta}\ \ln\left\{1 + \exp(-\beta(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda))\right\}\right)\right]}\right] \\ &= \prod_i \left[\frac{\exp\left[-\beta e_i\ \left(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda\right)\right]}{1 + \exp\left[-\beta\left(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda\right)\right]}\right] \\ &= \prod_i P(e_i|\mathbf{Z}) \end{aligned}$$

where,

$$P(e_i|\mathbf{Z}) \equiv \left[\frac{\exp\left[-\beta e_i\ \left(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda\right)\right]}{1 + \exp\left[-\beta\left(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda\right)\right]}\right]$$

Thus, we have,

$$
\begin{aligned}
P(\mathbf{e}_i = 0|\mathbf{Z}) &= \left[\frac{1}{1+\exp\left[-\beta\left(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda\right)\right]}\right] \\
P(\mathbf{e}_i = 1|\mathbf{Z}) &= \left[\frac{\exp\left[-\beta\ \left(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda\right)\right]}{1+\exp\left[-\beta\left(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda\right)\right]}\right] \\
&= \left[\frac{\exp\left[\beta\ \lambda\right]}{\exp\left[\beta\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1}\right] + \exp\left[\beta\lambda\right]}\right] \\
&= \left[\frac{1}{1+\exp\left[\beta\left(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda\right)\right]}\right]
\end{aligned}
$$

Therefore, expected value of random variable $\mathbf{e}_i$, denoted by $< \mathbf{e}_i >$, can be evaluated as

$$
\begin{aligned}
< \mathbf{e}_i > &= \sum_{\mathcal{E}}\sum_{\mathbf{Z}} e_i\ P(E(\mathbf{Z},\mathcal{E})) \\
&= \sum_{\mathbf{e}_i\in\{0,1\}}\sum_{\mathbf{Z}} e_i\ P(\mathbf{Z},\mathbf{e}_i) \\
&= \sum_{\mathbf{e}_i\in\{0,1\}} e_i \sum_{\mathbf{Z}} P(e_i|\mathbf{Z})\ P(E(\mathbf{Z})) \\
&= \sum_{\mathbf{Z}} P(\mathbf{e}_i = 1|\mathbf{Z})\ P(E(\mathbf{Z})) \\
&= \sum_{\mathbf{Z}} \left[\frac{1}{1+\exp\left[\beta\left(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda\right)\right]}\right]\ P(E(\mathbf{Z})) \\
&= \left\langle \frac{1}{1+\exp\left[\beta\left(\frac{1}{2}\sum_{v\in A} z_{vi}\ z_{vi+1} - \lambda\right)\right]} \right\rangle \\
&\approx \frac{1}{1+\exp\left[\beta\left(\frac{1}{2}\sum_{v\in A} < z_{vi} > < z_{vi+1} > -\lambda\right)\right]}
\end{aligned}
$$

Images showing the edge distributions in chapter 8 plot the $< \mathbf{e}_i >$ value for each pixel in a portal image.

A simple analysis helps us to put bounds on the possible values of λ.

- Suppose that there is an edge in the location i. Then, we have, $\sum_{v \in A} z_{vi}\ z_{vi+1} = 0$.

  Thus, we get,

$$\begin{aligned} P(\mathbf{e}_i = 1|\mathbf{Z}) &= \frac{\exp(\beta\lambda)}{1 + \exp(\beta\lambda)} \\ &= \frac{1}{1 + \exp(-\beta\lambda)} \end{aligned}$$

  Thus, we see that for $\lambda > 0$, $P(\mathbf{e}_i = 1|\mathbf{Z}) \rightarrow 1$ as $\beta \rightarrow \infty$.

- Now suppose that there is no edge at the ith location, that is, $\sum_{v \in A} z_{vi}\ z_{vi+1} = 1$.

  Thus, we get,

$$P(\mathbf{e}_i = 0|\mathbf{Z}) = \frac{1}{1 + \exp[-\beta(0.5 - \lambda)]}$$

  Thus, we see that for $\lambda < 0.5$, $P(\mathbf{e}_i = 0|\mathbf{Z}) \rightarrow 1$ as $\beta \rightarrow \infty$.

Thus, for the consistency, we have, $0.0 < \lambda < 0.5$. In our experiments, we chose $\lambda = 0.25$.

D.5 Evaluation of various entropy terms

D.5.1 Evaluation of the term $H(X, Y|M)$

First we will write down the various probability terms used to estimate the various joint and conditional entropies.

$$
\begin{aligned}
p(X,Y,M) &= p(X,Y|M)P(M) \\
p(X,Y|M) &= p(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n | m_1, m_2, \ldots, m_n) \\
&= p(x_n, y_n | x_{n-1}, y_{n-1}, m_n, m_{n-1}) \cdots p(x_2, y_2 | x_1, y_1, m_2, m_1) \cdot p(x_1, y_1 | m_1) \\
&= \prod_{i=2}^{N} p(x_i, y_i | x_{i-1}, y_{i-1}, m_i, m_{i-1}) \cdot p(x_1, y_1 | m_1) \\
\ln p(X,Y|m) &= \sum_{i=2}^{N} \ln p(x_i, y_i | x_{i-1}, y_{i-1}, m_i, m_{i-1}) + \ln p(x_1, y_1 | m_1)
\end{aligned}
$$

Thus, the joint conditional entropy, $H(X, Y|M)$, can be written as:

$$
\begin{aligned}
&-H(X,Y|M) \\
&\quad = \sum_M \iint dX\ dY p(M,X,Y) \ln p(X,Y|M) \\
&\quad = \sum_M \iint dX\ dY p(M,X,Y) \left[\sum_{i=2}^{N} \ln p(x_i, y_i | x_{i-1}, y_{i-1}, m_i, m_{i-1}) + \ln p(x_1, y_1 | m_1) \right] \\
&\quad = \sum_M \left[\sum_{i=2}^{N} \iint dX\ dY p(M,X,Y) \ln p(x_i, y_i | x_{i-1}, y_{i-1}, m_i, m_{i-1}) + \right. \\
&\qquad \left. \iint dX\ dY p(M,X,Y) \ln p(x_1, y_1 | m_1) \right] \\
&\quad = \sum_M P(M) \left[\sum_{i=2}^{N} \iint dX\ dY p(X,Y|M) \ln p(x_i, y_i | x_{i-1}, y_{i-1}, m_i, m_{i-1}) \right. \\
&\qquad \left. + \iint dX\ dY p(X,Y|M) \ln p(x_1, y_1 | m_1) \right]
\end{aligned}
$$

$$= \sum_M P(M) \left[\sum_{i=2}^{N} \iiiint dx_i \; dx_{i-1} \; dy_i \; dy_{i-1} \right.$$

$$p(x_i, y_i, x_{i-1}, y_{i-1} | m_i, m_{i-1}) \ln p(x_i, y_i | x_{i-1}, y_{i-1}, m_i, m_{i-1})$$

$$\left. + \iint dx_1 \; dy_1 p(x_1, y_1 | m_1) \ln p(x_1, y_1 | m_1) \right]$$

$$= \sum_{i=2}^{N} \sum_{\{\alpha,\beta\}} P(\mathbf{m}_i = \alpha, \mathbf{m}_{i-1} = \beta) \iiiint dx_i \; dx_{i-1} \; dy_i \; dy_{i-1}$$

$$p(x_i, y_i, x_{i-1}, y_{i-1} | m_i, m_{i-1}) \ln p(x_i, y_i | x_{i-1}, y_{i-1}, m_i, m_{i-1})$$

$$+ \sum_{\alpha \in A} P(\mathbf{m}_1 = \alpha) \iint dx_1 \; dy_1 p(x_1, y_1 | m_1) \ln p(x_1, y_1 | m_1)$$

$$= -\sum_{i=2}^{N} \sum_{\{\alpha,\beta\}} P(\mathbf{m}_i = \alpha, \mathbf{m}_{i-1} = \beta) H_{\alpha\beta}(x_i, y_i | x_{i-1}, y_{i-1}) - \sum_{\alpha \in A} P(\mathbf{m}_1 = \alpha) H_\alpha(x_1, y_1)$$

Thus, we have

$$\begin{aligned} H(X, Y | M) &= \sum_{i=2}^{N} \sum_{\{\alpha,\beta\}} P(\mathbf{m}_i = \alpha, \mathbf{m}_{i-1} = \beta) H_{\alpha\beta}(x_i, y_i | x_{i-1}, y_{i-1}) \\ &\quad + \sum_{\alpha \in A} P(\mathbf{m}_1 = \alpha) H_\alpha(x_1, y_1) \\ &= \sum_{i=2}^{N} \sum_{\{\alpha,\beta\}} P(\mathbf{m}_i = \alpha, \mathbf{m}_{i-1} = \beta) H_{\alpha\beta}(x_i, y_i, x_{i-1}, y_{i-1}) \\ &\quad - \sum_{i=2}^{N} \sum_{\alpha \in A} P(\mathbf{m}_i = \alpha) H_\alpha(x_i, y_i) + \sum_{\alpha \in A} P(\mathbf{m}_1 = \alpha) H_\alpha(x_1, y_1) \end{aligned}$$

Assuming stationary random field on the distribution of the pixel intensities, for each class of pixel labels, we can reduce the above expression to

$$
\begin{aligned}
H(X,Y|M) &= \sum_{\{\alpha,\beta\}} H_{\alpha\beta}(x_2,y_2,x_1,y_1)\sum_{i=2}^{N} P(\mathbf{m}_i=\alpha,\mathbf{m}_{i-1}=\beta) \\
&\quad - \sum_{\alpha\in A} H_\alpha(x,y)\sum_{i=2}^{N} P(\mathbf{m}_i=\alpha) + \sum_{\alpha\in A} P(\mathbf{m}_1=\alpha)H_\alpha(x,y) \\
&= \sum_{\{\alpha,\beta\}} H_{\alpha\beta}(x_2,y_2,x_1,y_1)\sum_{i=2}^{N} P(\mathbf{m}_i=\alpha,\mathbf{m}_{i-1}=\beta) \\
&\quad - \sum_{\alpha\in A} H_\alpha(x,y)\left[\sum_{i=2}^{N} P(\mathbf{m}_i=\alpha) - P(\mathbf{m}_1=\alpha)\right] \\
&= \sum_{\{\alpha,\beta\}} H_{\alpha\beta}(x_2,y_2,x_1,y_1)\sum_{i=2}^{N} < z_{\alpha i} z_{\beta i-1} > \\
&\quad - \sum_{\alpha\in A} H_\alpha(x,y)\left[\sum_{i=2}^{N} <z_{\alpha i}> - < z_{\alpha 1} >\right]
\end{aligned}
$$

Thus,

$$
\begin{aligned}
H(X,Y|M) &= (N-1)\left[\sum_{\{\alpha,\beta\}} H_{\alpha\beta}(x_2,y_2,x_1,y_1)\frac{\sum_{i=2}^{N} < z_{\alpha i} z_{\beta i-1} >}{N-1}\right. \\
&\quad \left. - \sum_{\alpha\in A} H_\alpha(x,y)\frac{\sum_{i=2}^{N} <z_{\alpha i}> - < z_{\alpha 1} >}{N-1}\right] \\
&\approx (N-1)\left[\sum_{\{\alpha,\beta\}} H_{\alpha\beta}(x_2,y_2,x_1,y_1) < \overline{z_\alpha z_\beta} > - \sum_{\alpha\in A} H_\alpha(x,y) < \overline{z_\alpha} >\right]
\end{aligned}
$$

D.5.2 Evaluation of the term $H(Y)$

Now let us consider the second term, $H(Y)$, the entropy of the 2D DRR.

$$p(Y) = \prod_{i=2}^{N} p(y_i|y_{i-1}) \cdot p(y_1)$$

Thus, the entropy of the DRR is given as:

$$\begin{aligned}
-H(Y) &= \int dY p(Y) \ln p(Y) \\
&= \int dY p(Y) \left[\sum_{i=2}^{N} \ln p(y_i|y_{i-1}) + \ln p(y_1) \right] \\
&= \sum_{i=2}^{N} \int dy_i \; dy_{i-1} \;\; p(y_i, y_{i-1}) \ln p(y_i|y_{i-1}) + \int dy_1 \; p(y_1) \ln p(y_1)
\end{aligned}$$

Thus, we have

$$H(Y) = \sum_{i=2}^{N} H(y_i|y_{i-1}) + H(y_1)$$

To reduce computational complexity, and also since the entropy of the DRR will not change significantly, we shall simplify the above expression to:

$$\begin{aligned}
H(Y) &= \sum_{i=2}^{N} H(y_i|y_{i-1}) + H(y_1) \\
&\leq \sum_{i=1}^{N} H(y_i) \\
&= N \cdot H(y)
\end{aligned}$$

D.5.3 Evaluation of the term $H(\mathcal{E}, G_e, T)$

As noted above, we back-project rays from the estimated edges in the portal image to the radiation source through the treatment planning 3D CT image dataset. When the 2D portal image is aligned to the 3D CT data set these rays should be tangential to the surface of the pelvic bone. Thus, we minimize the distances from the projection lines to the pelvic bone surface in a approach similar to that proposed by Lavallee *et al.* [66]. Distance from projection ray to the bone surface is defined as the minimum distance along the ray to the pelvic surface. When the images are not registered, the back projection rays might cut through the pelvic bone surface. Thus, to handle this situation, we follow the signed distance approach proposed by Lavallee *et al.* [66]. Computationally, to estimate a minimum distance from the projection ray to the pelvic bone surface, we follow a different approach than the quad–tree approach in [66], both to improve the speed–up and the accuracy of the algorithm. The term G_e represents a signed distance transform of the 3D CT dataset. To help in formulation, let $S(T) \equiv S'$ represent a 2D image whose the ith pixel value corresponds to the minimum distance along the projection ray from the ith pixel to the pelvic bone surface. Thus, the entropy $H(\mathcal{E}, G_e, T)$ is evaluated as the joint entropy $H(\mathcal{E}, S')$, i.e.,

$$H(\mathcal{E}, G_e, T) \equiv H(\mathcal{E}, S')$$

To evaluate the entropy $H(\mathcal{E}, S')$, consider the joint density function $p(\mathcal{E}, S')$. In this formulation we will assume that each pixel is independent of the other.

$$p(\mathcal{E}, S') = \prod_{i=1}^{N} p(e_i, s'_i)$$

$$= \prod_{i=1}^{N} P(e_i) p(s'_i|e_i)$$

Taking logarithm, we have,

$$\ln p(\mathcal{E}, S') = \sum_{i=1}^{N} [\ln p(s'_i|e_i) + \ln P(e_i)]$$

Thus, the joint entropy $H(\mathcal{E}, S')$ is evaluated as:

$$\begin{aligned}
-H(\mathcal{E}, S') &= \sum_{\mathcal{E}} \int dS' \; p(\mathcal{E}, S') \ln p(\mathcal{E}, S') \\
&= \sum_{\mathcal{E}} \int dS' \; p(\mathcal{E}, S') \sum_{i=1}^{N} [\ln p(s'_i|e_i) + \ln P(e_i)] \\
&= \sum_{\mathcal{E}} \int dS' \; p(e, S') \sum_{i=1}^{N} \ln p(s'_i|e_i) + \sum_{\mathcal{E}} \int dS' \; p(e, S') \sum_{i=1}^{N} \ln P(e_i) \\
&= \sum_{i=1}^{N} \sum_{\mathcal{E}} \int dS' \; p(\mathcal{E}, S') \ln p(s'_i|e_i) + \sum_{i=1}^{N} \sum_{\mathcal{E}} \int dS' \; p(\mathcal{E}, S') \ln P(e_i) \\
&= \sum_{i=1}^{N} \sum_{\mathcal{E}} \int dS' \; p(\mathcal{E}, S') \ln p(s'_i|e_i) + \sum_{i=1}^{N} \sum_{e_i \in \{0,1\}} P(e_i) \ln P(e_i) \qquad \text{(D.6)}
\end{aligned}$$

The first term on the right hand side of equation (D.6) can be derived as follows:

$$\begin{aligned}
\sum_{i=1}^{N} \sum_{\mathcal{E}} \int dS' \; p(\mathcal{E}, S') \ln P(e_i) &= \sum_{i=1}^{N} \sum_{\mathcal{E}} \ln P(e_i) \int dS' \; p(\mathcal{E}, S') \\
&= \sum_{i=1}^{N} \sum_{\mathcal{E}} P(\mathcal{E}) \; \ln P(e_i) \\
&= \sum_{i=1}^{N} \sum_{\mathcal{E}} \left(\prod_{j} P(e_j) \right) \; \ln P(e_i)
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{i=1}^{N} \left[\sum_{\mathcal{E}} \left(\prod_{j \neq i} P(e_j) \right) \; P(e_i) \; \ln P(e_i) \right] \\
&= \sum_{i=1}^{N} \sum_{e_i \in \{0,1\}} P(e_i) \ln P(e_i)
\end{aligned}
$$

As the second term on the right hand side of the equation (D.6) remains constant while estimating the transformation parameters, it can be dropped. Thus, the simplified joint entropy $H(\mathcal{E}, S')$ can be written as:

$$
\begin{aligned}
-H(\mathcal{E}, S') &= \sum_{i=1}^{N} \sum_{\mathcal{E}} \int dS' \; p(\mathcal{E}, S') \ln p(s'_i | e_i) \\
&= \sum_{i=1}^{N} \sum_{e_i \in \{0,1\}} \int ds'_i \; p(e_i, s'_i) \; \ln p(s'_i | e_i) \\
&= \sum_{i=1}^{N} \sum_{e_i \in \{0,1\}} P(e_i) \int ds'_i \; p(s'_i | e_i) \; \ln p(s'_i | e_i) \\
&= \sum_{i=1}^{N} P(\mathbf{e}_i = 1) \int ds'_i \; p(s'_i | \mathbf{e}_i = 1) \; \ln p(s'_i | \mathbf{e}_i = 1) \\
&\quad + \sum_{i=1}^{N} P(\mathbf{e}_i = 0) \int ds'_i \; p(s'_i | \mathbf{e}_i = 0) \; \ln p(s'_i | \mathbf{e}_i = 0)
\end{aligned}
$$

Therefore,

$$
H(\mathcal{E}, S') = \sum_{i=1}^{N} P(\mathbf{e}_i = 1) H_{i1}(s') + \sum_{i=1}^{N} P(\mathbf{e}_i = 0) H_{i0}(s')
$$

Since most of the pixels in $\mathcal{E}$ will be not edge, the second term will remain constant. Thus, ignoring this term, we get,

$$
H(\mathcal{E}, S') = \sum_{i=1}^{N} P(\mathbf{e}_i = 1) H_{i1}(s')
$$

$$
\begin{aligned}
&= H_1(s') \sum_{i=1}^{N} P(\mathbf{e}_i = 1) \\
&= N \cdot H_1(s') \, \frac{\sum_{i=1}^{N} P(\mathbf{e}_i = 1)}{N} \\
&= N \cdot H_1(s') \, \frac{\sum_{i=1}^{N} < e_i >}{N} \\
&= N \cdot < \overline{e} > \; H_1(s')
\end{aligned}
$$

Note that $H_1(s')$ denotes the entropy of the distances from the edges to the 3D pelvic surface and have defined

$$
< \overline{e} > \equiv \frac{\sum_{i=1}^{N} < e_i >}{N}
$$

Note that the values $< e_i >$ are being estimated in the *min step*. Since we will be minimizing the entropy of the distances, we feel that this will lead to more robust strategy as compared to minimization of mean squared distances, especially in the presence of the outliers.

D.6 Parzen Window Stochastic Estimates of the Entropies

In this section, we formulate the Parzen window based estimates of the various probability density functions which are used in stochastic estimates of the entropy terms. Gaussian kernels are used in the Parzen window method. The Gaussian kernels are further assumed to be radially symmetric, that is we assume that the covariance matrices are diagonal. One of the main advantage of the Parzen window estimate with Gaussian kernels is that the derivatives of the densities with respect to the transformation parameters can be analytically evaluated.

D.6.1 Evaluating $H_\alpha(x, y)$

In this section we evaluate the entropy $H_\alpha(x, y)$ and its derivatives w.r.t. the transformation parameters. Derivatives w.r.t. to the covariance matrix is also evaluated.

The joint entropy is defined to be:

$$H_\alpha(x, y) = -\int\int dx\ dy\ p_\alpha(x, y) \ln p_\alpha(x, y)$$

and the Parzen window estimate of the density function and the stochastic approximation of the joint entropy can be written as:

$$\begin{aligned} p_\alpha(x, y) &\equiv p_\alpha(w) \\ &\approx \frac{1}{\sum_{w_i \in \mathbf{I}} P_i(\alpha)} \sum_{w_i \in \mathbf{I}} P_i(\alpha) G_{\Psi_\alpha}(w - w_i) \\ H_\alpha(x, y) &\approx \left(\frac{-1}{\sum_{w_j \in \mathbf{J}} P_j(\alpha)} \right) \sum_{w_j \in \mathbf{J}} [P_j(\alpha)\ \ln (p_\alpha(w_j))] \end{aligned}$$

$$= \left(\frac{-1}{\sum_{w_j \in \mathbf{J}} P_j(\alpha)}\right) \sum_{w_j \in \mathbf{J}} \left[P_j(\alpha) \ \ln \left(\frac{1}{\sum_{w_i \in \mathbf{I}} P_i(\alpha)} \sum_{w_i \in \mathbf{I}} P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i) \right) \right]$$

where we have defined $w \equiv (x, y)^T$ to simplify the notation.

D.6.1.1 Evaluation of the derivative

Let, $\Psi_\alpha = \text{Diag}(\psi_{\alpha xx}, \psi_{\alpha yy})$, be a diagonal covariance matrix. Then we can write the Gaussian kernel as

$$G_{\Psi_\alpha}(w) \quad = \quad (2\pi)^{-1} \ |\Psi_\alpha|^{-1/2} \ \exp\left(\frac{-1}{2} w^T \ \Psi_\alpha^{-1} \ w\right)$$

Thus, the derivative of the joint entropy $H_\alpha(x, y)$ with respect to the transformation parameters can now be written as:

$$\begin{aligned}
&\frac{d}{dT} H_\alpha(x, y) \\
&= \left(\frac{-1}{\sum_{w_j \in \mathbf{J}} P_j(\alpha)}\right) \sum_{w_j \in \mathbf{J}} \left[P_j(\alpha) \ \frac{\sum_{w_i \in \mathbf{I}} P_i(\alpha) \frac{d}{dT} G_{\Psi_\alpha}(w_j - w_i)}{\sum_{w_i \in \mathbf{I}} P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)} \right] \\
&= \left(\frac{1}{\sum_{w_j \in \mathbf{J}} P_j(\alpha)}\right) \sum_{w_j \in \mathbf{J}} \left[P_j(\alpha) \ \frac{\sum_{w_i \in \mathbf{I}} P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)(y_j - y_i) \ \psi_{\alpha yy}^{-1} \ \frac{d}{dT}(y_j - y_i)}{\sum_{w_i \in \mathbf{I}} P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)} \right] \\
&= \left(\frac{1}{\sum_{w_j \in \mathbf{J}} P_j(\alpha)}\right) \sum_{w_j \in \mathbf{J}} \left[P_j(\alpha) \ \sum_{w_i \in \mathbf{I}} \frac{P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)}{\sum_{w_i \in \mathbf{I}} P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)} (y_j - y_i) \ \psi_{\alpha yy}^{-1} \ \frac{d}{dT}(y_j - y_i) \right] \\
&= \left(\frac{1}{\sum_{w_j \in \mathbf{J}} P_j(\alpha)}\right) \sum_{w_j \in \mathbf{J}} \left[P_j(\alpha) \ \sum_{w_i \in \mathbf{I}} W_{\Psi_\alpha}(w_j - w_i) \ \ (y_j - y_i) \ \psi_{\alpha yy}^{-1} \ \frac{d}{dT}(y_j - y_i) \right]
\end{aligned}$$

where we have defined:

$$W_{\Psi_\alpha}(w_j - w_i) \equiv \frac{P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)}{\sum_{w_i \in \mathbf{I}} P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)}$$

D.6.1.2 Evaluation of the covariance matrix parameters

Define $\sigma_{\alpha xx} \equiv \sqrt{\psi_{\alpha xx}}$ and $\sigma_{\alpha yy} \equiv \sqrt{\psi_{\alpha y}}$.

$$\begin{aligned}
G_{\Psi_\alpha}(w) &= (2\pi)^{-1} \, |\Psi_\alpha|^{-1/2} \, \exp\left(\frac{-1}{2} w^T \, \Psi_\alpha^{-1} \, w\right) \\
&= G_{\psi_{\alpha xx}}(x) \;\; G_{\psi_{\alpha yy}}(y) \\
\frac{d}{d\sigma_{\alpha yy}} G_{\Psi_\alpha}(w) &= G_{\psi_{\alpha xx}}(x) \;\; G_{\psi_{\alpha yy}}(y) \, \frac{1}{\sigma_{\alpha yy}} \left[\frac{y^2}{\sigma_{\alpha yy}^2} - 1\right] \\
&= G_{\Psi_\alpha}(w) \, \frac{1}{\sigma_{\alpha yy}} \left[\frac{y^2}{\sigma_{\alpha yy}^2} - 1\right] \\
\frac{d}{d\sigma_{\alpha xx}} G_{\Psi_\alpha}(w) &= G_{\Psi_\alpha}(w) \, \frac{1}{\sigma_{\alpha xx}} \left[\frac{x^2}{\sigma_{\alpha xx}^2} - 1\right]
\end{aligned}$$

Thus, the derivative of $H_\alpha(x, y)$ w.r.t. $\sigma_{\alpha xx}$ is evaluated as:

$$\begin{aligned}
&\frac{dH_\alpha(x,y)}{d\sigma_{\alpha xx}} \\
&= \left(\frac{-1}{\sum_{w_j \in \mathbf{J}} P_j(\alpha)}\right) \sum_{w_j \in \mathbf{J}} \left[P_j(\alpha) \, \frac{\sum_{w_i \in \mathbf{I}} P_i(\alpha) \frac{d}{d\sigma_{\alpha xx}} G_{\Psi_\alpha}(w_j - w_i)}{\sum_{w_i \in \mathbf{I}} P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)}\right] \\
&= \left(\frac{1}{\sum_{w_j \in \mathbf{J}} P_j(\alpha)}\right) \sum_{w_j \in \mathbf{J}} \left[P_j(\alpha) \, \sum_{w_i \in \mathbf{I}} \frac{P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)}{\sum_{w_i \in \mathbf{I}} P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)} \, \frac{1}{\sigma_{\alpha xx}} \, \left(\frac{(x_j - x_i)^2}{\sigma_{\alpha xx}^2} - 1\right)\right] \\
&= \left(\frac{1}{\sum_{w_j \in \mathbf{J}} P_j(\alpha)}\right) \sum_{w_j \in \mathbf{J}} \left[P_j(\alpha) \, \sum_{w_i \in \mathbf{I}} W_{\Psi_\alpha}(w_j - w_i) \;\; \frac{1}{\sigma_{\alpha xx}} \, \left(\frac{(x_j - x_i)^2}{\sigma_{\alpha xx}^2} - 1\right)\right]
\end{aligned}$$

Similarly,

$$
\begin{aligned}
&\frac{dH_\alpha(x,y)}{d\sigma_{\alpha yy}} \\
&= \left(\frac{-1}{\sum_{w_j\in\mathbf{J}} P_j(\alpha)}\right) \sum_{w_j\in\mathbf{J}} \left[P_j(\alpha) \, \frac{\sum_{w_i\in\mathbf{I}} P_i(\alpha) \frac{d}{d\sigma_{\alpha yy}} G_{\Psi_\alpha}(w_j - w_i)}{\sum_{w_i\in\mathbf{I}} P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)} \right] \\
&= \left(\frac{1}{\sum_{w_j\in\mathbf{J}} P_j(\alpha)}\right) \sum_{w_j\in\mathbf{J}} \left[P_j(\alpha) \, \sum_{w_i\in\mathbf{I}} \frac{P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)}{\sum_{w_i\in\mathbf{I}} P_i(\alpha) G_{\Psi_\alpha}(w_j - w_i)} \, \frac{1}{\sigma_{\alpha yy}} \, \left(\frac{(y_j - y_i)^2}{\sigma^2_{\alpha yy}} - 1 \right) \right] \\
&= \left(\frac{1}{\sum_{w_j\in\mathbf{J}} P_j(\alpha)}\right) \sum_{w_j\in\mathbf{J}} \left[P_j(\alpha) \, \sum_{w_i\in\mathbf{I}} W_{\Psi_\alpha}(w_j - w_i) \, \frac{1}{\sigma_{\alpha yy}} \, \left(\frac{(y_j - y_i)^2}{\sigma^2_{\alpha yy}} - 1 \right) \right]
\end{aligned}
$$

The derivatives are used in a stochastic gradient descent based strategy to estimate the optimal $\sigma_{\alpha yy}$ and $\sigma_{\alpha xx}$.

D.6.2 Evaluating $H(y)$

This section evaluates the entropy of a random variable which is distributed according to the pixel intensities in the DRR obtained by rendering the 3D CT image data at known transformation parameters. The Parzen window estimate of the density function and the stochastic estimate of the entropy are evaluated as follows:

$$
\begin{aligned}
H(y) &= -\int dy \, p(y) \, \ln p(y) \\
p(y) &\approx \frac{1}{N_I} \sum_{y_i\in\mathbf{I}} G_\Psi(y - y_i) \\
H(y) &\approx \frac{-1}{N_J} \sum_{y_j\in\mathbf{J}} \ln p(y_j) \\
&= \frac{-1}{N_J} \sum_{y_j\in\mathbf{J}} \ln \left(\frac{1}{N_I} \sum_{y_i\in\mathbf{I}} G_\Psi(y_j - y_i) \right)
\end{aligned}
$$

D.6.2.1 Evaluation of the derivatives

The Gaussian kernel in estimating the marginal entropy, $H(y)$, is given as:

$$G_\Psi(y) \quad = \quad (2\pi)^{-1/2}\ |\psi_y|^{-1/2}\ \exp\left(\frac{-1}{2} y\ \psi_y^{-1}\ y\right)$$

Thus, the derivative of the joint entropy $H(y)$ with respect to the transformation parameters can now be written as:

$$\begin{aligned}
\frac{d}{dT}H(y) &= \left(\frac{-1}{N_J}\right)\sum_{y_j\in\mathbf{J}}\left[\frac{\sum_{y_i\in\mathbf{I}}\frac{d}{dT}G_\Psi(y_j-y_i)}{\sum_{y_i\in\mathbf{I}}G_\Psi(y_j-y_i)}\right] \\
&= \left(\frac{1}{N_J}\right)\sum_{y_j\in\mathbf{J}}\left[\frac{\sum_{y_i\in\mathbf{I}}G_\Psi(y_j-y_i)(y_j-y_i)\psi_y^{-1}\frac{d}{dT}(y_j-y_i)}{\sum_{y_i\in\mathbf{I}}G_\Psi(y_j-y_i)}\right] \\
&= \left(\frac{1}{N_J}\right)\sum_{y_j\in\mathbf{J}}\left[\sum_{y_i\in\mathbf{I}}\left(\frac{G_\Psi(y_j-y_i)}{\sum_{y_i\in\mathbf{I}}G_\Psi(y_j-y_i)}\right)(y_j-y_i)\psi_y^{-1}\frac{d}{dT}(y_j-y_i)\right] \\
&= \left(\frac{1}{N_J}\right)\sum_{y_j\in\mathbf{J}}\left[\sum_{y_i\in\mathbf{I}}W_\Psi(y_j,y_i)\ (y_j-y_i)\psi_y^{-1}\frac{d}{dT}(y_j-y_i)\right]
\end{aligned}$$

where,

$$W_\Psi(y_j,y_i) \equiv \left(\frac{G_\Psi(y_j-y_i)}{\sum_{y_k\in\mathbf{I}}G_\Psi(y_j-y_k)}\right)$$

D.6.2.2 Evaluation of the variance ψ_y

Here we evaluate the derivative of the entropy $H(y)$ w.r.t. ψ_y, where the derivatives are used in the optimization strategy to estimate ψ_y. Let, $\sigma_y \equiv \sqrt{\psi_y}$.

$$
\begin{aligned}
H(y) &= \frac{-1}{N_J} \sum_{y_j \in \mathbf{J}} \ln \left(\frac{1}{N_I} \sum_{y_i \in \mathbf{I}} G_\Psi(y_j - y_i) \right) \\
G_\Psi(y) &= (2\pi)^{-1/2} \, |\psi_y|^{-1/2} \, \exp\left(\frac{-1}{2} y \, \psi_y^{-1} \, y \right) \\
\frac{d}{d\sigma_y} G_\Psi(y) &= (2\pi)^{-1/2} \frac{d}{d\sigma_y} \left[\sigma_y^{-1} \exp\left(\frac{-1}{2} y \, \sigma_y^{-2} \, y \right) \right] \\
&= (2\pi)^{-1/2} \left[-\sigma_y^{-2} \exp\left(\frac{-1}{2} y \, \sigma_y^{-2} \, y \right) + \sigma_y^{-1} \exp\left(\frac{-1}{2} y \, \sigma_y^{-2} \, y \right) \frac{y^2}{\sigma_y^3} \right] \\
&= G_\Psi(y) \left[\frac{y^2}{\sigma_y^3} - \frac{1}{\sigma_y} \right] \\
&= \frac{1}{\sigma_y} G_\Psi(y) \left[\frac{y^2}{\sigma_y^2} - 1 \right]
\end{aligned}
$$

Thus, we have,

$$
\begin{aligned}
\frac{d}{d\sigma_y} H(y) &= \frac{-1}{N_J} \sum_{y_j \in \mathbf{J}} \frac{\sum_{y_i \in \mathbf{I}} \frac{d}{d\sigma_y} G_\Psi(y_j - y_i)}{\sum_{y_i \in \mathbf{I}} G_\Psi(y_j - y_i)} \\
&= \frac{-1}{N_J} \sum_{y_j \in \mathbf{J}} \sum_{y_i \in \mathbf{I}} \left[\frac{1}{\sigma_y} \frac{G_\Psi(y_j - y_i)}{\sum_{y_i \in \mathbf{I}} G_\Psi(y_j - y_i)} \left(\frac{(y_j - y_i)^2}{\sigma_y^2} - 1 \right) \right] \\
&= \frac{-1}{N_J} \sum_{y_j \in \mathbf{J}} \sum_{y_i \in \mathbf{I}} \left[\frac{1}{\sigma_y} W_\Psi(y_j, y_i) \left(\frac{(y_j - y_i)^2}{\sigma_y^2} - 1 \right) \right]
\end{aligned}
$$

D.6.3 Evaluating $H_1(s')$

$H_1(s')$ denotes the entropy of the signed distances for the edges detected in the portal images. Note that these signed distances are evaluated in the following manner. First,

the 3D CT image data is signed distance transformed. Then, the projection rays from the detected edges are back–projected, through the signed distance transformed 3D CT image, to the radiation source. The minimum signed distance along a projection ray is the signed distance for that edge.

The stochastic estimates of the entropy is evaluated as:

$$
\begin{aligned}
H_1(s') &= -\int ds'\; p_1(s')\; \ln p_1(s') \\
p_1(s') &\approx \frac{1}{\sum_{s'_i \in \mathbf{I}} P(\mathbf{e}_i = 1)} \sum_{s'_i \in \mathbf{I}} P(\mathbf{e}_i = 1)\; G_{\Psi_1}(s' - s'_i) \\
&= \frac{1}{\sum_{s'_i \in \mathbf{I}} < e_i >} \sum_{s'_i \in \mathbf{I}} < e_i >\; G_{\Psi_1}(s' - s'_i) \\
H_1(s') &\approx \left(\frac{-1}{\sum_{s'_j \in \mathbf{J}} < e_j >} \right) \sum_{s'_j \in \mathbf{J}} < e_j >\; \ln p_1(s'_j) \\
&= \left(\frac{-1}{\sum_{s'_j \in \mathbf{J}} < e_j >} \right) \sum_{s'_j \in \mathbf{J}} \left[< e_j >\; \ln \left(\frac{1}{\sum_{s'_i \in \mathbf{I}} < e_i >} \sum_{s'_i \in \mathbf{I}} < e_i >\; G_{\Psi_1}(s'_j - s'_i) \right) \right]
\end{aligned}
$$

D.6.3.1 Evaluating the derivatives

Similar to the previous subsection, the Gaussian kernel in estimating the marginal entropy, $H_1(s')$, is given as:

$$
G_{\Psi_1}(s) = (2\pi)^{-1/2}\; |\psi_1|^{-1/2}\; \exp\left(\frac{-1}{2} s\; \psi_1^{-1}\; s \right)
$$

Thus, the derivative of the joint entropy $H_1(s')$ with respect to the transformation parameters can now be written as:

$$
\frac{d}{dT} H_1(s')
$$

$$
\begin{aligned}
&= \left(\frac{-1}{\sum_{s'_j \in \mathbf{J}} < e_j >}\right) \sum_{s'_j \in \mathbf{J}} \left[< e_j > \frac{\sum_{s'_i \in \mathbf{I}} < e_i > \frac{d}{dT} G_{\Psi_1}(s'_j - s'_i)}{\sum_{s'_i \in \mathbf{I}} < e_i > G_{\Psi_1}(s'_j - s'_i)} \right] \\
&= \left(\frac{1}{\sum_{s'_j \in \mathbf{J}} < e_j >}\right) \sum_{s'_j \in \mathbf{J}} \left[< e_j > \frac{\sum_{s'_i \in \mathbf{I}} < e_i > G_{\Psi_1}(s'_j - s'_i)(s'_j - s'_i)\psi_1^{-1} \frac{d}{dT}(s'_j - s'_i)}{\sum_{s'_i \in \mathbf{I}} < e_i > G_{\Psi_1}(s'_j - s'_i)} \right] \\
&= \left(\frac{1}{\sum_{s'_j \in \mathbf{J}} < e_j >}\right) \sum_{s'_j \in \mathbf{J}} \left[< e_j > \sum_{s'_i \in \mathbf{I}} \left(\frac{< e_i > G_{\Psi_1}(s'_j - s'_i)}{\sum_{s'_i \in \mathbf{I}} < e_i > G_{\Psi_1}(s'_j - s'_i)} \right) (s'_j - s'_i)\psi_1^{-1} \frac{d}{dT}(s'_j - s'_i) \right] \\
&= \left(\frac{1}{\sum_{s'_j \in \mathbf{J}} < e_j >}\right) \sum_{s'_j \in \mathbf{J}} \left[< e_j > \sum_{s'_i \in \mathbf{I}} W_{\Psi_1}(s'_j, s'_i)\ (s'_j - s'_i)\psi_1^{-1} \frac{d}{dT}(s'_j - s'_i) \right]
\end{aligned}
$$

where we have defined

$$
W_{\Psi_1}(s'_j, s'_i) \equiv \left(\frac{< e_i > G_{\Psi_1}(s'_j - s'_i)}{\sum_{s'_k \in \mathbf{I}} < e_k > G_{\Psi_1}(s'_j - s'_k)} \right)
$$

Evaluation of the terms $\frac{d}{dT}(s'_j - s'_i)$ is a major concern which needs to be resolved before implementation. Note that $\frac{d}{dT} s'_i$ represents a vector of values,

$$
\frac{d}{dT} s'_i = \begin{pmatrix} \frac{d}{d\alpha} \\ \frac{d}{d\beta} \\ \frac{d}{d\gamma} \\ \frac{d}{dt_x} \\ \frac{d}{dt_y} \\ \frac{d}{dt_z} \end{pmatrix} s'_i
$$

where, $\alpha, \beta, \gamma, t_x, t_y, t_z$ denote the six parameters of the rigid transformation. Also note that s'_i represents the value of a 3D voxel in the signed distance transformed 3D CT image data set. Thus, $\frac{ds'_i}{dt_x}$, for example, represents how does s'_i changes with small change in

the transformation parameter t_x. Thus, we write,

$$\frac{ds'_i}{dt_x} = \frac{ds'_i}{dx}\,\frac{dx}{dt_x} + \frac{ds'_i}{dy}\,\frac{dy}{dt_x} + \frac{ds'_i}{dz}\,\frac{dz}{dt_x}$$

Remember now that s'_i is the value in three–space of the voxel with the minimum value, in the distance transformed 3D dataset, along the projection ray from the ith pixel in the 2D image. Note that in the above formulation we have assumed that, for small changes in transformation parameters, the same point in the three–space will have the minimum distance along the projection ray. This assumption is not true in general, even in the limiting case of very small changes in the transformation parameters.

D.6.3.2 Evaluation of the variance

Define, $\sigma_1 = \sqrt{\psi_1}$. Thus, we have,

$$\begin{aligned}
G_{\Psi_1}(s) &= (2\pi)^{-1/2}\,|\Psi_1|^{-1/2}\,\exp\left(\frac{-1}{2}s\,\psi_1^{-1}\,s\right) \\
&= (2\pi)^{-1/2}\,\sigma_1^{-1}\,\exp\left(\frac{-1}{2}s\,\sigma_1^{-2}\,s\right) \\
\frac{d}{d\sigma_1}G_{\Psi_1}(s) &= G_{\Psi_1}(s)\,\frac{1}{\sigma_1}\left[\frac{s^2}{\sigma_1^2} - 1\right]
\end{aligned}$$

Thus, we have

$$\begin{aligned}
&\frac{d}{d\sigma_1}H_1(s') \\
&= \left(\frac{-1}{\sum_{s'_j\in\mathbf{J}} < e_j >}\right)\sum_{s'_j\in\mathbf{J}}\left[< e_j > \frac{\sum_{s'_i\in\mathbf{I}} < e_i > \frac{d}{d\sigma_1}G_{\Psi_1}(s'_j - s'_i)}{\sum_{s'_i\in\mathbf{I}} < e_i > G_{\Psi_1}(s'_j - s'_i)}\right]
\end{aligned}$$

$$= \left(\frac{1}{\sum_{s'_j \in \mathbf{J}} < e_j >}\right) \sum_{s'_j \in \mathbf{J}} \left[< e_j > \frac{\sum_{s'_i \in \mathbf{I}} < e_i > G_{\Psi_1}(s'_j - s'_i) \frac{1}{\sigma_1} \left(\frac{(s'_j - s'_i)^2}{\sigma_1^2} - 1 \right)}{\sum_{s'_i \in \mathbf{I}} < e_i > G_{\Psi_1}(s'_j - s'_i)} \right]$$

$$= \left(\frac{1}{\sum_{s'_j \in \mathbf{J}} < e_j >}\right) \sum_{s'_j \in \mathbf{J}} \left[< e_j > \sum_{s'_i \in \mathbf{I}} \left(\frac{< e_i > G_{\Psi_1}(s'_j - s'_i)}{\sum_{s'_i \in \mathbf{I}} < e_i > G_{\Psi_1}(s'_j - s'_i)} \right) \frac{1}{\sigma_1} \left(\frac{(s'_j - s'_i)^2}{\sigma_1^2} - 1 \right) \right]$$

$$= \left(\frac{1}{\sum_{s'_j \in \mathbf{J}} < e_j >}\right) \sum_{s'_j \in \mathbf{J}} \left[< e_j > \sum_{s'_i \in \mathbf{I}} W_{\Psi_1}(s'_j, s'_i) \, \frac{1}{\sigma_1} \left(\frac{(s'_j - s'_i)^2}{\sigma_1^2} - 1 \right) \right]$$

$$= \left(\frac{-1}{\sigma_1}\right) + \left(\frac{1}{\sum_{s'_j \in \mathbf{J}} < e_j >}\right) \sum_{s'_j \in \mathbf{J}} \left[< e_j > \sum_{s'_i \in \mathbf{I}} W_{\Psi_1}(s'_j, s'_i) \, \frac{(s'_j - s'_i)^2}{\sigma_1^3} \right]$$

D.6.4 Evaluating $H_{\alpha\beta}(x_2, y_2, x_1, y_1)$

Let, $w_2 \equiv (x_2, y_2)^T$ and $w_1 \equiv (x_1, y_1)^T$ to reduce the notational complexity. Thus, we have

$$\begin{aligned} H_{\alpha\beta}(x_2, y_2, x_1, y_1) &\equiv H_{\alpha\beta}(w_2, w_1) \\ &= -\int\int dw_2 \, dw_1 \, p_{\alpha\beta}(w_2, w_1) \, \ln p_{\alpha\beta}(w_2, w_1) \end{aligned}$$

The joint density function $p_{\alpha\beta}(w_2, w_1)$ is approximated as:

$$\begin{aligned} p_{\alpha\beta}(w_2, w_1) &\approx \frac{1}{\sum_{\{w_{2i}, w_{1i}\} \in \mathbf{I}} P_i(\alpha, \beta)} \sum_{\{w_{2i}, w_{1i}\} \in \mathbf{I}} P_i(\alpha, \beta) \, G_{\Psi_{\alpha\beta}}(w_2 - w_{2i}, w_1 - w_{1i}) \\ H_{\alpha\beta}(w_2, w_1) &\approx \left(\frac{-1}{\sum_{\{w_{2j}, w_{1j}\} \in \mathbf{J}} P_j(\alpha, \beta)} \right) \sum_{\{w_{2j}, w_{1j}\} \in \mathbf{J}} P_j(\alpha, \beta) \, \ln p_{\alpha\beta}(w_{2j}, w_{1j}) \\ &= \left(\frac{-1}{\sum_{\{w_{2j}, w_{1j}\} \in \mathbf{J}} P_j(\alpha, \beta)} \right) \sum_{\{w_{2j}, w_{1j}\} \in \mathbf{J}} \left[P_j(\alpha, \beta) \, \ln \left(\frac{1}{\sum_{\{w_{2i}, w_{1i}\} \in \mathbf{I}} P_i(\alpha, \beta)} \right.\right. \\ &\qquad \left.\left. \sum_{\{w_{2i}, w_{1i}\} \in \mathbf{I}} P_i(\alpha, \beta) \, G_{\Psi_{\alpha\beta}}(w_{2j} - w_{2i}, w_{1j} - w_{1i}) \right) \right] \end{aligned}$$

D.6.4.1 Evaluation of the derivatives

Let, $\Psi_{\alpha\beta} = \text{Diag}(\psi_{\alpha\beta x2}, \psi_{\alpha\beta y2}, \psi_{\alpha\beta x1}, \psi_{\alpha\beta y1})$ be a diagonal covariance matrix. The Gaussian kernel used for the Parzen window estimates of the joint density is given as:

$$
\begin{aligned}
G_{\Psi_{\alpha\beta}}(w_2, w_1) &= (2\pi)^{-2}|\Psi_{\alpha\beta}|^{-1/2} \exp\left[\frac{-1}{2}(w_2 \;\; w_1)\Psi_{\alpha\beta}\begin{pmatrix} w_2 \\ w_1 \end{pmatrix}\right] \\
\frac{d}{dT}G_{\Psi_{\alpha\beta}}(w_2, w_1) &= G_{\Psi_{\alpha\beta}}(w_2, w_1) \;\; \left(\frac{-1}{2}\right)\frac{d}{dT}\left[(w_2 \;\; w_1)\Psi_{\alpha\beta}\begin{pmatrix} w_2 \\ w_1 \end{pmatrix}\right] \\
&= -G_{\Psi_{\alpha\beta}}(w_2, w_1) \;\; \left[y_2 \;\; \psi^{-1}_{\alpha\beta y2} \;\; \frac{d}{dT}y_2 + y_1 \;\; \psi^{-1}_{\alpha\beta y1} \;\; \frac{d}{dT}y_1\right]
\end{aligned}
$$

Therefore, the derivative can now be written as:

$$
\begin{aligned}
&\frac{d}{dT}G_{\Psi_{\alpha\beta}}(w_{2j} - w_{2i}, w_{1j} - w_{1i}) \\
&\qquad = -G_{\Psi_{\alpha\beta}}(w_{2j} - w_{2i}, w_{1j} - w_{1i}) \left[(y_{2j} - y_{2i}) \;\; \psi^{-1}_{\alpha\beta y2} \;\; \frac{d}{dT}(y_{2j} - y_{2i}) + \right. \\
&\qquad\qquad \left. (y_{1j} - y_{1i}) \;\; \psi^{-1}_{\alpha\beta y1} \;\; \frac{d}{dT}(y_{1j} - y_{1i})\right]
\end{aligned}
$$

Thus, the derivatives of the joint density function with respect to the transformation parameters is now given as

$$
\begin{aligned}
&\frac{d}{dT}H_{\alpha\beta}(w_2, w_1) \\
&= \left(\frac{-1}{\sum_{\{w_{2j}, w_{1j}\}\in\mathbf{J}} P_j(\alpha, \beta)}\right) \sum_{\{w_{2j}, w_{1j}\}\in\mathbf{J}} P_j(\alpha, \beta) \\
&\qquad \left[\frac{\sum_{\{w_{2i}, w_{1i}\}\in\mathbf{I}} P_i(\alpha, \beta) \; \frac{d}{dT}G_{\Psi_{\alpha\beta}}(w_{2j} - w_{2i}, w_{1j} - w_{1i})}{\sum_{\{w_{2i}, w_{1i}\}\in\mathbf{I}} P_i(\alpha, \beta) \; G_{\Psi_{\alpha\beta}}(w_{2j} - w_{2i}, w_{1j} - w_{1i})}\right]
\end{aligned}
$$

$$
\begin{aligned}
&= \left(\frac{1}{\sum_{\{w_{2j},w_{1j}\}\in\mathbf{J}} P_j(\alpha,\beta)}\right) \sum_{\{w_{2j},w_{1j}\}\in\mathbf{J}} \Bigg[P_j(\alpha,\beta) \sum_{\{w_{2i},w_{1i}\}\in\mathbf{I}} \\
&\left(\frac{P_i(\alpha,\beta)\, G_{\Psi_{\alpha\beta}}(w_{2j}-w_{2i}, w_{1j}-w_{1i})}{\sum_{\{w_{2i},w_{1i}\}\in\mathbf{I}} P_i(\alpha,\beta)\, G_{\Psi_{\alpha\beta}}(w_{2j}-w_{2i}, w_{1j}-w_{1i})}\right) \\
&\left((y_{2j}-y_{2i})\ \ \psi^{-1}_{\alpha\beta y2}\ \ \frac{d}{dT}(y_{2j}-y_{2i}) + (y_{1j}-y_{1i})\ \ \psi^{-1}_{\alpha\beta y1}\ \ \frac{d}{dT}(y_{1j}-y_{1i})\right)\Bigg] \\
&= \left(\frac{1}{\sum_{\{w_{2j},w_{1j}\}\in\mathbf{J}} P_j(\alpha,\beta)}\right) \sum_{\{w_{2j},w_{1j}\}\in\mathbf{J}} \Bigg[P_j(\alpha,\beta) \sum_{\{w_{2i},w_{1i}\}\in\mathbf{I}} W_{\Psi_{\alpha\beta}}(w_{2j}-w_{2i}, w_{1j}-w_{1i}) \\
&\left((y_{2j}-y_{2i})\ \ \psi^{-1}_{\alpha\beta y2}\ \ \frac{d}{dT}(y_{2j}-y_{2i}) + (y_{1j}-y_{1i})\ \ \psi^{-1}_{\alpha\beta y1}\ \ \frac{d}{dT}(y_{1j}-y_{1i})\right)\Bigg]
\end{aligned}
$$

where,

$$
W_{\Psi_{\alpha\beta}}(w_{2j}-w_{2i}, w_{1j}-w_{1i}) \equiv \left(\frac{P_i(\alpha,\beta)\, G_{\Psi_{\alpha\beta}}(w_{2j}-w_{2i}, w_{1j}-w_{1i})}{\sum_{\{w_{2i},w_{1i}\}\in\mathbf{I}} P_i(\alpha,\beta)\, G_{\Psi_{\alpha\beta}}(w_{2j}-w_{2i}, w_{1j}-w_{1i})}\right)
$$

D.6.4.2 Evaluation of the covariance matrix parameters

Let, $\sigma_{\alpha\beta x2} = \sqrt{\psi_{\alpha\beta x2}}$, $\sigma_{\alpha\beta y2} = \sqrt{\psi_{\alpha\beta y2}}$, $\sigma_{\alpha\beta x1} = \sqrt{\psi_{\alpha\beta x1}}$, $\sigma_{\alpha\beta y1} = \sqrt{\psi_{\alpha\beta y1}}$.

Then the Gaussian kernel can be written in the following factored form:

$$
\begin{aligned}
G_{\Psi_{\alpha\beta}}(w_2, w_1) &= G_{\sigma_{\alpha\beta x2}}(x_2)\, G_{\sigma_{\alpha\beta y2}}(y_2)\, G_{\sigma_{\alpha\beta x1}}(x_1)\, G_{\sigma_{\alpha\beta y1}}(y_1) \\
\frac{dG_{\Psi_{\alpha\beta}}(w_2, w_1)}{\sigma_{\alpha\beta x2}} &= G_{\Psi_{\alpha\beta}}(w_2, w_1)\, \frac{1}{\sigma_{\alpha\beta x2}} \left[\frac{x_2^2}{\sigma^2_{\alpha\beta x2}} - 1\right]
\end{aligned}
$$

Thus, the derivative of $H_{\alpha\beta}(w_2, w_1)$ with respect to the standard deviation $\sigma_{\alpha\beta x2}$ is now given as:

$$
\frac{dH_{\alpha\beta}(w_2, w_1)}{d\sigma_{\alpha\beta x2}}
$$

$$
\begin{aligned}
&= \left(\frac{-1}{\sum_{\{w_{2j},w_{1j}\}\in\mathbf{J}} P_j(\alpha,\beta)}\right) \sum_{\{w_{2j},w_{1j}\}\in\mathbf{J}} P_j(\alpha,\beta) \\
&\quad \left[\frac{\sum_{\{w_{2i},w_{1i}\}\in\mathbf{I}} P_i(\alpha,\beta)\ \frac{d}{d\sigma_{\alpha\beta x2}} G_{\Psi_{\alpha\beta}}(w_{2j}-w_{2i},w_{1j}-w_{1i})}{\sum_{\{w_{2i},w_{1i}\}\in\mathbf{I}} P_i(\alpha,\beta)\ G_{\Psi_{\alpha\beta}}(w_{2j}-w_{2i},w_{1j}-w_{1i})}\right] \\
&= \left(\frac{1}{\sum_{\{w_{2j},w_{1j}\}\in\mathbf{J}} P_j(\alpha,\beta)}\right) \sum_{\{w_{2j},w_{1j}\}\in\mathbf{J}} \left\{ P_j(\alpha,\beta) \sum_{\{w_{2i},w_{1i}\}\in\mathbf{I}} \right. \\
&\quad \left. \left(\frac{P_i(\alpha,\beta)\ G_{\Psi_{\alpha\beta}}(w_{2j}-w_{2i},w_{1j}-w_{1i})}{\sum_{\{w_{2i},w_{1i}\}\in\mathbf{I}} P_i(\alpha,\beta)\ G_{\Psi_{\alpha\beta}}(w_{2j}-w_{2i},w_{1j}-w_{1i})}\right) \frac{1}{\sigma_{\alpha\beta x2}} \left[\frac{(x_{2j}-x_{2i})^2}{\sigma^2_{\alpha\beta x2}} - 1\right]\right\} \\
&= \left(\frac{1}{\sum_{\{w_{2j},w_{1j}\}\in\mathbf{J}} P_j(\alpha,\beta)}\right) \sum_{\{w_{2j},w_{1j}\}\in\mathbf{J}} \left\{ P_j(\alpha,\beta) \sum_{\{w_{2i},w_{1i}\}\in\mathbf{I}} W_{\Psi_{\alpha\beta}}(w_{2j}-w_{2i},w_{1j}-w_{1i}) \right. \\
&\quad \left. \frac{1}{\sigma_{\alpha\beta x2}} \left[\frac{(x_{2j}-x_{2i})^2}{\sigma^2_{\alpha\beta x2}} - 1\right]\right\}
\end{aligned}
$$

The derivatives with respect to other standard deviations can be similarly computed.

Bibliography

[1] J.A. Antolak et al. Prostate target volume variation during a course of radiotherapy. *Int. J. Radiation Oncology Biol. Phys.*, 42(3):661–672, 1998.

[2] K. S. Arun, T. S. Huang, and S. D. Blostein. Least–squares fitting of two 3–d point sets. *IEEE Trans. on PAMI*, PAMI–9(5):698–700, Sept. 1987.

[3] J. Balter, C. Pelizzari, et al. Correlation of projection radiographs in radiation therapy using open curve segments and points. *Med. Phys.*, 19:329–334, 1992.

[4] R. Bansal, L. Staib, Z. Chen, A. Rangarajan, J. Knisely, R. Nath, and J.S. Duncan. A novel approach for the registration of 2D portal and 3D CT images for treatment setup verification in radiotherapy. *Medical Image Computing and Computer–Assisted Intervention (MICCAI'98)*, LNCS–1496:1075–1086, 10–12 October 1998.

[5] Martin Berger and Guido Gerig. Motion Measurments in Low–Contrast X–ray Imagery. In W.M. Wells et al., editors, *Medical Image Computing and Computer–Assisted Intervention –MICCAI'98*, volume 1496 of *LNCS*, pages 832–841. Springer, 1998.

[6] M. Bertero et al. Ill–posed problems in early vision. *Proc. of the IEEE*, 76(8):869–889, Aug 1988.

[7] J. Besag. Spatial interaction and statistical analysis of lattice systems. *J. of the Royal Statistical Society, B*, 36:192–225, 1974.

[8] J. Besag. On the statistical analysis of dirty pictures. *J. of the Royal Statistical Society, B*, 48(3):259–302, June 1986.

[9] P. J. Besl. A method for registration of 3–d shapes. *IEEE Trans. on PAMI*, 14(2):239–256, February 92.

[10] R. M. Bevensee. *Maximum Entropy Solutions to Scientific Problems.* P T R Prentice Hall, 1993.

[11] A. Blake and A. Zisserman. *Visual Reconstruction.* MIT Press, Cambridge, 1987.

[12] G. Borgefors. *On hierarchical edge matching in digital images using distance transformations.* PhD thesis, The Royal Institute of Technology. Stockholm. Sweden, 1986.

[13] R. P. Brent. *Algorithms for minimization without derivatives.* Prentice–Hall, Englewood Cliffs, 1973.

[14] L. Brown and T. E. Boult. Registration of planar film radiographs with computed tomography. *Mathematical Methods in Biomedical Image Analysis*, pages 42–51, June 1996. IEEE Proc.

[15] L.G. Brown. A survey of image registration techniques. *ACM Computing Surveys*, 24(4):325–376, 1992.

[16] H. Burkhardt and N. Diehl. Simultaneous estimation of rotation and translation in image sequences. In I.T. Young et al., editors, *Signal Processing III: Theories*

and Applications, pages 821–824. Elsevier Science Publishers B.V. (North–Holland), 1986.

[17] J. Cai et al. A simple algorithm for planar image registration in radiation therapy. *Medical Physics*, 25(6):824–9, Jan 1998.

[18] David Chandler. *Introduction to Modern Statistical Mechanics*. Oxford University Press, 1987.

[19] E. E. Christensen, T. S. Curry, and J. Nunnally. *An Introduction to the Physics of Diagnostic Radiology*. Henry Kimpton Publishers, 2nd edition, 1973.

[20] G.E. Christensen et al. 3d brain mapping using a deformable neuroanatomy. 39:609–618, 1994.

[21] A. Collignon et al. New high–performance 3D registration algorithms for 3D medical images. *SPIE. Image Processing*, 1898:779–788, 1993.

[22] A. Collignon, F. Maes, et al. Automated multimodality image registration using information theory. *Info. Proc. in Med. Imaging (IPMI)*, pages 263–274, June 1995.

[23] T.F. Cootes et al. Active shape models–their training and application. *Computer Vision and Image Understanding*, 61(1):38–59, 1995.

[24] T. Cover and J. Thomas. *Elements of Information Theory*. John Wiley and Sons, Inc., 1991.

[25] L.A. Dawson et al. Target position variability throughput prostate radiotherapy. *Int. J. Radiation Oncology Biol. Phys.*, 42(5):1155–1161, 1998.

[26] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via EM algorithm. *J. Royal Statistical Soc., Ser. B*, 39:1–38, 1977.

[27] H. Derin and H. Elliott. Modeling and segmentation of noisy and textured images using Gibb's random fields. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 9:39–55, 1987.

[28] N. Diehl and H. Burkhardt. Motion estimation in image sequences. *High Precision Navigation*, pages 297–312, 1989. Springer–Verlag, New York.

[29] L. Dong and A. L. Boyer. An image correlation procedure for digitally reconstructed radiographs and electronic portal images. *Int. J. Radiation Oncology Biol. Phys.*, 33(5):1053–1060, 1995.

[30] L.L. Doss. Localization error and local recurrence in upper airway carcinoma. In L.E. Reinstein, editor, *Workshop on Quality Control in Radiotherapy Departments*, 1979.

[31] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons, 1973.

[32] P.A. Elsen et al. Automatic registration of CT and MR brain images using correlation of geometric features. *IEEE Trans. Med. Img.*, 14(2):384–396, 1995.

[33] L.M. Florack et al. Scale and differential structure of images. *Image and Vision Computing*, 10(6):376–388, 1992.

[34] D. S. Fritsch et al. Core–based portal image registration for automatic radiotherapy treatment verification. *Int. J. Radiation Oncology Biol. Phys.*, 33(5):1287–1300, 1995.

[35] D.S. Fritsch. *Registration of radiotherapy images using multiscale medial descriptions of image structure.* PhD thesis, University of North Carolina at Chapel Hill, 1993.

[36] T. Gaens et al. Non–rigid multimodal image registration using mutual information. *Medical Image Computing and Computer–Assisted Intervention*, LNCS–1496:1099–1106, 1998.

[37] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions and the bayesian restoration of images. *IEEE Trans. PAMI*, 6(6):721–741, Nov. 1984.

[38] K. Gilhuijs. *Automated verification of radiation treatment geometry.* PhD thesis, Univ. of Amsterdam, Radiotherapy Dept., The Netherlands, 1995.

[39] K. Gilhuijs et al. Interactive three dimensional inspection of patient setup in radiation therapy using digital portal images and computed tomography data. *Int. J. Radiation Oncology Biol. Phys.*, 34(4):873–885, 1996.

[40] K. G. A. Gilhuijs and M. A. van Herk. Automatic on–line inspection of patient setup in radiation therapy using digital portal images. *Med. Phys.*, 20(3):667–677, May/June 1993.

[41] J.D. Graham et al. A noninvasive relocatable stereotactic frame for fractionated radiotherapy and multiple imaging. *Radiother. Oncol.*, 21:60–62, 1991.

[42] A. Hamadeh et al. Automated 3–Dimensional Computed Tomographic and Fluoroscopic Image Registration. *Comp. Aid. Surg.*, 3:11–19, 1998.

[43] J. Hanley et al. The effects of out–of–plane rotations on two dimensional portal image registration in conformal radiotherapy of the prostate. *Int. J. Radiation Oncology Biol. Phys.*, 33(5):1331–1343, 1995.

[44] F. Hansen and H. Elliott. Image segmentation using simple markov random field models. *Computer Graphics and Image Understanding*, 20:101–132, 1982.

[45] G. Hardy et al. *Inequalities.* Cambridge University Press, 2nd edition, 1952.

[46] R. J. Hathaway. Another interpretation of the EM algorithm for mixture distributions. *Statistics & Probability Letters, North Holland*, 4:53–56, 1986.

[47] S. Haykin. *Neural Networks: A Comprehensive Foundation.* Macmillan College Publishing, 1994.

[48] D. H. Hiristov and B. G. Fallone. A gray–level image alignment algorithm for registration of portal images and digitally reconstructed radiographs. *Med. Phys.*, 23(1):75–84, Jan 1996.

[49] T. Hofmann and J. Buhmann. Pairwise Data Clustering by Deterministic Annealing. *IEEE Trans. on PAMI*, 19(1):1–14, Jan. 1997.

[50] J. H. Hubble. *Photon Cross Sections, Attenuation Coefficients, and Energy Absorption Coefficients From 10KeV to 100 GeV.* Nat. Stand. Ref. Data. Ser., Nat. Bur. Stand. (U.S.), August 1969.

[51] R.A. Hummel and S. Zucker. On the foundations of relaxation labeling processes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 5:267–287, 1983.

[52] E. T. Jaynes. Information theory and statistical mechanics I. In R. D. Rosenkrantz, editor, *E. T. Jaynes: Papers on probability, statistics and statistical physics*, volume 158, pages 4–16. D. Reidel Publishing Company, Boston. USA, 1983.

[53] E. T. Jaynes. Information theory and statistical mechanics II. In R. D. Rosenkrantz, editor, *E. T. Jaynes: Papers on probability, statistics and statistical physics*, volume 158, pages 17–38. D. Reidel Publishing Company, Boston. USA, 1983.

[54] E. T. Jaynes. Prior probabilities. In R. D. Rosenkrantz, editor, *E. T. Jaynes: Papers on probability, statistics and statistical physics*, volume 158, pages 114–130. D. Reidel Publishing Company, Boston. USA, 1983.

[55] T. Kapur et al. Enhanced spatial priors for segmentation of magnetic resonance imagery. In W.M. Wells et al., editors, *Medical Image Computing and Computer–Assisted Intervention*, volume 1496, pages 457–468. LNCS, Springer, 1998.

[56] S. Kirkpatrick et al. Optimization by simulated annealing. *Science*, 220:671–680, 1983.

[57] J.J. Kosowsky and A.L. Yuille. The Invisible Hand Algorithm: Solving the Assignment Problem With Statistical Physics. *Neural Networks*, 7(3):477–490, 1994.

[58] H. Kushner and D.S. Clark. *Stochastic Approximation Methods for Constrained and Unconstrained Systems.* Springer–Verlag, 1978.

[59] S. Lavallee. Registration for computer–integrated surgery: Methodology, state of the art. In Russell H. Taylor et al., editors, *Computer–Integrated Surgery: Technology and Clinical Applications*, pages 77–97. The MIT Press, Cambridge, 1996.

[60] S. Lavallee, R. Szeliski, and L. Brunie. Anatomy–based registration of three–dimensional medical images, range images, X–ray projections, and three–dimensional models using octree–splines. In Russell H. Taylor et al., editors, *Computer–Integrated Surgery: Technology and Clinical Applications*, pages 115–143. The MIT Press, Cambridge, Massachusetts, 1996.

[61] R. Leahy et al. Applications of markov random fields in medical imaging. *Information Processing in Medical Imaging (IPMI)*, pages 1–14, 1991.

[62] M. Lee. *Bayesian Reconstruction In Emission Tomography Using Gibbs Priors*. PhD thesis, Yale University, May 1994. Online:http://noodle.med.yale.edu/lee/thesis/title.html.

[63] L. Lemieux et al. A patient–to–computed–tomography image registration method based on digitally reconstructed radiographs. *Med. Phys.*, 21(11):1749–1760, November 1994.

[64] K. W. Leszczynski, S. Loose, and S. Boyko. An image registration scheme applied to verification of radiation therapy. *The British Journal of Radiology*, 71(844):413–426, 1998.

[65] M. Leventon and E. Grimson. Multi–modal volume registration using joint distributions. In W.M. Wells et al., editors, *Medical Image Computing and Computer–Assisted Intervention*, volume 1496, pages 1057–1066. LNCS, Springer, 1998.

[66] L. Ljung. Analysis of recursive stochastic algorithms. *IEEE Transactions on Automatic Control*, AC–22:551–575, 1977.

[67] J.B. Maintz and M.A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2(1):1–36, 1998.

[68] C.R. Maurer et al. Registration of 3D images using weighted geometrical features. *IEEE Trans. on Med. Imag.*, 15(6):1163–1178, 1996.

[69] H. Meertens, J. Bijhold, and J. Strackee. A method for the measurement of field displacement errors in digital portal images. *Phys. Med. Biol.*, 35(3):299–323, 1990.

[70] X. Meng and D. Dyk. The EM algorithm– an old folk–song sung to a fast new tune. *J. R. Statis. Soc. B.*, 59(3):511–567, 1997.

[71] T. K. Moon. The expectation–maximization algorithm. In *IEEE Signal Processing Magazine*, pages 47–60. IEEE, November 1996.

[72] C. D. Mubata, A. M. Bidmead, et al. Portal imaging protocol for radical dose–escalated radiotherapy treatment of prostate cancer. *Int. J. Radiation Oncology Biol. Phys.*, 40(1):221–231, 1998.

[73] R. M. Neal and G. E. Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In M. I. Jordan, editor, *Learning in Graphical Models*. Kluwer Academic Press. To Appear. Web address: http://www.cs.toronto.edu/∼radford/papers-online.html.

[74] S. Negahdaripour et al. A generalized brightness change model for computing optical flow. *Int. Conf. on Comp. Vis.*, pages 2–11, 1993.

[75] J. A. Nelder and R. Mead. A simplex method for function minimization. *Comp. J.*, 7:308–313, 1965.

[76] W. D. Neve, F. Heuvel, et al. Interactive use of on–line portal imaging in pelvic radiation. *Int. J. Radiation Oncology Biol. Phys.*, 25:517–524, 1993.

[77] A. Papoulis. *Probability, Random Variable, and Stochastic Processes.* McGraw–Hill, Inc., 3 edition, 1991.

[78] S. Pappu and A. Rangarajan. Articulated matching with point features. *submitted to IEEE PAMI*, 1997.

[79] G. Parisi. *Statistical Field Theory.* Addison–Wesley, 1988.

[80] C.A. Pelizzari et al. Accurate three dimensional registration of ct, pet and/or mr images of the brain. *J. of Comp. Assis. Tomo.*, 13:20–26, 1989.

[81] S.M. Pizer et al. Object shape before boundary shape: Scale–space medial axes. *J. Math. Imaging Vision*, 4:303–313, 1994.

[82] T. Poggio et al. Computational vision and regularization theory. *Nature*, 317(26):314–319, Sept. 1985.

[83] M. J. D. Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *Comput. J.*, 7:155–163, 1964.

[84] A. Rangarajan and R. Chellapa. A continuation method for image estimation and segmentation. Technical report, Unin. of Maryland, Center for Automation Research, Oct. 1991.

[85] A. Rangarajan et al. A robust point matching algorithm for autoradiograph alignment. In K.H. Hohne and R. Kikinis, editors, *Visualization in Biomedical Computing (VBC)*, pages 277–286. 1996.

[86] Anand Rangarajan et al. The softassign procrustes matching algorithm. *Information Processing in Medical Imaging*, pages 29–42, 1997.

[87] R. A. Redner and H. F. Walker. Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review*, 26(2):195–239, 1984.

[88] F. Reza. *An Introduction to Information Theory*. Dover, 1994.

[89] H. Robins and S. Munroe. A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407, 1951.

[90] B.t.H. Romeny et al. A multiscale geometric model of human vision. In B. Hendee and P. Wells, editors, *Perception of Visual Information*. 1992. Springer–Verlag, Berlin, Germany.

[91] A. Rosenfeld and A. Kak. *Digital Picture Processing*, volume 1,2. Academic, 2nd edition, 1982.

[92] W. Schlegel. Computer systems and mechanical tools for stereotactically guided conformation therapy with linear accelerator. *Int. J. Radiation Oncology Biol. Phys.*, 24:781–787, 1992.

[93] W. Schlegel. Requirements in computer–assisted radiotherapy. In Grigore C. Burdea Russell H. Taylor, Stephane Lavallle and Ralph Mosges, editors, *Computer–Integrated Surgery: Technology and Clinical Applications*, pages 681–691. The MIT Press, Cambridge, Massachusetts, 1996.

[94] T. Schormann et al. A new approach to fast elastic alignment with application to human brains. *Visualization in Biomedical Computing*, pages 338–342.

[95] Sensor Systems, Inc. *MEDx^{TM}: Multimodality Radiological Image Processing For Unix Workstations*, 1998. Version 2.1.

[96] J. Serra. *Image Analysis and Mathematical Morphology*, volume 1. Academic, 2 edition, 1988.

[97] C. E. Shannon and W. Weaver. *The Mathematical Theory of Communication*. The University of Illinois Press, 1949.

[98] C. Studholme et al. Incorporating connected region labelling into automated image registration using mutual information. In *Proc. of MMBIA'96*, pages 23–31. IEEE, 1996.

[99] Colin Studholme. *Measures of 3D Medical Image Alignment*. PhD thesis, Guy's and St. Thomas's Hospitals, University of London, Aug 1997.

[100] H. Tagare. A gentle introduction to the EM algorithm: Part I: Theory. Technical Report TR 99–03, Yale University, 1999.

[101] H. D. Tagare, G. Zubal, and J. Duncan. Using hierarchical landmarks for 3-d registration of images. Technical report, Yale University, 1993.

[102] J.P. Thirion. Image matching as a diffusion process: an analogy with maxwell's demons. *Medical Image Analysis*, 2(3):243–260, 1998.

[103] J. Troccaz, , V. Menguy, et al. Conformal external radiotherapy: Preliminary results in the treatment of prostatic carcinoma. In Grigore C. Burdea Russell H. Taylor, Stephane Lavallle and Ralph Mosges, editors, *Computer–Integrated Surgery: Technology and Clinical Applications*, pages 707–714. The MIT Press, Cambridge, Massachusetts, 1996.

[104] T.G. Turkington et al. Accuracy of surface fit registration for pet and mr brain images using full and incomplete brain surfaces. *J. of Comp. Aid. Tomo.*, 19:117–124, 1995.

[105] P. van den Elsen, J. Maintz, E. Pol, and M. Viergever. Image fusion using geometical features. In S. Pizer, editor, *Visualization in Biomedical Computing*, pages 172–186. SPIE, 1992.

[106] P. A. van den Elsen. *Multimodality Matching of Brain Images*. PhD thesis, Utrecht University, 1993.

[107] P. A. van den Elsen, E. D. Pol, and M. A. Viergever. Medical image matching– a review with classification. *IEEE Eng. in Med. and Biol.*, 12(1):26–39, 1993.

[108] M. van Herk et al. Effect of image artifacts, organ motion and poor segmentation on the reliability and accuracy of three–dimensional chamfer matching. *Comp. Aid. Surg.*, 2:346–355, 1997.

[109] B. C. Vemuri et al. An efficient motion estimator with application to medical image registration. *Medical Image Analysis*, 2(1):79–98, 1998.

[110] P. Viola. *Alignment by Maximization of Mutual Information*. A. I. technical report no. 1548, M.I.T., June 1995. Download from www.ai.mit.edu.

[111] P. Viola and W. M. Wells. Alignment by maximization of mutual information. *Fifth Int. Conf. on Computer Vision*, pages 16–23, 1995.

[112] Y. Wang and L. Staib. Elastic model based non–rigid registration incorporating statistical shape information. *Medical Image Computing and Computer–Assisted Intervention*, LNCS–1496:1162–1173, 1998.

[113] Yongmei Wang. *Statistical Shape Analysis for Image Segmentation and Physical Model–Based Non–Rigid Registration.* PhD thesis, Yale University, May 1999.

[114] J. Weese et al. 2D/3D registration of pre–operative ct images and intra–operative x–ray projections for image guided surgery. In H.Lemke et al., editors, *Comp. Assist. Rad. and Surg.*, pages 833–838. Elsevier Science, 1997.

[115] W. Wells. *Statistical Object Recognition.* A. I. technical report no. 1398, M.I.T., January 1993. Download from www.ai.mit.edu.

[116] W. M. Wells, P. Viola, and R. Kikinis. Multi-modal volume registration by maximization of mutual information. *2nd Ann. Int. Symp Med. Robot and Comput. Assisted Surg*, 1995.

[117] J.E. White et al. The influence of radiation therapy quality control on susvival, response and sites of relapse in oat cell carcinoma of the lung. *Cancer*, 50:1084–1090, 1982.

[118] R. Woods, S. Cherry, and J. Mazziotta. Rapid automated algorithm for aligning and reslicing PET images. *J. Comp. Assist. Tomography*, 16(4):620–633, 1992.

[119] G. Zubal, H. Tagare, et al. 3–D registration of intermodality medical images. *Ann. Int. Conf. IEEE Engng Med. Biol. Soc.*, 13:293–294, 1991. IEEE Comp Soc Press, Los Alamitos, CA.