Reconstruction and Restoration of PET Images.

Ph.D. Thesis

Peter Alshede Philipsen

LYNGBY 1998

IMM-PHD-1998-54

IMM

ISSN 0909–3192

Printed by IMM, Technical University of Denmark

Abstract

The subject of this Ph.D. thesis is a description of my work with reconstruction and restoration of Positron Emission Tomography (PET) images carried out at the Section of Digital Signal Processing, Department of Mathematical Modelling at the Technical University of Denmark in the period from March 1995 though September 1998.

The thesis is divided into 4 parts: The first chapter contains a short introduction to PET imaging. The second part, chapters 2 to 4, describe the PET scanner in mathematical terms by the use of the Radon transform, and the inversion of the Radon transform in both 2D and 3D. The third part, chapter 5, contains a description of a Bayesian signal processing approach, using Markov Random Fields as prior model for PET images for both reconstruction and restoration of PET images. The last part, chapter 6, contains two methods of restoration of PET using anatomical information from MR images i.e. MR is used in enhancement of PET.

Resumé (Abstract in Danish)

Emnet for denne Ph.D. afhandling er en beskrives af mit arbejde med rekontruktion og restorering af Positron Emission Tomography (PET) billeder, som er udført ved Sektionen for Digital Signal Behandling, Institut for Matematisk Modellering på Danmarks Tekniske Universitet i perioden fra marts 1995 til og med september 1998.

Afhandlingen indeholder 4 dele: Det første kapitel er en kort beskrivelse af PET og Magnetic Resonance (MR) efterfuldt af kapitel 2 til 4, som matematisk beskriver PET skanneren ved brug af Radon transformationen, samt inversion af denne i både to og tre dimensioner. Tredje del indeholder en beskrivelse af Bayesiansk signal behandlings metoder med Markov Random Field som prior model til både rekonstruktion af restorering af PET billeder. Den sidste del, kapitel 6, indeholder 2 metoder til PET restorering hvor anatomisk information fra MR billeder udnyttes til at forbedre PET kvaliteten.

Preface

This thesis presents the work at the Section for Digital Processing, Department of Mathematical Modelling at the Technical University of Denmark in the period March 1995 through September 1998 with Lars Kai Hansen and Claus Svarer as supervisors. Lars Kai Hansen is associate professor at the Section for Digital Processing and Claus Svarer is research associate at the National University Hospital in Copenhagen (Rigshospitalet).

The thesis contains six chapters. The first chapter gives a short introduction to PET imaging. Chapter 2 contains a description of the PET scanner in mathematical terms using the Radon transform and some of the properties of the Radon transform are given. In chapter 3 algorithms to 2D reconstruction of PET images, i.e. inversion of the Radon transform, are derived and examined on both phantoms and real data. In chapter 4 the basics of 3D reconstruction and the derivation of some of the standard inversion methods are presented. Chapter 5 contains a description of a Bayesian signal processing approach using Markov Random Fields as prior model for PET images for both reconstruction and restoration of PET images. The last chapter, chapter 6, contains two methods of restoration of PET using anatomical information from MR images i.e. MR is used in enhancement of PET.

Many people have been helpful during my work on the project and I would like to thank my supervisors for their supervision and support. I would also like to thank Dr. Chin-Tu Chin at the University of Chicago for the input in reconstruction and Markov Random Fields during my visit there and Peter Toft for introducing me to the Radon transform. Most of the work with standard 2D and 3D reconstruction has been inspired by his work or done in collaboration with him. I would like to thank Dr. Terry Jernigan at the Brain Image Analysis Laboratory for my inspiring stay at UCSD and Sarah Archibald for discussions about physical background in MR images on which my brain segmentation program presented in appendix A is based. I would also like to thank the people at the Section for Digital Processing, including my former roommates Mads Hintz-Madsen, Niels Mørch and Ulrik Kjems (with whom I had many inspiring discussions and a fruitful work with Markov Random Fields) who have been supportive all through the period.

Finally I want to thank my girlfriend Ulla without whose support this work would never have been done.

Lyngby, 30. September 1998.

Contents

Chapter 1

Introduction

In the work with studies of the living brain the Positron Emission Tomography (PET) scanners are one way to obtain in vivo images of the functional behavior of the human brain but other scanners exist such as Single Photon Emission Computed Tomography (SPECT), functional Magnetic Resonance Imaging (fMRI) and Electro-EncephaloGram (EEG). Other scanners can be used to obtain anatomical information about the brain such as Magnetic Resonance Imaging (MRI or just MR), Computerized Tomography (CT).

The images produced in PET and SPECT scanners are the results of induced radioactive tracers which are bound to the biochemical processes in the brain. Images produced by fMRI are based on local changes in nuclear-magnetic response of the brain, which again are (mostly) based on changes of the blood-flow in the brain. The EEG images are based on measured rhythmic electrical activity recorded from cerebral cortex and recorded by electrodes on the surface of the skull.

The anatomical image produced by a CT scanner is a measure of the different absorbency of X-ray (photons) in the different tissue types of the brain. The MR image is a measure of the local nuclear-magnetic response of the tissues in the brain.

The image modalities studied in this thesis are the PET images and anatomical information obtained from MR images. An example of a PET image can be seen in Figure (1.1) and a MR image of the same subject can be seen in Figure (1.2)

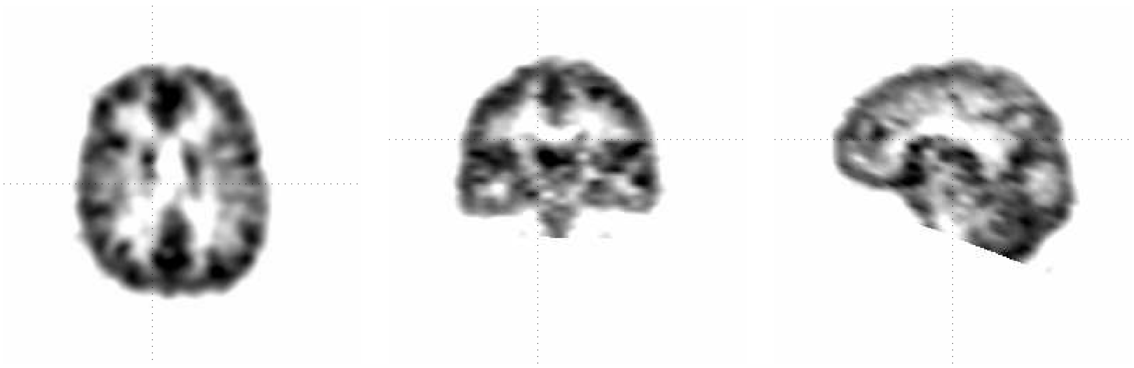

Figure 1.1: Slice of a PET image: Axial, Coronal and Sagittal slices are shown.

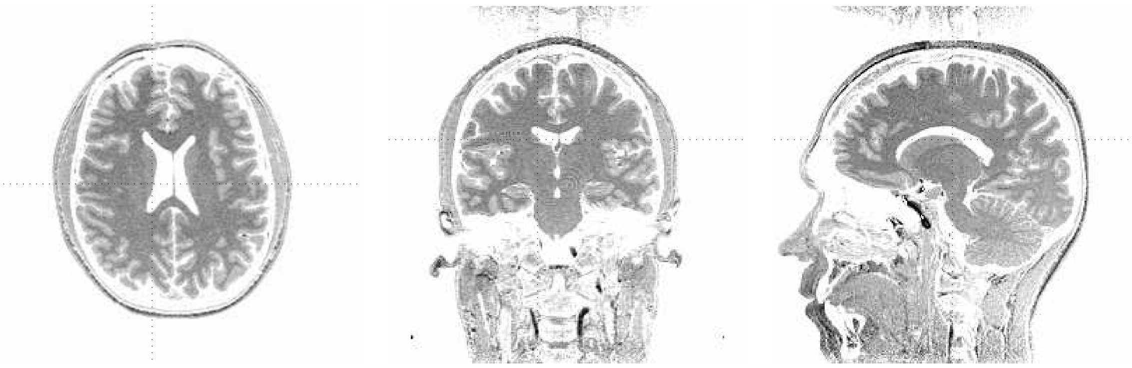

Figure 1.2: Slices of an MR image: Axial, Coronal and Sagittal are shown.

1.1 Positron Emission Tomography Imaging

Functional studies using PET images imply collaborations of many different fields of science: Physicists and chemists for the construction of the radioactive tracers, physicists and engineers for the construction of the PET scanner and development of algorithms for processing of the measured data into images, engineers and mathematicians for the statistical processing and finally neurologists for the interpretation of the final results.

In each of these fields research is performed to improve the quality of PET images and to obtain more accurate information in the final interpretation.

This thesis only investigates the algorithms for processing of the measured data (called sinogram) into images. To be able to develop the algorithms some basic knowledge about the PET scanner is needed.

The PET scanner consists of rings of detectors placed around the head of the subject. The detectors measure photon pairs emitted from the injected radioactive tracer and from these measurements an image of the distribution of the radioactive tracer can be computed; this computation is called reconstruction of the image. The Advance General Electric PET Scanner at the National University Hospital in Copenhagen contains 18 rings and can be seen in Figure (1.3).

The influence of non-ideal scintillation detectors, usually made of bismuth germanate (BGO) [DeGrado and et. al, 1994, Cho et al., 1993], the detection of random and scatter coincidence and how to overcome some of these problems are not a part of this thesis, but can be found in [Cho et al., 1993, Freifelder and et al., 1994, Chan and et al., 1996, DeGrado and et. al, 1994]

1.1.1 Reconstruction of PET Images

The mathematical background of 2D PET scanning is the Radon transform, and this transformation is described in Chapter 2. Standard inversion (reconstruction) algorithms of the Radon transform are presented in Chapter 3 together with experimental results of their performances. The mathematical background of 3D tomography is presented in Chapter 4 where some of the standard 3D inversion algorithms are derived.

The reconstructed PET images usually have a limited spatial resolution and signal to noise ratio. This is due both to the limited number of photon detectors and to the limited number of photon pairs measured. This limited measurement is partly caused by the

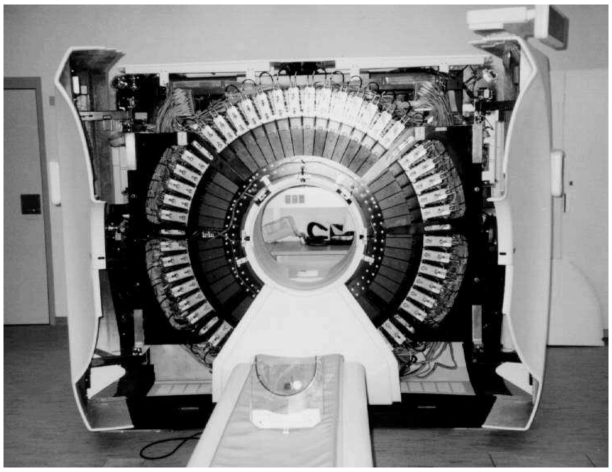

Figure 1.3: The Advance General Electric PET Scanner at the National University Hospital in Copenhagen

absorption in the brain tissue (approximately 86% of all photons passing 20 cm of brain are absorbed) and partly by photon scattering within the brain. The number of photon pairs cannot be raised since this requires that a higher radioactive dose is injected, which can harm the patient who absorbs most of the photons.

In the final interpretation of the PET scans additional smoothing is needed to increase the signal to noise ratio and remove undesired noise in the reconstructed images for improvement of the result of statistical analysis. A result of this additional smoothing is that some of the activation originally placed within the brain is replaced outside the brain. One approach to avoid this is to use anatomical information about the structures in brain and then only apply the smoothing within these boundaries. The anatomical information can be obtained from additional sources like MR images

Another approach is to incorporate structural information of high spatial resolution into the reconstruction process. This can be done since the distribution of the tracers follow the anatomical structures in the brain. Within the Bayesian framework it is possible to incorporate anatomical information with the goal of improving the reconstruction [Lipinski et al., 1997, Zhou et al., 1994, Gindi et al., 1995, Ouyang et al., 1994], and this is the background of the work in Chapter 5 and Chapter 6.

The interest in using statistical methods in image reconstruction and image restoration in a Bayesian framework is extensive [Li, 1995, Mumcuoglu et al., 1996, Chen et al., 1990, Zerubia and Chellappa, 1993, Johnson et al., 1991, Geman and Geman, 1984]. The different approaches usually include definition of an image model used as prior in the enhancement of the images. The focus in this thesis will be on Markov Random Fields as the prior model in the Bayesian framework, which allows discontinuities in the images and therefore has an easy way of incorporating the additional information from the MR images.

In this context only algorithms based on the Maximum A Posterior (MAP) estimate of the system is considered. The algorithms are tested for performance and stability in the choice of model parameters, Chapter 5. The MAP algorithms are compared to the standard reconstruction techniques in Chapter 3.

1.1.2 Structural Information from MR Images

In this thesis two ways of extracting information about the structural edges in MR images are explored: First, the use of active contour models which have the nature of adopting to the gradients in a image, and secondly the use of tissue segmentation of MR image. These methods are presented in Chapter 6.

One major problem in the use of MR information in PET images is the need for alignment of the two modalities. The results are based on the use of the Automated Image Registration (AIR) program package, [Woods and Cherry, 1993], which uses a 6 parameter rigid transformation to determine translation and rotation based on image ratio measures. Despite the very good performance problems may occur since the method requires MR images which only contains brain and is dependent on how this is done. To segment the MR image into brain/non-brain the semi-automated program described in appendix A is used.

Chapter 2

Definition of the Radon Transform - Reconstruction of PET

In this chapter a short introduction to and definition of the Radon transform will be given. The coupling between Computerized Tomography and the Radon transform is explained. The inversion of the Radon transform is given in the next chapter.

2.1 Basics of Computerized Tomography and Positron Emission Tomography

In this section a short introduction to Computerized Tomography will be given, but only in 2D to make things simple, but it is easily expanded to 3D.

Computerized tomography can be defined [Schalkoff, 1989] as the description of an N-dimensional object by a set of N-1 dimensional integrations. A 3D object will then be represented by a set of 2D projections. This will be plane integrals in different directions. In 2D the projections are line integrals.

In x-ray tomography the absorbent of x-rays is measured. Looking at a 1D object with a non-homogeneous absorbent coefficient μ, and having an initial intensity I_0 of the incident beam, the intensity at the point x will be

$$I(x) = I_0\, e^{-\int_0^x \mu\, ds} \tag{2.1}$$

Expanding Eq. 2.1 to 2D and making the assumption that the absorbent object, which is measured, is of limited size

$$I_L = I_0\, e^{-\int_L \mu(x,y)\, ds} \tag{2.2}$$

$$f_L \triangleq \ln\left(\frac{I_0}{I_L}\right) = \int_L \mu(x,y)\, ds \tag{2.3}$$

where f_L is the line integral of $\mu(x,y)$ along the line L. This only gives information about the attenuation along a single line, but if all lines through the object are covered it is possible to reconstruct $\mu(x,y)$ from these projections [Radon, 1917]. The line integral transformation is called the Radon transform, see Section 2.2. The line L is usually parameterized by (ρ, θ), section 2.2, and g Radon transformed is symbolized by $\breve{g}$.

2.1.1 Positron Emission Tomography

The goal of Positron Emission Tomography (PET) is to measure the functional activity inside an object, like the human body. The measured data is a photon pair which are generated by a positron-electron annihilation. The positrons are generated by radioactive nucleuses which are injected into the object. After the emission the positron travels a few millimeters before the annihilation with an electron. This phenomenon is called the point spread function and is dependent on the kinetic energy after the emission since the annihilation is most probable at low speed. After the annihilation the two photons will travel in (almost) direct opposite directions, but normally a small angle difference will occur [Herman, 1980]. The data is collected by a ring of photon-detectors around the object, in Figure (2.1) a schematic model of a PET scanner is shown.

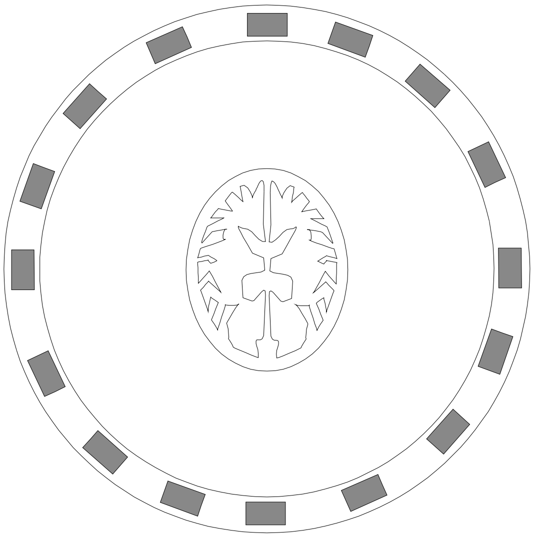

Figure 2.1: Simplified model of a PET scanner with 16 detectors

In brain imaging PET tracers like F-18, O-15 or C-11 are used. These tracers are produced by a cyclotron and incorporated in molecules like glucose, but O-15 is normally incorporated in water. If two detectors at the same time [1] detect photons, an annihilation has occurred along the line between the detectors. Within the scanning time T_e the two detectors will collect all the incidents coming from point sources along the line $L(\rho, \theta)$ between them, see Figure (2.2),

$$E(\rho, \theta) = T_e \int_{L(\rho,\theta)} A(x, y) ds \tag{2.4}$$

where the $A(x, y)$ is the spatially dependent emission activity. The activity is normally time dependent and therefore it is necessary to limit the scanning time. This is still an approximation since some of the photons are absorbed on their travel from the annihilation point to the detectors.

Consider two photons traveling from (x_0, y_0) as shown in Figure (2.3). The photon

[1] The photons travel with the speed of light and a small difference in arrival time of 2 nanoseconds is normal, but neglected in the following.

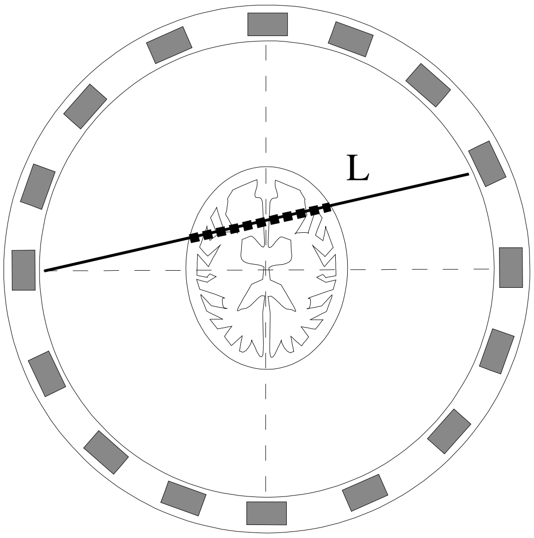

Figure 2.2: During scan time all incidents along line L are detected by the two associated detectors.

traveling along line piece L_1 would be detected with the probability P_1

$$P_1 = e^{-\int_{L_1} \mu(x,y)\, ds} \tag{2.5}$$

and similarly with the second photon

$$P_2 = e^{-\int_{L_2} \mu(x,y)\, ds} \tag{2.6}$$

The two photons travel independently, and the probability of detecting both photons is

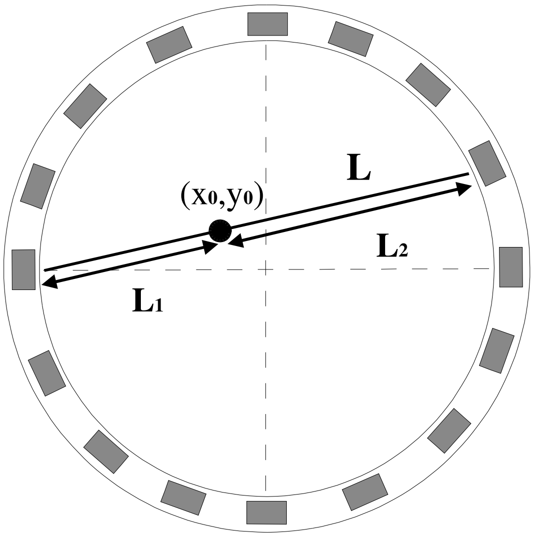

Figure 2.3: Travel path for two photons leaving point (x_0, y_0)

the product $P = P_1 P_2$

$$P_{L(\rho,\theta)} = e^{-\int_{L(\rho,\theta)} \mu(x,y)\, ds} \tag{2.7}$$

which is dependent on the line parameters (ρ, θ) so Eq. 2.4 turns into

$$E(\rho,\theta) = T_e \int_{L(\rho,\theta)} P_{L(\rho,\theta)} \, A(x,y) \, ds \tag{2.8}$$

$$= T_e \, e^{-\int_{L(\rho,\theta)} \mu(x,y) \, ds} \int_{L(\rho,\theta)} A(x,y) \, ds \tag{2.9}$$

$$= T_e \, \breve{A}(\rho,\theta) e^{-\int_{L(\rho,\theta)} \mu(x,y) \, ds} \tag{2.10}$$

where

$$\breve{A}(\rho,\theta) = \int_{L(\rho,\theta)} A(x,y) ds \tag{2.11}$$

and the factor $e^{-\int_{L(\rho,\theta)} \mu(x,y) \, ds}$ can be estimated by another scan using an external source which is rotated around the object, Figure (2.4).

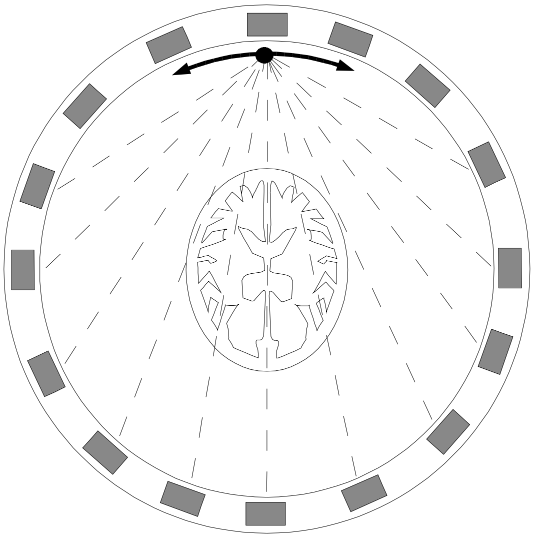

Figure 2.4: External source to estimate absorption coefficients

This scan is called the transmission scan T and will have the form

$$T(\rho,\theta) = T_t \, I_0(\rho,\theta) \, e^{-\int_{L(\rho,\theta)} \mu(x,y) \, ds} \tag{2.12}$$

where $I_0(\rho,\theta)$ is the strength of the external source. The strength is dependent on the line parameters due to absorption in the scanner materials and on the fact that the detectors are not ideal in practice, but will have different absorption profiles. To correct this a third scan, called a blank scan B, is measured without the object in the scanner and can be approximated by

$$B(\rho,\theta) = T_b \, I_0(\rho,\theta) \tag{2.13}$$

The formula for the Radon transformed of the emission activity can now be expressed by

$$\breve{A}(\rho,\theta) = \frac{E(\rho,\theta) \, B(\rho,\theta)}{T(\rho,\theta)} \frac{T_t}{T_e \, T_b} \tag{2.14}$$

which has some potential problems if the absorbency along a certain line is high and therefore $T(\rho,\theta)$ will be close to zero and could be dominated by noise [Toft, 1996, Cherry et al., 1991, DeGrado and et. al, 1994].

2.2 Theory of Radon Transform

The Radon transform is an integral transformation defined by the line integral of all lines in the plane. This means if f is a function defined in the region $D \in \mathbb{R}^2$ the Radon transform of f, symbolized by $\check{f}_L$, among all lines L spanning D is

$$\check{f}_L = \int_L f(x,y)ds \tag{2.15}$$

where ds is an incrementer in the length direction of L.

The Radon transform can be defined in many ways, but one of the most common in Computerized Tomography is [Deans, 1983, Toft, 1996]

$$\check{f}(\rho,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\,\delta(\rho - x\cos\theta - y\sin\theta)\,dx\,dy \tag{2.16}$$

The interpretation of Eq. 2.16 is an integration over the line $L : \rho - x\cos\theta - y\sin\theta = 0$ as seen in Figure (2.5), where ρ is the shortest distance from the origin of the coordinate system to the line, and θ is the angle to the ordinate.

Introducing a new coordinate system with axis rotated by the angle θ, labeled (ρ, s),

$$\begin{aligned} x &= \rho\cos\theta - s\sin\theta \\ y &= \rho\sin\theta + s\cos\theta \end{aligned}$$

the Radon transform in Eq. 2.16 can then have the explicit form

$$\check{f}(\rho,\theta) = \int_{-\infty}^{\infty} f(\rho\cos\theta - s\sin\theta, \rho\sin\theta + s\cos\theta)\,ds \tag{2.17}$$

This expresses directly that the Radon transform is a line integral transformation.

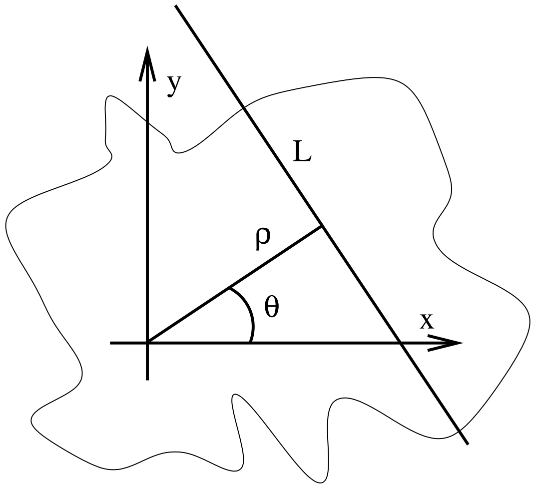

Figure 2.5: Definition of line parameters ρ and θ for line L

It should be noticed that $\check{f}(\rho,\theta) = \check{f}(-\rho,\theta+\pi)$, and this gives two obvious choices of parameter limitations to describe all lines

$$0 \leq \theta < 2\pi \quad \text{and} \quad 0 \leq \rho \leq \rho_{max} \tag{2.18}$$
or
$$0 \leq \theta < \pi \quad \text{and} \quad -\rho_{max} \leq \rho \leq \rho_{max} \tag{2.19}$$

In Computerized Tomography the parameter definition in Eq. 2.19 is normally used and will be used in this thesis.

From the definition Eq. 2.16 it can be seen that the Radon transform is a linear transformation, and several other properties can be found in [Deans, 1983, Toft, 1996, Jensen and Philipsen, 1995]. Only a few will be mentioned here.

2.2.1 Properties of the Radon Transform

To obtain information about the Radon transform the transformation of a point source in (x^1, y^1) is investigated.

$$\begin{aligned}
f(x,y) &= \delta(x-x^1)\,\delta(y-y^1) && (2.20)\\
&\Downarrow \\
\check{f}(\rho,\theta) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \delta(x-x^1)\,\delta(y-y^1)\delta(\rho - x\cos\theta - y\sin\theta)\,dx\,dy && (2.21)\\
&= \delta(\rho - x^1\cos\theta - y^1\sin\theta) && (2.22)\\
&= \delta(\rho - \rho^1\cos(\theta-\theta^1)) && (2.23)
\end{aligned}$$

where $x^1 = \rho^1\cos\theta^1$ and $y^1 = \rho^1\sin\theta^1$. This is a sinus curve in the Radon domain, see Figure (2.6) and Figure (2.7), and this has given the result of a Radon transform the name *sinogram*.

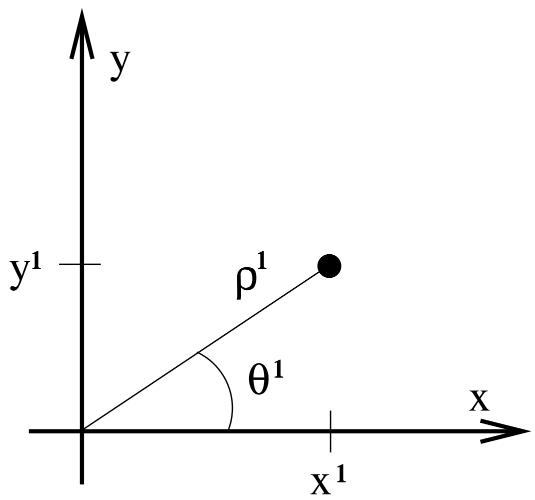

Figure 2.6: Point source

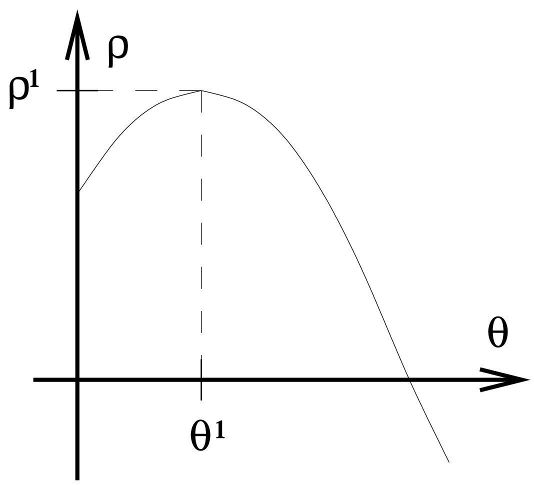

Figure 2.7: The Radon transformed, the sinogram, of a point source.

From Eq. 2.23 it is possible to obtain the Radon transform of any function $f(x, y)$

$$
\begin{aligned}
f(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x^1,y^1)\delta(x-x^1)\,\delta(y-y^1)\,dx^1\,dy^1 && (2.24)\\
&\Downarrow \\
\check{f}(\rho,\theta) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x^1,y^1)\delta(\rho-x^1\cos\theta-y^1\sin\theta)\,dx^1\,dy^1 && (2.25)
\end{aligned}
$$

From this we can derive

$$
\text{if } f(x,y)=0 \text{ for } \sqrt{x^2+y^2}>\rho_{max} \Rightarrow \check{f}(\rho,\theta)=0 \text{ for } \mid\rho\mid>\rho_{max} \qquad (2.26)
$$

Another important basic primitive is the Radon transform of a line with parameters (ρ^1, θ^1), in this case with infinite values at the line represented by the delta function

$$
\begin{aligned}
f(x,y) &= \delta(\rho^1-x\cos\theta^1-y\sin\theta^1) && (2.27)\\
&\Downarrow \\
\check{f}(\rho,\theta) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\delta(\rho^1-x\cos\theta^1-y\sin\theta^1)\delta(\rho-x\cos\theta-y\sin\theta)\,dx\,dy && (2.28)\\
&= \int_{-\infty}^{\infty}\frac{1}{\mid\sin\theta\mid}\delta(\rho^1-x\cos\theta^1-\frac{\rho-x\cos\theta}{\sin\theta}\sin\theta^1)\,dx\ ,\ 0<\theta<\pi && (2.29)\\
&= \int_{-\infty}^{\infty}\frac{1}{\mid\sin\theta\mid}\delta\left(\rho^1-\rho\frac{\sin\theta}{\sin\theta}-x(\cos\theta^1-\frac{\cos\theta\sin\theta^1}{\sin\theta})\right)dx && (2.30)\\
&= \frac{1}{\mid\sin\theta\mid}\frac{1}{\mid\cos\theta^1-\frac{\cos\theta\sin\theta^1}{\sin\theta}\mid} && (2.31)\\
&= \frac{1}{\mid\sin\theta\cos\theta^1-\cos\theta\sin\theta^1\mid}\ ,\ \theta\neq\theta^1\ ,\ 0<\theta<\pi && (2.32)\\
& && (2.33)
\end{aligned}
$$

and if $\theta = \theta^1$ e.q. $(\sin\theta = \sin\theta^1)$, beginning with Eq. 2.30,

$$
\begin{aligned}
\check{f}(\rho,\theta) &= \int_{-\infty}^{\infty}\frac{1}{\mid\sin\theta\mid}\delta\left(\rho^1-\rho\frac{\sin\theta}{\sin\theta}-x(\cos\theta^1-\frac{\cos\theta\sin\theta^1}{\sin\theta})\right)dx && (2.34)\\
&= \int_{-\infty}^{\infty}\frac{1}{\mid\sin\theta\mid}\delta(\rho^1-\rho)\,dx && (2.35)\\
&= \begin{cases} 0, & \text{if } \rho\neq\rho^1\\ \int_{-\infty}^{\infty}\delta(0)\,dx, & \text{if } \rho=\rho^1\end{cases} && (2.36)
\end{aligned}
$$

As seen this is an (infinite) peak at $\rho = \rho^1$ and $\theta = \theta^1$ and finite values in the rest of the parameter space.

2.2.2 Discrete Radon Transform

For most applications a discrete version of the Radon transform is needed. A straightforward method is to sample all parameters with equal spacing.

$$
\begin{array}{llll}
x & = x_m & = x_{min}+m*\Delta x, & m=0,1,2,\dots,M-1\\
y & = y_n & = y_{min}+n*\Delta y, & n=0,1,2,\dots,N-1\\
\rho & = \rho_r & = \rho_{min}+r*\Delta\rho, & r=0,1,2,\dots,R-1\\
\theta & = \theta_t & = \theta_{min}+t*\Delta\theta, & t=0,1,2,\dots,T-1
\end{array} \qquad (2.37)
$$

Apparently there are quite a few free parameters, but several of them are usually fixed. Some normal considerations could be equal sampling and image size in the x and y directions

$$\Delta x = \Delta y \text{ and } M = N \tag{2.38}$$

To obtain the smallest number of ρ samples the samples should be symmetrical around origin, and this implies

$$\begin{aligned} x_{min} &= -x_{max} = -\tfrac{1}{2}(M-1)\Delta x & (2.39)\\ y_{min} &= -y_{max} = x_{min} = -\tfrac{1}{2}(M-1)\Delta x & (2.40)\\ \rho_{min} &= -\rho_{max} = -\tfrac{1}{2}(R-1)\Delta\rho & (2.41)\end{aligned}$$

The angular sampling follows the interval definition in Eq. 2.19, and with linear sampling this gives

$$\theta_{min} = 0 \text{ and } \Delta\theta = \frac{\pi}{T} \tag{2.42}$$

This leaves $\Delta x, \Delta\rho, M, R$ and T to be determined. For a real PET scanner $\Delta\rho, R$ and T are given. Assuming that all parameters are set, a definition of a discrete Radon transform could be to discrete Eq. 2.16 directly

$$\begin{aligned} \check{f}(\rho_r, \theta_t) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y)\delta(\rho - x\cos\theta - y\sin\theta)\,dx\,dy & (2.43)\\ &\approx \Delta x\Delta y \sum_{m=0}^{M-1}\sum_{n=0}^{M-1} g(x_n, y_m)\delta_D(\rho_r - x_n\cos\theta_t - y_m\sin\theta_t) & (2.44)\end{aligned}$$

where δ_D is the discrete delta function, also called Kronecker's Delta function,

$$\delta_D(n) = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases} \tag{2.45}$$

and $g(x,y)$ is the discrete version of $f(x,y)$.

This implementation of the Radon transformation is time consuming due to the double sum. Implementing Eq. 2.17 instead requires two interpolations, which gives worse results, but a reformulation of Eq. 2.16 can reduce it to a single sum with only one interpolation

$$\begin{aligned} \check{f}(\rho_r, \theta_t) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y)\delta(\rho - x\cos\theta - y\sin\theta)\,dx\,dy & (2.46)\\ &= \int_{-\infty}^{\infty} g(x, \frac{\rho - x\cos\theta}{\sin\theta})\tfrac{1}{|\sin\theta|}\,dx & (2.47)\\ &\approx \frac{\Delta x}{|\sin\theta_t|}\sum_{n=0}^{M-1} g(x_n, [\frac{\rho_r - x_n\cos\theta_t}{|\sin\theta_t|}]), \text{ for } \theta \in]\tfrac{\pi}{4}; \tfrac{3\pi}{4}[& (2.48)\\ &= \frac{\Delta x}{|\sin\theta_t|}\sum_{n=0}^{M-1} g(n, [\tfrac{\rho_r}{\sin\theta_t} - n\cot\theta_t]), \text{ for } \theta \in]\tfrac{\pi}{4}; \tfrac{3\pi}{4}[& (2.49)\end{aligned}$$

or

$$\begin{aligned} &= \int_{-\infty}^{\infty} g(\frac{\rho - y\sin\theta}{\cos\theta}, y)\tfrac{1}{|\cos\theta|}\,dy & (2.50)\\ &\approx \frac{\Delta y}{|\cos\theta_t|}\sum_{m=0}^{M-1} g([\frac{\rho_r - y_n\sin\theta_t}{|\cos\theta_t|}], y_m), \text{ for } \theta \in [0; \tfrac{\pi}{4}] \cup [\tfrac{3\pi}{4}; \pi[& (2.51)\end{aligned}$$

where $[\cdot]$ can be rounding to the nearest neighbor or linear interpolation.

2.2.3 Linear Algebra Formalism of the Radon Transform

Since the Radon transform is a linear transform it can be described in standard linear algebra. Wrapping the image f into a vector x and similarly in the Radon domain $\check{f} \rightsquigarrow b$,

$$\check{f} = \mathcal{R}f \tag{2.52}$$

$$\Downarrow$$

$$b = \mathbf{A}x \tag{2.53}$$

each of the elements in $\mathbf{A}$ can be determined by examination of Eq. 2.44. Each element of $\mathbf{A}$, $a_{i,j}$, can be calculated in different ways. The simplest derivation is to set the $a_{i,j}$ elements to $\Delta x \Delta y$ if the corresponding pixel (j) is crossed by the line (i), Figure (2.8), $a_{i,j} = \Delta x \Delta y \delta_D(\rho_r - x_n \cos\theta_t - y_m \sin\theta_t)$ for example with $i = r + Rt$ and $j = n + Nm$.

A better method is to make $a_{i,j}$ proportional to the length of the line through the pixel. This is like the approximation in Eq. 2.17 with $a_{i,j} = \Delta s$.

Alternatively δ_D can be replaced with other approximations like the ray driven method [Deans, 1983], where the line is assumed to have a finite width ($\Delta\rho$) and $a_{i,j}$ is calculated as the common area of the square ($\Delta x \times \Delta y$) representing the pixel and the line, Figure (2.9).

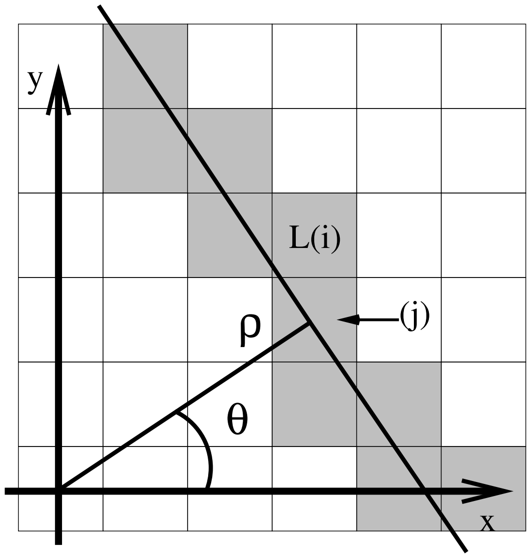

Figure 2.8: The element $a_{i,j} = \Delta x \Delta y$ if the corresponding pixel (j) is crossed by the line (i)

Other interpolation methods have been investigated such as the sinc interpolation [O'Sullivan et al., 1993, Lewitt, 1983, Guedon et al., 1992], but these methods require more computer time.

2.3 Analytical Radon Transformation

In this section the Radon transformed of a few primitives will be presented; more examples are given in [Deans, 1983, Jensen and Philipsen, 1995, Toft, 1996, Jain, 1989]. These primitives can be used to generate more complex phantoms used in the evaluation of the different inversion methods.

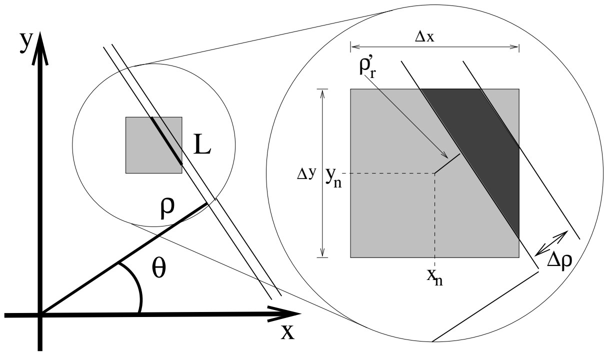

Figure 2.9: The line has a finite width ($\Delta\rho$) and $a_{i,j}$ is calculated as the common area of the square ($\Delta x \times \Delta y$) representing the pixel and the line

2.3.1 The Unit Circle

The unit circle is defined as

$$g(x,y) = \begin{cases} 1, & \text{for } x^2 + y^2 \leq 1 \\ 0, & \text{otherwise} \end{cases} \tag{2.54}$$

Since the unit circle is independent of rotation around origin, the Radon transformed will be independent of θ and can be calculated with $\theta = 0$

$$\begin{aligned} \breve{g}(\rho,0) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y)\delta(\rho - x)\,dx\,dy && (2.55)\\ &= \int_{-1}^{1}(\int_{-\sqrt{1-x^2}}^{\sqrt{1-x^2}} 1\,dy)\,\delta(\rho - x)\,dx && (2.56)\\ &= \int_{-1}^{1} 2\sqrt{1-x^2}\delta(\rho - x)\,dx && (2.57)\\ &= 2\sqrt{1-\rho^2} \text{ for } |\rho| \leq 1 && (2.58)\end{aligned}$$

This can also be expressed as the length of the line crossing the unit circle.

$$\breve{g}(\rho,\theta) = \begin{cases} 2\sqrt{1-\rho^2} & \text{for } |\rho| \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{2.59}$$

The sinogram of the unit circle, see Figure (2.10), can be seen in Figure (2.11).

Using the unit circle as base and incorporating scaling, shifting and rotating properties more complex phantoms can be constructed like the Shepp-Logan phantom in Figure (2.12) with the corresponding sinogram in Figure (2.13) [Jain, 1989].

2.3.2 The Gaussian Bell

Another simple primitive is the Gaussian bell centered at $(0,0)$

$$g(x,y) = \exp(-x^2 - y^2) \tag{2.60}$$

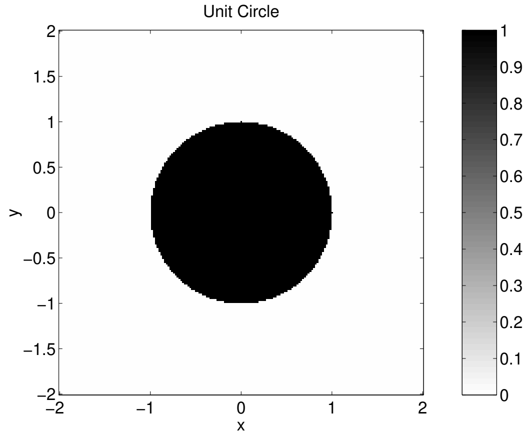

Figure 2.10: Unit circle with constant value of 1

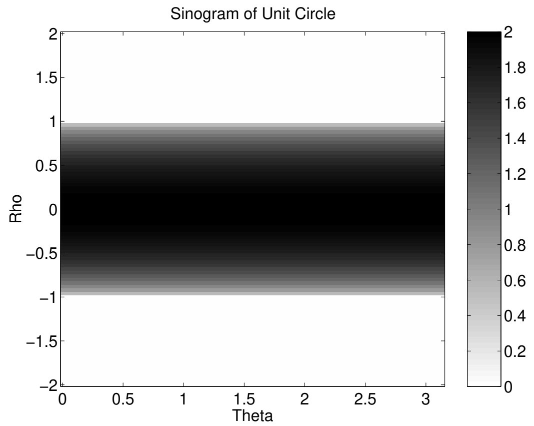

Figure 2.11: Sinogram of unit circle

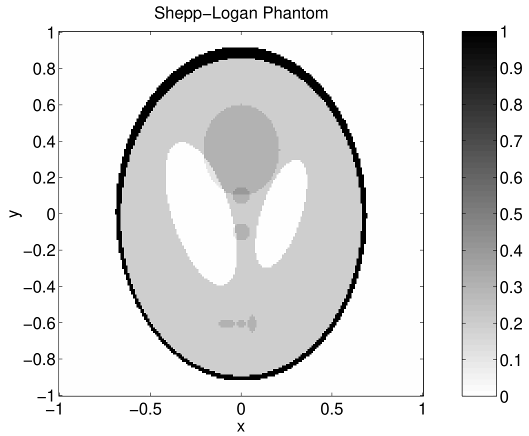

Figure 2.12: Shepp-Logan phantom

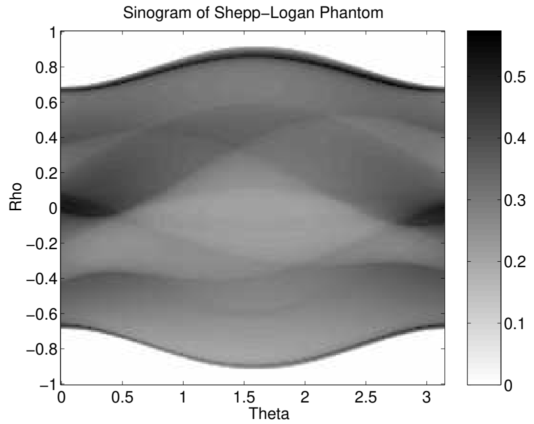

Figure 2.13: Sinogram of Shepp-Logan phantom

Again the Radon transform will be independent of rotation and

$$\breve{g}(\rho,\theta) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-x^2-y^2}\delta(\rho - x\cos\theta - y\sin\theta)\,dx\,dy \tag{2.61}$$

$$= \int_{-\infty}^{\infty} e^{-y^2}\int_{-\infty}^{\infty} e^{-x^2}\delta(\rho - x\cos\theta - y\sin\theta)\,dx\,dy \tag{2.62}$$

$$= \int_{-\infty}^{\infty} e^{-y^2}\,\frac{1}{|\cos\theta|}\,e^{-(\frac{\rho - y\sin\theta}{\cos\theta})^2}\,dy \tag{2.63}$$

$$= \sqrt{\pi}\,e^{-\rho^2} \tag{2.64}$$

and it can be seen that the Radon transformed is independent of θ. The sinogram of the Gauss bell, Figure (2.14), can be seen in Figure (2.15).

The problem with the Radon transformed of the Gauss Bell is that it is not bounded and will never exist in practice, but an approximation can be used by limiting the width of the gauss bell.

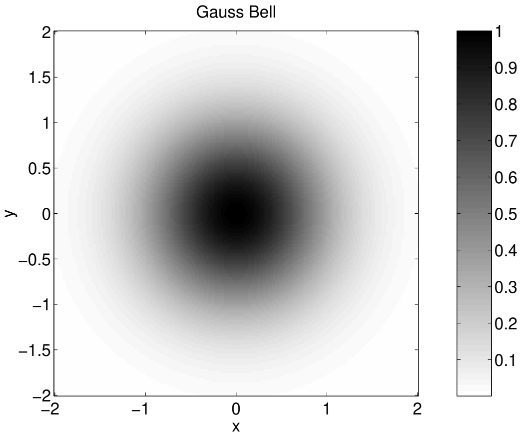

Figure 2.14: Gauss bell

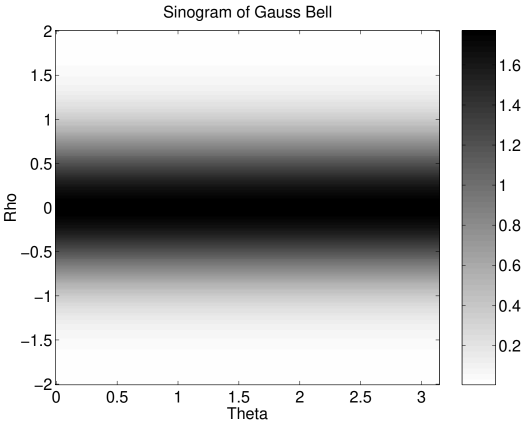

Figure 2.15: Sinogram of Gaussian bell

Chapter 3

Inversion of the Radon Transform

The inversion of the Radon transform can be categorized into *Direct Methods* and *Iterative Methods* [Jensen and Philipsen, 1995, Deans, 1983]. The Direct Methods are based on the relationship between the Radon transform and the Fourier transform and therefore also called *Fourier Methods*. The Iterative Methods are based on the linearity of the Radon transform and linear algebra methods to inverse the transform.

3.1 Direct Inversion Methods

In the following a few of the most common direct 2D inversion methods are described.

3.1.1 The Fourier Slice Method

The Fourier Slice Theorem is also known as the Central Slice Theorem [Mersereau, 1973, Dudgeon and Merserau, 1984] and is based on the 2D Fourier transform of $g(x, y)$

$$G(k_x, k_y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x, y)\, e^{-j2\pi(k_x x + k_y y)}\, dxdy \tag{3.1}$$

and the inverse Fourier transform

$$g(x, y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(k_x, k_y)\, e^{j2\pi(k_x x + k_y y)}\, dk_x dk_y \tag{3.2}$$

Using polar parameters in the frequency domain

$$\begin{pmatrix} k_x \\ k_y \end{pmatrix} = \nu \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} \tag{3.3}$$

and inserting in Eq. 3.1 gives

$$\begin{aligned}
G(\nu\cos\theta, \nu\sin\theta) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x, y) e^{-j2\pi\nu(x\cos\theta + y\sin\theta)}\, dxdy \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x, y) \left[\int_{-\infty}^{\infty} e^{-j2\pi\nu\rho}\ \delta(\rho - x\cos\theta + y\sin\theta)\, d\rho\right] dxdy \\
&= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x, y)\delta(\rho - x\cos\theta + y\sin\theta)\, dxdy\right] e^{-j2\pi\nu\rho}\, d\rho \\
&= \int_{-\infty}^{\infty} \breve{g}(\rho, \theta)\, e^{-j2\pi\nu\rho}\, d\rho \qquad (3.4)
\end{aligned}$$

This means that the inverse of the Radon transform can be calculated by a one-dimensional Fourier transform in the $\rho \to \nu$ direction to obtain the 2D Fourier spectrum $G(\nu\cos\theta, \nu\sin\theta)$. From this a 2D Fourier inversion , Eq. 3.2, can be be applied to obtain $g(x,y)$. It should be noticed that the theorem can be used to calculate the forward Radon transform.

In implementation of the Fourier Slice inversion the 2D interpolation in the spectrum is the major drawback [O'Sullivan et al., 1993]. The Fourier transformation can be calculated using Fast Fourier Transform (FFT), i.e. expanding the image to nearest Radix-2 and then use a Radix-2 FFT. The interpolation can be calculated fast using a nearest neighbor interpolation, but the results are poor [Trussell et al., 1987, Jensen and Philipsen, 1995]. A more time consuming approach, with fewer expected artifacts in the image, is to use bilinear interpolations from the nearest four point, Figure (3.1).

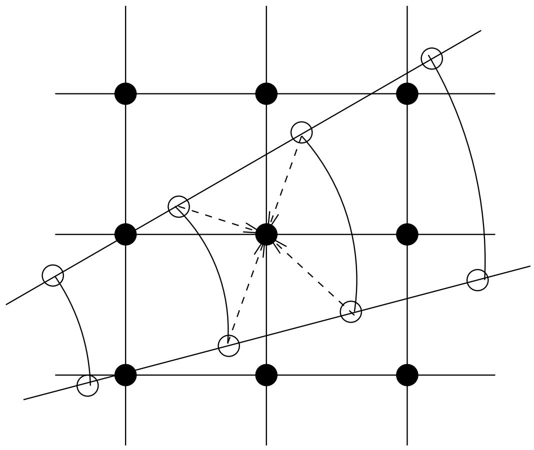

Figure 3.1: Interpolation from the 4 nearest neighbors in the polar coordinate.

This interpolation method does not use all the points near the origin in the polar form of the spectrum, but higher order interpolation filters or non-linear sampling techniques can be used to obtain better results [Jensen and Philipsen, 1995, Magnusson, 1989, Edholm and G. T. Herman, 1987], but with an increase in the computational load.

3.1.2 Filtered Backprojection

The Filtered Backprojection scheme is the most popular direct method and can be derived from Eq. 3.2 using polar coordinates

$$\begin{aligned}
g(x,y) &= \int_0^{2\pi}\int_0^{\infty} \nu G(\nu\cos\theta, \nu\sin\theta) e^{j2\pi\nu(x\cos\theta+y\sin\theta)}\, d\nu\, d\theta \\
&= \int_0^{\pi}\int_{-\infty}^{\infty} |\nu| G(\nu\cos\theta, \nu\sin\theta) e^{j2\pi\nu(x\cos\theta+y\sin\theta)}\, d\nu\, d\theta \\
&= \int_0^{\pi}\int_{-\infty}^{\infty} |\nu| \left[\int_{-\infty}^{\infty} \breve{g}(\breve{\rho},\theta) e^{-j2\pi\nu\breve{\rho}}\, d\breve{\rho}\right] e^{j2\pi\nu(x\cos\theta+y\sin\theta)}\, d\nu\, d\theta \qquad (3.5)\\
&= \int_0^{\pi}\int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} |\nu| \left(\int_{-\infty}^{\infty} \breve{g}(\breve{\rho},\theta) e^{-j2\pi\nu\breve{\rho}}\, d\breve{\rho}\right) e^{j2\pi\nu\tilde{\rho}}\, d\nu\right] \delta(\tilde{\rho} - x\cos\theta - y\sin\theta)\, d\tilde{\rho} d\theta
\end{aligned}$$

This is normally written in two parts: A filtering part, inside the brackets [], followed by an integration part

$$\tilde{\breve{g}}(\rho,\theta) = \int_{-\infty}^{\infty} |\nu| \left(\int_{-\infty}^{\infty} \breve{g}(\breve{\rho},\theta) e^{-j2\pi\nu\breve{\rho}}\, d\breve{\rho} \right) e^{j2\pi\nu\tilde{\rho}}\, d\nu \tag{3.6}$$

$$= \mathrm{IFT}_{\nu\to\rho}\{\, |\nu|\; \mathrm{FT}_{\rho\to\nu}\{\breve{g}(\rho,\theta)\}\} \tag{3.7}$$

$$g(x,y) = \int_0^{\pi} \int_{-\infty}^{\infty} \tilde{\breve{g}}(\rho,\theta)\delta(\tilde{\rho} - x\cos\theta - y\sin\theta)\, d\tilde{\rho}\, d\theta \tag{3.8}$$

$$= \int_0^{\pi} \tilde{\breve{g}}(x\cos\theta + y\sin\theta,\theta)\, d\theta \tag{3.9}$$

The operation in Eq. 3.9 is called backprojection and performs an integration along a sine curve in the sinogram. The backprojection is related to the adjoint Radon transform, [Deans, 1983]: the adjoint Radon transform is two times the backprojection operator.

The filtering part is a 1D high pass filter $|\nu|$ for each of the angles, but can also be expressed by convolution [Deans, 1983, Toft, 1996, Jain, 1989]

$$\tilde{\breve{g}}(\rho,\theta) = \frac{\partial\breve{g}(\rho,\theta)}{\partial\rho} * \frac{1}{2\pi^2\rho} \tag{3.10}$$

where $*$ is the one-dimensional convolution in the ρ direction.

To avoid enhancement of noise in the sinogram different windowing functions have be added to the filter part to stabilize the algorithm. Some of the filters in [Deans, 1983, Jain, 1989] are given, normalized to the allowed upper frequency $\nu_{upper} = \frac{1}{2\Delta\rho}$,

The Ram-Lak Filter is just a cropped version of the $|\nu|$

$$H(\nu)_{Ram-Lak} = |\nu|, \text{ for } |\nu| \leq \nu_{upper} \tag{3.11}$$

and zero elsewhere.

The Generalized Hamming Filter is

$$H(\nu)_{Hamming} = |\nu| \left(\alpha + (1-\alpha)\cos\left(\frac{\pi\nu}{\nu_{upper}}\right)\right), \text{ for } |\nu| \leq \nu_{upper} \tag{3.12}$$

with typical values of α just above 0.5

The Hann Filter is a special case of the generalized Hamming with $\alpha = 0.5$

$$H(\nu)_{Hann} = |\nu| \frac{1}{2}\left(1 + \cos\left(\frac{\pi\nu}{\nu_{upper}}\right)\right), \text{ for } |\nu| \leq \nu_{upper} \tag{3.13}$$

and zero elsewhere.

The Shepp-Logan Filter is $|\nu|$ multiplied with a sinc window

$$H(\nu)_{Shepp-Logan} = |\nu| \frac{\sin\left(\frac{\pi\nu}{2\,\nu_{upper}}\right)}{\frac{\pi\nu}{2\,\nu_{upper}}}, \text{ for } |\nu| \leq \nu_{upper} \tag{3.14}$$

and zero elsewhere.

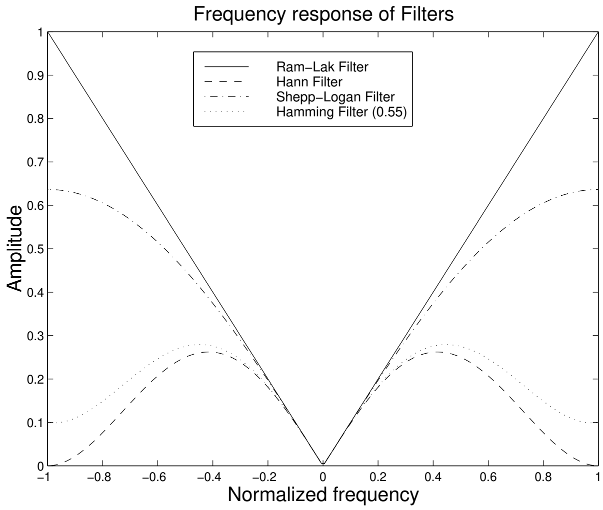

Figure 3.2: The Ram-Lak, Shepp-Logan,Hann and Generalized Hamming with $\alpha = 0.55$.

Other filters like the **Stochastic Filter** in [Jain, 1989] suggest that the filter should adopt the actual characteristic of the noise.

Some of the filters can be seen in Figure (3.2), all with normalized upper frequency $\nu_{upper} = 1$. The actual cutoff frequency depends on the noise level in the observed sinogram.

All the filters have a built-in problem by setting the mean of $\tilde{\breve{g}}(\rho, \theta)$ to zero. Since

$$\int_{-\infty}^{\infty} \breve{g}(\rho, \theta)\, d\rho = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y)\, dx\, dy \tag{3.15}$$

for each angle θ which represents an integration over the full image. The mean value of the reconstructed image can be estimated using the average of the integrated sinogram lines noticing the proper sampling parameters.

3.1.3 Filtering After Backprojection

The inversion of the Radon transform can also be done as a Backprojection followed by a filtering [Deans, 1983, Jain, 1989]. This scheme is also called Filter of Backprojections.

The filter can be determined by examination of a single point source

$$\begin{aligned} g(x, y) &= \delta(x - x^*)\delta(y - y^*) && (3.16) \\ &\Downarrow \\ \breve{g}(\rho, \theta) &= \delta(\rho - x^* \cos\theta - y^* \sin\theta) && (3.17) \end{aligned}$$

from Eq. 2.23. Inserting in Eq. 3.9 gives

$$\begin{aligned}
\tilde{g}(x,y) &= \int_0^{\pi} \delta(x\cos\theta + y\sin\theta - x^*\cos\theta - y^*\sin\theta)\, d\theta && (3.18)\\
&= \int_0^{\pi} \delta((x-x^*)\cos\theta + (y-y^*)\sin\theta)\, d\theta && (3.19)\\
&= \left.\frac{1}{|-(x-x^*)\sin\theta + (y-y^*)\cos\theta|}\right|_{(x-x^*)\cos\theta+(y-y^*)\sin\theta=0} && (3.20)\\
&= \frac{1}{|(x-x^*)\sin\arctan(\frac{y-y^*}{x-x^*}) + (y-y^*)\cos\arctan(\frac{y-y^*}{x-x^*})|} && (3.21)\\
&= \frac{1}{\sqrt{(y-y^*)^2+(x-x^*)^2}} = g(x,y) ** \frac{1}{\sqrt{y^2+x^2}} && (3.22)
\end{aligned}$$

where $**$ is a two-dimensional convolution. The filtering can also be expressed in the frequency domain. Using $h(x,y) = \frac{1}{\sqrt{y^2+x^2}}$ and the 2D Fourier transform of Eq. 3.22

$$\begin{aligned}
\tilde{G}(k_x,k_y) &= G(k_x,k_y)H(k_x,k_y) \Rightarrow && (3.23)\\
G(k_x,k_y) &= \frac{\tilde{G}(k_x,k_y)}{H(k_x,k_y)} && (3.24)
\end{aligned}$$

where $H(k_x,k_y) = \frac{1}{\sqrt{k_y^2+k_x^2}}$ is the Fourier transform of $h(x,y)$, and it can be seen that $\frac{1}{H(k_x,k_y)}$ is a 2D high pass filter. The inversion of $\breve{g}(\rho,\theta)$ can be done like this

$$\begin{aligned}
\tilde{g}(x,y) &= \int_0^{\pi} \breve{g}(x\cos\theta + y\sin\theta, \theta)\, d\theta && (3.25)\\
\tilde{G}(k_x,k_y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \tilde{g}(x,y)e^{-j2\pi(k_x x+k_y y)}\, dx dy && (3.26)\\
g(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \sqrt{k_y^2+k_x^2}\; \tilde{G}(k_x,k_y)e^{j2\pi(k_x x+k_y y)}\, dk_x\, dk_y && (3.27)
\end{aligned}$$

This algorithm is very similar to the Filtered Backprojection, but is usually slower because a 2D filtering is used instead of a 1D filtering in Filtered Backprojection.

Like the 1D filter kernel in Filtered Backprojection, the filter kernel $\sqrt{k_y^2+k_x^2}$ has problems with enhancement of high frequency noise. The 2D filter kernel can also be weighted with 2D versions of the window functions used for Filtered Backprojections.

However it is not a correct inversion scheme since the mean value is set at zero because the filter value in $(k_x,k_y) = (0,0)$ equals zero. This can be corrected by making an assumption that the level in the image e.g. in PET outside the head will be zero, or by using the estimate based on Eq. 3.15.

3.2 Linear Algebra Inversion Methods

As shown in section 2.2.2 the Radon transform can be expressed in standard linear algebra representation and therefore standard linear algebra inversion methods can be used to overcome some of the problems of the direct methods.

- The mathematical background of linear algebra is very well developed.

- The methods can be used both for 2D and 3D.
- Missing angles and different geometries can be included in the matrix formalism, like non-zero detector geometry and detector sensibility.
- The system matrix **A** is not quadric, so standard methods like the Singular Value Decomposition [Toft, 1996] is not usable.
- The system matrix **A** is almost singular, i.e. has small singular values, so the linear algebra formalism of the reconstruction is ill-posed. This is due to the fact that some image values are very determined, center values, and others are under determined near the edge of the selected region, and regularization is often needed.
- The system matrix **A** is normally very large, and the inversion is very computer demanding.
- The system matrix is sparse and because only a few lines interact with a single pixel, e.g. if only one line crosses a pixel for each angle, there will only be $\frac{1}{TR}$ non-empty elements in **A**.

Before turning to the inversion schemes a few things about the connection between the Radon transform and the matrix formalism should be noticed.

The Radon transform of the discrete image $g(m,n) = \mathbf{x}$ is transformed into $\breve{g}(r,t) = \mathbf{b}$

$$\mathbf{b} = \mathbf{A}\mathbf{x} \tag{3.28}$$

Some iterative algorithms like ART and MART, see the following subsections, use the transform of the image $\mathbf{x}$ into a single sample in the Radon domain

$$b_i = \mathbf{a}_i^T \mathbf{x} \tag{3.29}$$

where $\mathbf{a}_i^T$ is the i row in the matrix $\mathbf{A}$. Another often used operator is the transpose of the matrix $\mathbf{A}$, which is the adjoint operator

$$\tilde{\mathbf{x}} = \mathbf{A}^T \mathbf{b} \tag{3.30}$$

The adjoint of the Radon transform is two times the Backproject operator, Eq. 3.9.

3.2.1 Algebraic Reconstruction Technique

The ART (Algebraic Reconstruction Technique) was the first technique used to reconstruct a tomographic image, and has been widely used since.

The basic operation required in ART is Eq. 3.29, and the updating scheme is formulated as follows for iteration k

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \frac{b_i - \mathbf{a}_i^T \mathbf{x}^{(k)}}{\mathbf{a}_i^T \mathbf{a}_i} \mathbf{a}_i \tag{3.31}$$

This equation is fulfilled in iteration $k+1$ since

$$\mathbf{a}_i^T \mathbf{x}^{(k+1)} = \mathbf{a}_i^T \mathbf{x}^{(k)} + \frac{b_i - \mathbf{a}_i^T \mathbf{x}^{(k)}}{\mathbf{a}_i^T \mathbf{a}_i} \mathbf{a}_i^T \mathbf{a}_i = b_i \tag{3.32}$$

which means that in the k'th update the reconstructed image $\mathbf{x}$ is modified so that the Radon transformed of $\mathbf{x}$, produces the correct result in sample b_i. The problem of choosing i is not trivial, since the obvious choice $i = k\,MOD\,I$ [Jain, 1989] is not very good [Herman and Meyer, 1993, Guan and Gordon, 1994]. But it is possible to choose i from a random sampling of a uniform distribution. As seen in Eq. 3.31 ART can simply be implemented since the denominator $\mathbf{a}_i^T\mathbf{a}_i$ can be calculated in an initialization step. Each iteration only requires a vector product and is very fast, but the gain is limited, so for comparison with other algorithms what is called one iteration includes a full loop through all rows, therefore the actually number of iterations is a factor of I higher than stated.

One way to speed up the convergence [Herman, 1980] is to use a weight factor λ_k

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \lambda_k \frac{b_i - \mathbf{a}_i^T\mathbf{x}^{(k)}}{\mathbf{a}_i^T\mathbf{a}_i}\,\mathbf{a}_i \tag{3.33}$$

λ_k can be chosen to be a simple function of k like a linear or exponential decay [Herman and Meyer, 1993].

This gives no guarantee of obtaining non-negative solutions, but constrains can easily be added [Censor, 1983]. A simple solution is to limit the output of each iteration with a lower and an upper bound, but this is very time consuming. A simple speedup is to only invoke the constraint after K iterations. One advantage is that it is possible to make the constraint spatially dependent.

An initialization guess $\mathbf{x}^{(0)}$ to the solution can be chosen as zero or as the result of a fast direct method. Using a direct method as a starting point would make the algorithm converge faster, but the result will be biased by the direct method. Another starting guess could be a constant [Kaufmann, 1987]

$$x_j^0 = \frac{\sum_{i=1}^{I} b_i}{\sum_{i=1}^{I}\sum_{j=1}^{J} a_{i,j}} \text{ for all } j \tag{3.34}$$

but this requires a special calculation of the denominator.

3.2.2 Multiplicative Algebraic Reconstruction Technique

Another inversion technique in the literature is the Multiplicative Algebraic Reconstruction Technique (MART) [Censor, 1983], but it is rarely used in practical reconstruction. The key idea is to maximize the entropy of the solution

$$-\sum_{j=1}^{J} x_j \log x_j \tag{3.35}$$

under the constraint

$$b_i = \mathbf{a}_i^T\mathbf{x}^{(k)} \text{ and } x_j \geq 0 \;\forall j \tag{3.36}$$

The updating algorithm looks like

$$x_j^{(k+1)} = x_j^{(k)} \left(\frac{b_i}{\mathbf{a}_i^T\mathbf{x}^{(k)}}\right)^{\lambda_k a_{i,j}} \tag{3.37}$$

As it can be seen this updating formula has a problem since $\mathbf{a}_i^T\mathbf{x}^{(k)}$ can be zero. The factor λ_k is a relaxation parameter $0 \leq \lambda_k \leq 1$. The initialization is to set all $x_j = e^{-1}$ for all j. Other versions of multiplicative algebraic reconstructions can be found in [Pierro, 1992].

3.2.3 Expectation Maximization

The Expectation Maximization (EM) algorithm differs from the previous inversion algorithms by using a statistical approach to deal with the structure of the noise. In PET imaging areas of interest have only a few detected incidents. The described algorithm is the Maximum Likelihood Expectation Maximization (ML-EM) [Shepp and Krustal, 1978, Vardi et al., 1985]. The assumption is that the measured data is uncorrelated and is generated by a Poisson process. This is the ideal underlying process of the emission tomography, but since attenuation is not modeled the scheme is only an approximation.

The idea is to maximize the likelihood

$$L(\mathbf{x}) = P(\mathbf{b}|\mathbf{x}) = \prod_{i=1}^{I} \frac{(b_i^*)^{b_i}}{b_i!} e^{-b_i^*} \tag{3.38}$$

where b_i^* is the mean value of the Poisson process generating b_i. b_i^* will be estimated under the assumption that

$$\mathbf{b}^* = \mathbf{A}\mathbf{x} \tag{3.39}$$

and under the constraint that x_j is non-negative. The system matrix is built of transition probabilities normalized row wise

$$1 = \sum_{i=1}^{I} a_{i,j} \tag{3.40}$$

which in the sense of PET means that a detected photon pair is emitted from within the observed region. An updating formula can be derived from Eq. 3.38 by setting the derivatives of the log Likelihood to zero

$$x_j^{(k+1)} = x_j^{(k)} \sum_{i=1}^{I} \frac{a_{i,j} b_i}{\mathbf{a}_i^T \mathbf{x}^{(k)}} \tag{3.41}$$

This is the most common formulation, but [Carson and Lange, 1985] have another formulation which does not need the normalization

$$x_j^{(k+1)} = \frac{x_j^{(k)}}{\sum_{i=1}^{I} a_{i,j}} \sum_{i=1}^{I} \frac{a_{i,j} b_i}{\mathbf{a}_i^T \mathbf{x}^{(k)}} \tag{3.42}$$

The EM algorithm is very computation demanding, which can be seen by expanding Eq. 3.42 into four steps

$$\mathbf{b}^r = \mathbf{A}\mathbf{x}^{(\mathbf{k})} \tag{3.43}$$

$$b_i^e = \frac{b_i}{b_i^r}, \text{ for all } i \tag{3.44}$$

$$\mathbf{x}^b = \mathbf{A}^T \mathbf{b}^e \tag{3.45}$$

$$x_j^{(k+1)} = x_j^{(k)} \frac{x_j^b}{s_j}, \text{ for all } j \tag{3.46}$$

where $s_j = \sum_{i=1}^{I} a_{i,j}$ which can be calculated once.

The first step, Eq. 3.43, is a forward Radon transform of the current estimate $\mathbf{x}^{(\mathbf{k})}$ to the solution. Next is the quotient between the estimated sinogram and the observed sinogram. Then the error is backprojected to the image domain as a factor in Eq. 3.46. This means that each EM step requires a full forward Radon transform and a backprojection. As an initial guess Eq. 3.34 can be used.

3.2.4 Bayesian Approach

Another approach is to constrain the solution with extra prior assumption about the image to be reconstructed. A Bayesian description of the system can be used as a basis for an inversion scheme, with the used prior as a Markov Random Field model as described in Chapter 5. No additional information like anatomical information from MR images will be used in this chapter for a more direct comparison. The method for finding the MAP estimate of the system is the Mean Field Annealing schedule, see Section 5.4.3, and is here called MRF-MFA.

3.3 2D Results

The test results in sections are calculated by a modified version of the freely available software packages developed [Jensen and Philipsen, 1995, Toft, 1996]. The quality of the individual reconstruction methods is both visual inspection and a calculated error measure.

The error measure used is the normalized second order norm called L_2

$$L_2 = \frac{\| \mathbf{g}_{estimated}(x_n, y_m) - \mathbf{g}_{reference}(x_n, y_m) \|_2}{\| \mathbf{g}_{reference}(x_n, y_m) \|_2} \tag{3.47}$$

where the 2-norm is calculated as

$$\| \mathbf{g}(x_n, y_m) \|_2 = \sqrt{\frac{1}{NM} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} g(x_n, y_m)^2} \tag{3.48}$$

This error measure is always positive, and for a 100% correct measure L_2 equals zero.

The primary test images will be the Shepp-Logan phantom presented in section 2.3 and the phantom called *Ph2*, which can be seen in Figure (3.5), both have sampling parameters $\Delta y = \Delta x = 0.01$ and $M = N = 201$. The sinograms have sampling parameters $\Delta\rho = 0.01$, $R = 201$ and $T = 201$, and the sinogram of Ph2 can be seen in Figure (3.6). The sampling parameters are the same as for both the Shepp-Logan phantom and Ph2. The Ph2 phantom is a combination of ellipses, triangles, squares and Gauss bells.

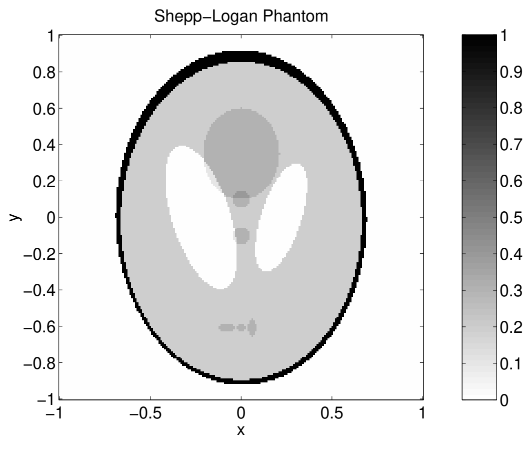

Figure 3.3: Shepp-Logan phantom

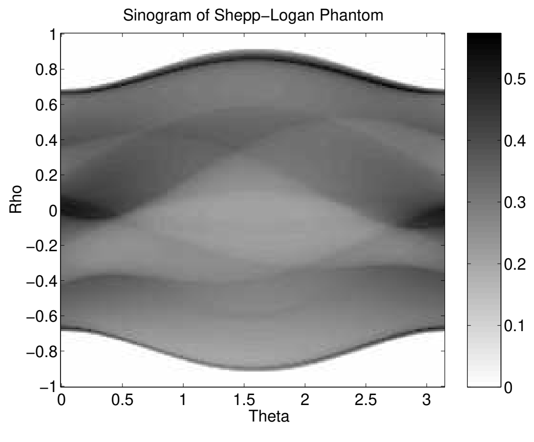

Figure 3.4: Sinogram of Shepp-Logan phantom

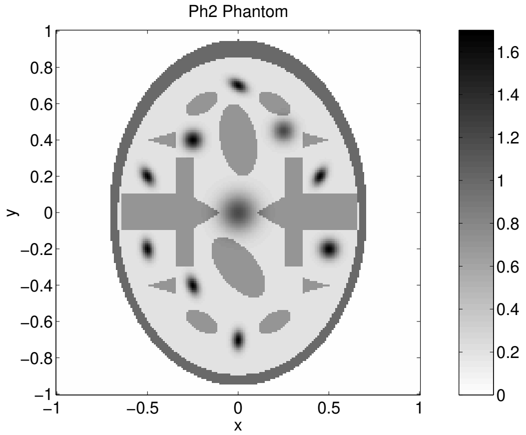

Figure 3.5: Ph2 phantom

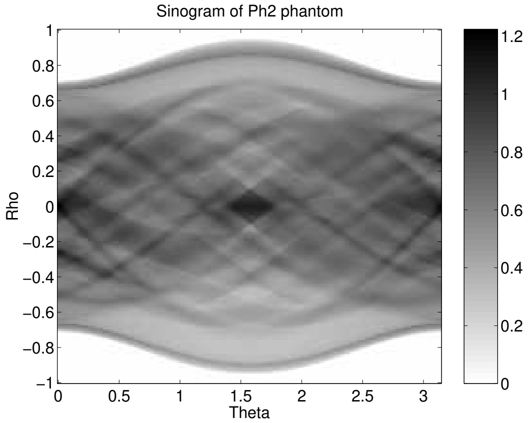

Figure 3.6: Sinogram of Ph2 phantom

3.3.1 Influence of Sampling Parameters

The first tests are made to examine the ability of the different reconstruction methods to reconstruct using different sampling parameters.

The sampling parameters for the sinogram are $\Delta\rho = 0.01$, $R = 201$ and $T = 200$. The image is equally sampled in the x and y direction always covering the area $-1 \leq x \leq 1$ and $-1 \leq y \leq 1$ with M in the range from 51 to 401, with the corresponding sampling Δx from 0.04 to 0.005. With $M = 51$ the image is overdetermined since the sinogram has $201 \times 200 = 40200$ samples and the image has $51 \times 51 = 2601$ samples, and similarly for $M = 401$ the image is underdetermined with $401 \times 401 = 160801$ samples.

The algorithms tested in this subsection are the Filtered Backprojection, Filtering After Backprojection, Fourier Slice, ART, MART and the ML-EM, and the results can be viewed in Figures (3.7) to (3.10).

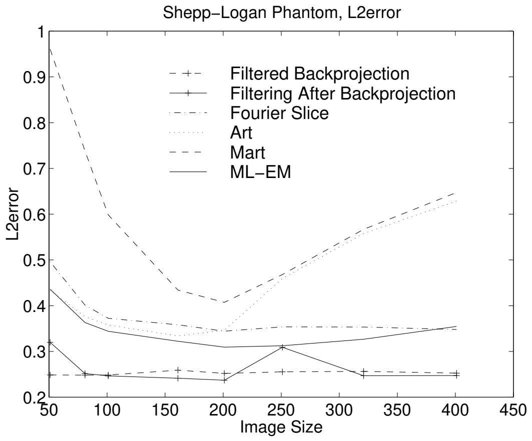

Figure 3.7: The L_2 error as function of the image size, Shepp-Logan phantom

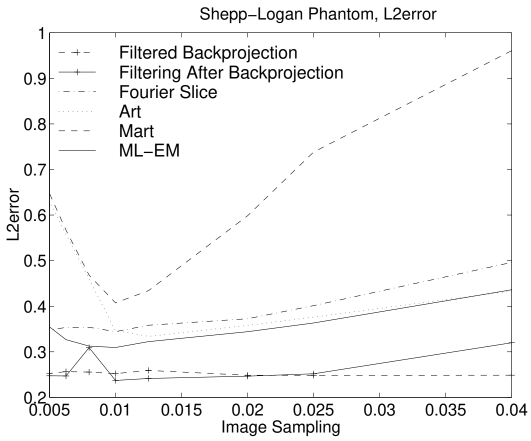

Figure 3.8: The L_2 error as function of the image sampling, Shepp-Logan phantom

In general the direct methods perform best with Filtered Backprojection and Filtering After Backprojection as the second best. The EM algorithm performs best of the iterative methods. These three algorithms are also the ones most independent of the image sam-

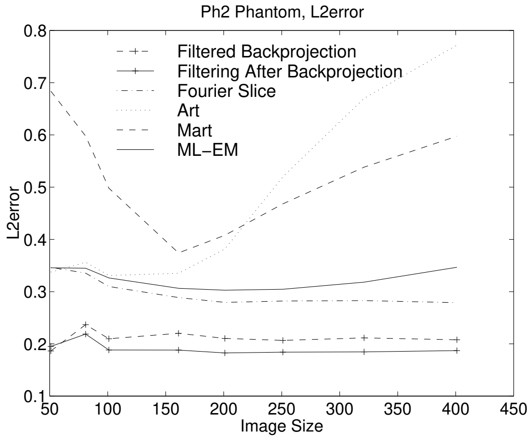

Figure 3.9: The L_2 error as function of the image size, Ph2 phantom

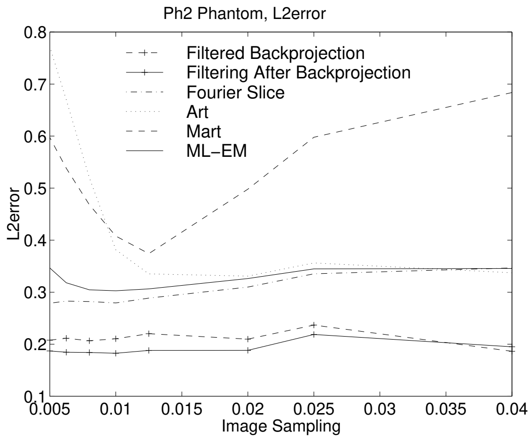

Figure 3.10: The L_2 error as function of the image sampling, Ph2 phantom

pling, with the MART algorithm as the worst performance. This is probably a built-in error of the MART algorithm so the entropy is maximized without proper weight of the input data.

The ART algorithm performs well as long as the image is determined, but could probably perform well at a lower sampling if a smoothness prior is applied to the image.

The reconstructed images of the Ph2 phantom with $M = 201$ of the Filtered Backprojection, Filtering After Backprojection, Fourier Slice, ART, MART and EM can be seen in Figures (3.11) to (3.16). The direct methods give the visually sharpest results together with ART, which has some artifacts. The MART and EM result in more blurred images, but the L_2 error is less for EM than for ART.

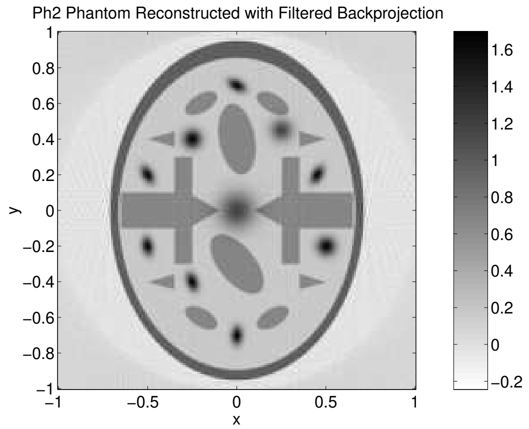

Figure 3.11: Ph2 phantom reconstructed with Filtered Backprojection, $M = 201$

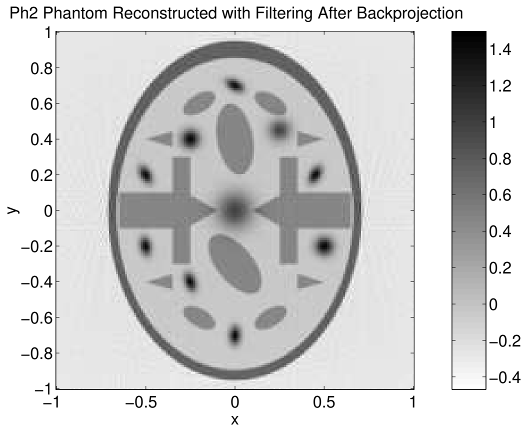

Figure 3.12: Ph2 phantom reconstructed with Filtering After Backprojection, $M = 201$

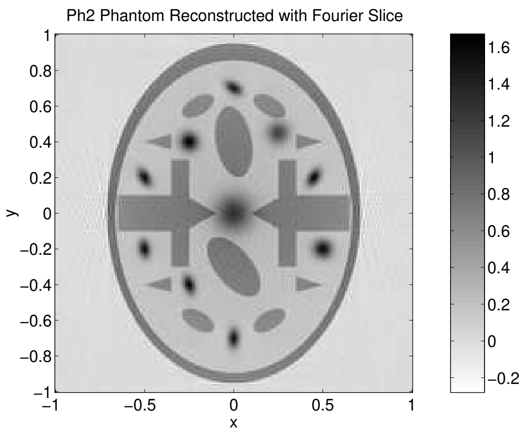

Figure 3.13: Ph2 phantom reconstructed with Fourier Slice, $M = 201$

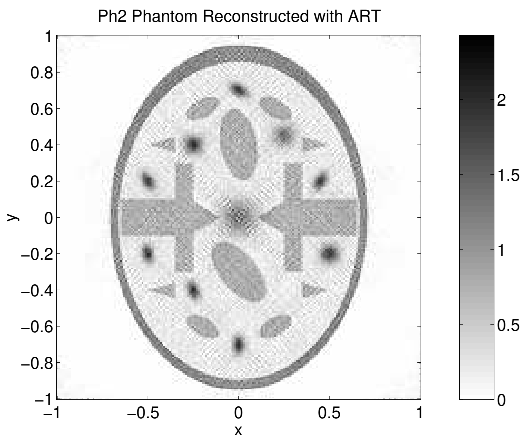

Figure 3.14: Ph2 phantom reconstructed with ART, $M = 201$

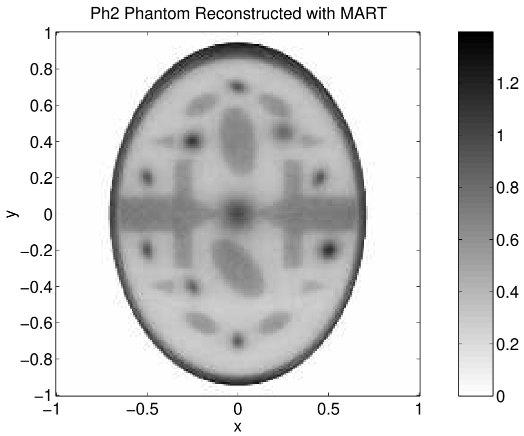

Figure 3.15: Ph2 phantom reconstructed with MART, $M = 201$

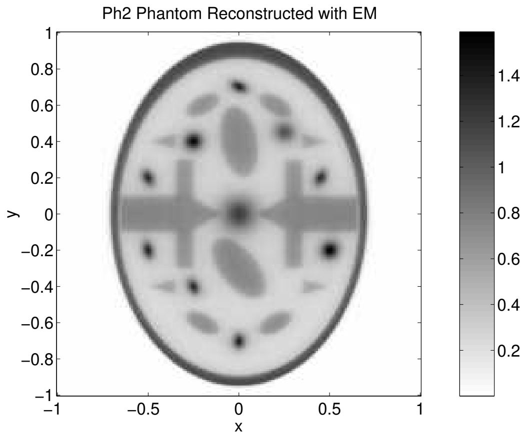

Figure 3.16: Ph2 phantom reconstructed with EM, $M = 201$

3.3.2 Influence of Noise in the Sinogram

In this subsection the different reconstruction algorithms are tested with noisy sinograms. First the noise is additive white Gaussian noise with varying standard deviation. The algorithms are tested in three cases underdetermined by a factor of 1.5, determined and overdetermined by a factor of 1.5. The added noise is in the range from $\sigma = 0.01$ to $\sigma = 0.9$. A sinogram of Ph2 with Gaussian noise of $\sigma = 0.07$ is shown in Figure (3.17).

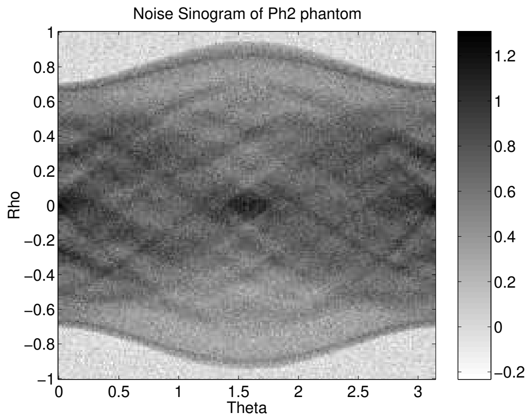

Figure 3.17: Sinogram of Ph2 with Gaussian noise with $\sigma = 0.07$

The L_2 errors of reconstruction of Ph2 with the different algorithms for the determined system can be seen in Figure (3.18)

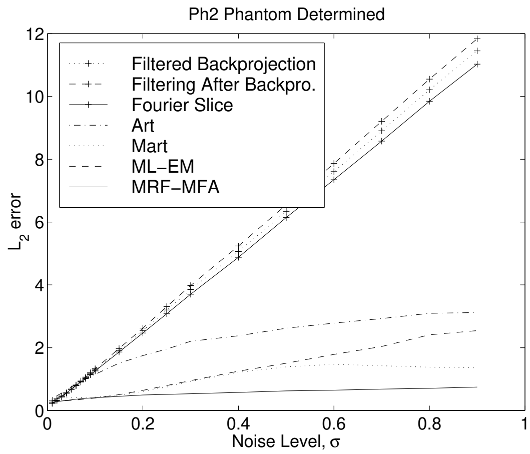

Figure 3.18: L_2 error as functions of the Gaussian noise level, for a determined system.

The L_2 errors of the underdetermined system can be seen in Figure (3.19) and of the overdetermined system can be seen in Figure (3.20)

From this we can conclude that for noise levels above 0.03 the iterative methods outperform the direct methods, especially the MART, EM and MRF-MFA algorithms. This is probably based on the built in smoothing in MART and EM and the neighbor connectivity in MRF-MFA. The fact that the direct methods are better at low noise levels could be the result of an incorrect estimate of the transformation matrix $\boldsymbol{A}$.

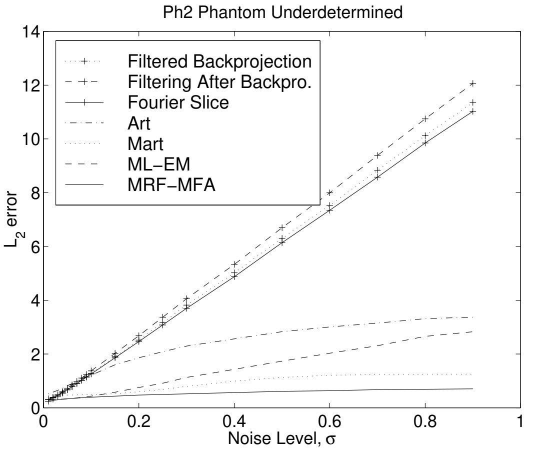

Figure 3.19: L_2 error as functions of the Gaussian noise level, for an underdetermined system.

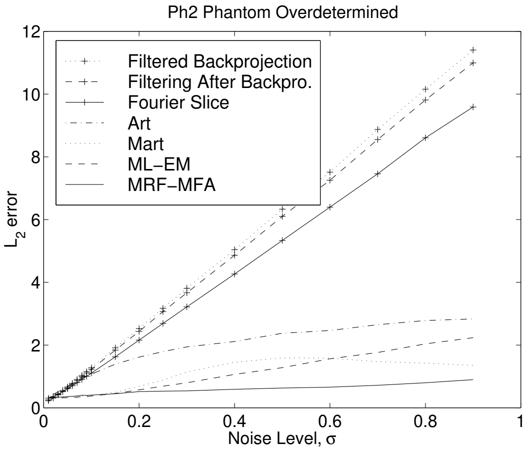

Figure 3.20: L_2 error as functions of the Gaussian noise level, for an overdetermined system.

This can be verified in Figures (3.21) to (3.27), but the pixel values have been limited to the range [0 : 1.7] to make the figures more comparable, which limit the noise especially on the direct methods.

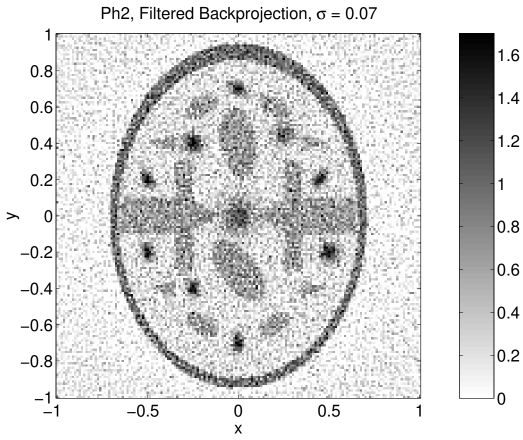

Figure 3.21: Ph2 phantom reconstructed with FB, noise level in sinogram at 0.07

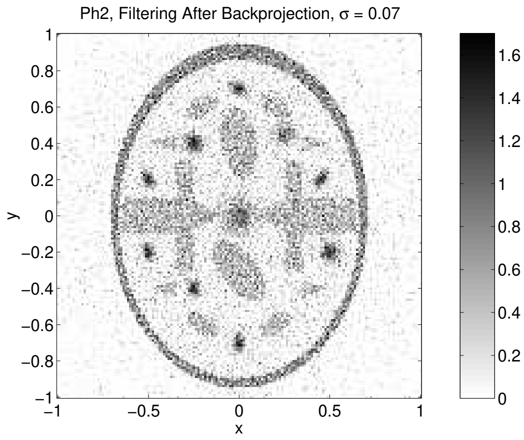

Figure 3.22: Ph2 phantom reconstructed with FAB, noise level in sinogram at 0.07

Then the different algorithms are tested with Poisson generated sinograms, meaning that the noisy sinograms are generated by individual uncorrelated Poisson processes for each point in the sinogram. The observation time is changed to simulate different lengths of scanning period. The length of scanning time is converted into a total number of counts in the sinogram so a weak tracer with long scanning time is equivalent to a very active tracer in a short scanning time. In Figure (3.28) the different algorithms are tested with an increasing number of total counts.

At low count rate the EM and the MART algorithms outperform the direct methods as well as the MRF-MFA method. This is probably caused by the incorrect model of the noise structure in the MRF-MFA algorithm. For a total number of counts of 10^6 the

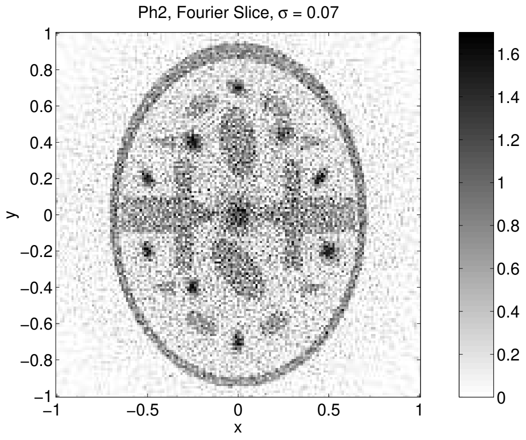

Figure 3.23: Ph2 phantom reconstructed with FS, noise level in sinogram at 0.07

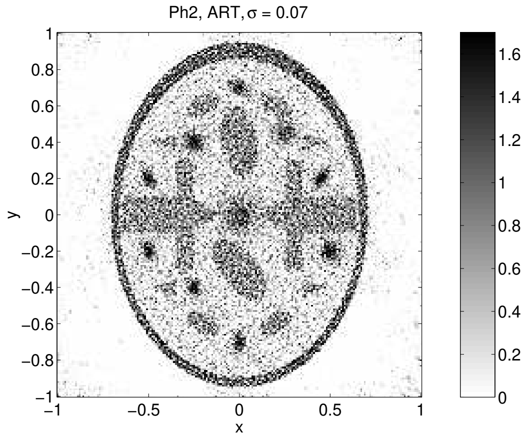

Figure 3.24: Ph2 phantom reconstructed with ART, noise level in sinogram at 0.07

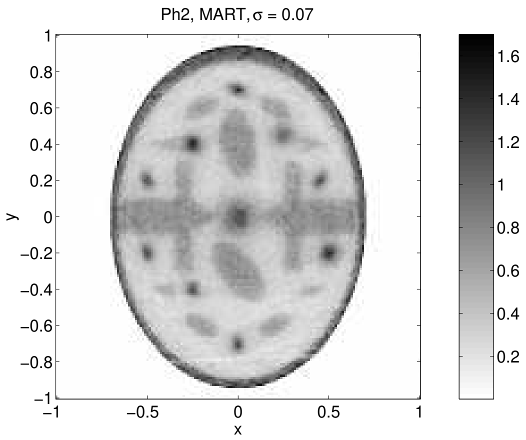

Figure 3.25: Ph2 phantom reconstructed with MART, noise level in sinogram at 0.07

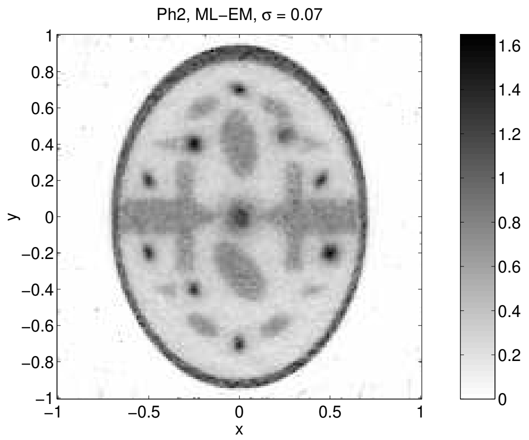

Figure 3.26: Ph2 phantom reconstructed with EM, noise level in sinogram at 0.07

results can be seen in Figures (3.30) to (3.35), all limited to the range from $[0; 1.7]$.

The best visual results are the results of the EM and the MRF-MFA algorithms, but the ART and the direct methods will visually improve with just a small smoothing of the final result. The MRF-MFA improvement is based on the sharper edges.

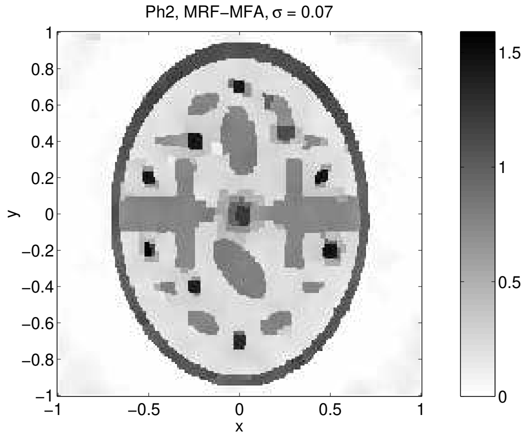

Figure 3.27: Ph2 phantom reconstructed with MRF-MFA, noise level in sinogram at 0.07

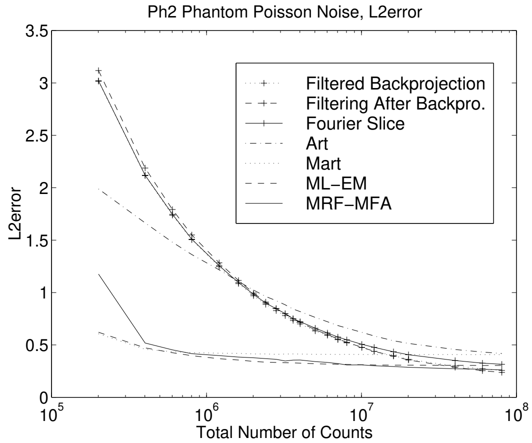

Figure 3.28: L_2 error as function of total number of counts in the sinogram.

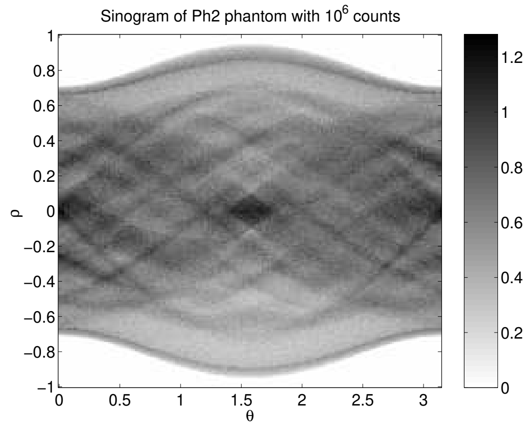

Figure 3.29: Sinogram of Ph2 phantom with a total number of counts of 10^6

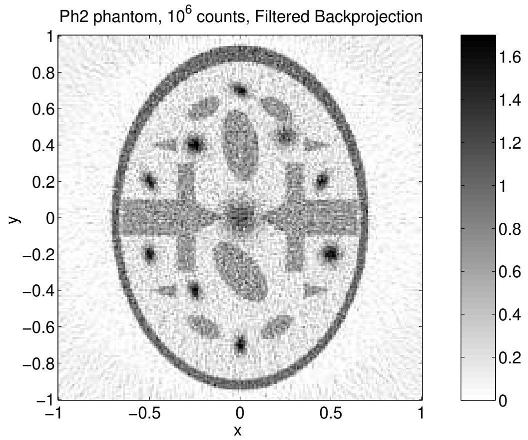

Figure 3.30: Ph2 phantom with a total number of counts of 10^6 reconstructed with Filtered Backprojection

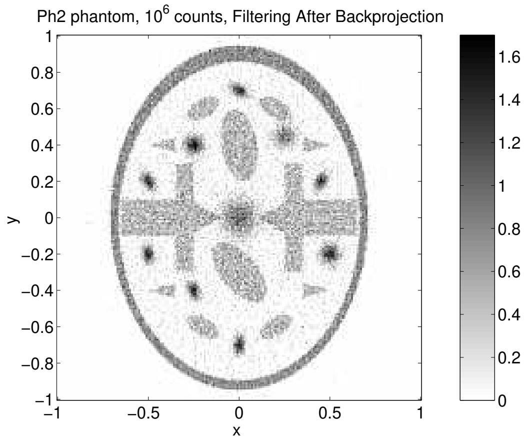

Figure 3.31: Ph2 phantom with a total number of counts of 10^6 reconstructed with Filtering After Backprojection

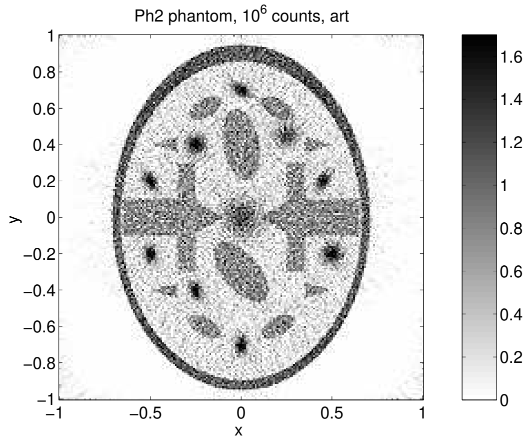

Figure 3.32: Ph2 phantom with a total number of counts of 10^6 reconstructed using ART

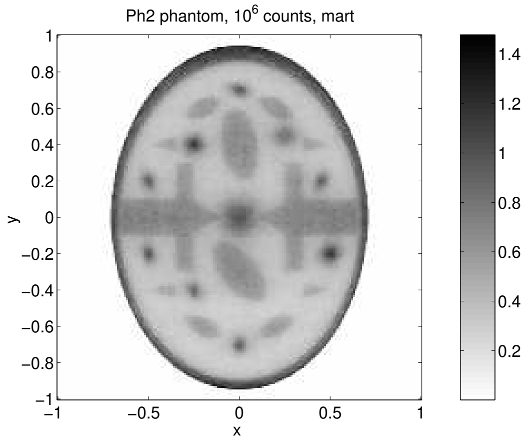

Figure 3.33: Ph2 phantom with a total number of counts of 10^6 reconstructed using MART

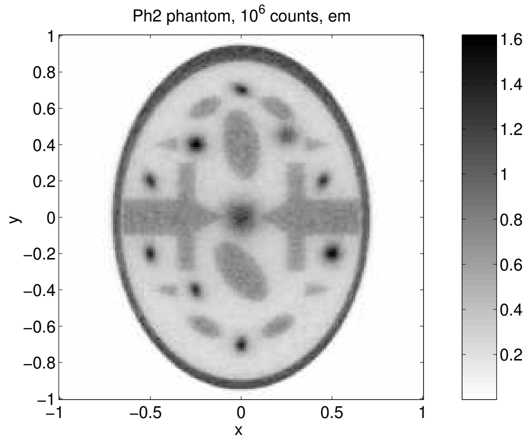

Figure 3.34: Ph2 phantom with a total number of counts of 10^6 reconstructed using ML-EM

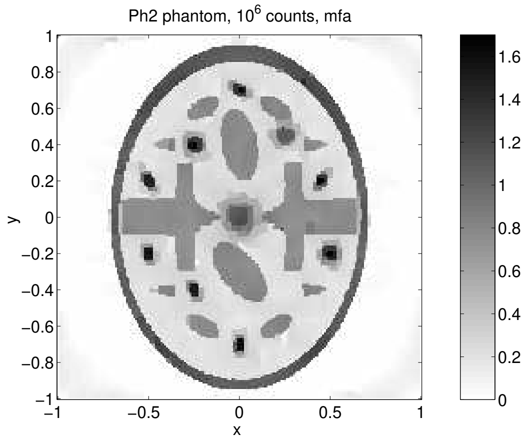

Figure 3.35: Ph2 phantom with a total number of counts of 10^6 reconstructed with Mean Field Annealing

3.3.3 Reconstruction of Real Brain

In this section the performance of some of the different algorithms are tested on real data from the GE-PET scanner at Rigshospitalet. The sinogram is intensity corrected using both transmission and blankscans. Since the true image is not known the algorithms can only be judged visually. The sinogram can be seen in Figure (3.36) and it corresponds to approximately $3 * 10^6$ counts.

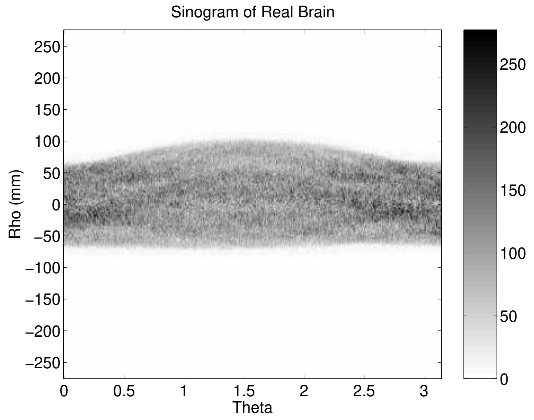

Figure 3.36: Sinogram of Real brain

The result of the different algorithms can be seen in Figures (3.37) to (3.40), where only the part withholding the brain is reconstructed, knowing that the area covered by the scanner is much larger. The images are reconstructed at a smaller sampling than the scanner geometry provides data to. This actually corresponds to an underdetermined system like in Section 3.3.1.

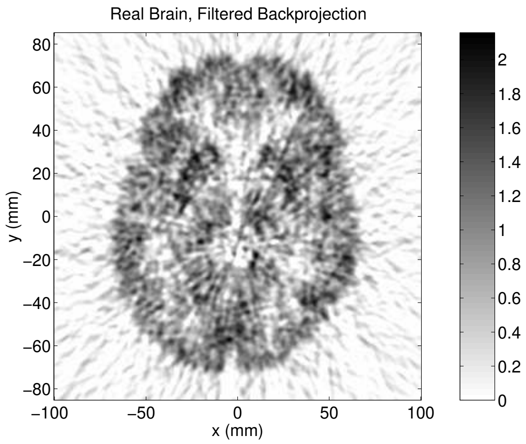

Figure 3.37: Reconstructed with Filtered backprojection

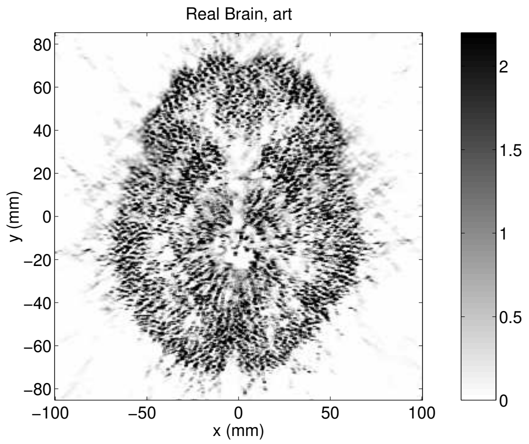

Figure 3.38: Reconstructed using ART

The best results are the EM and the MRF-MFA, but the EM seems to have a higher visual resolution. The Filtered Backprojection and the ART are both contaminated with noise.

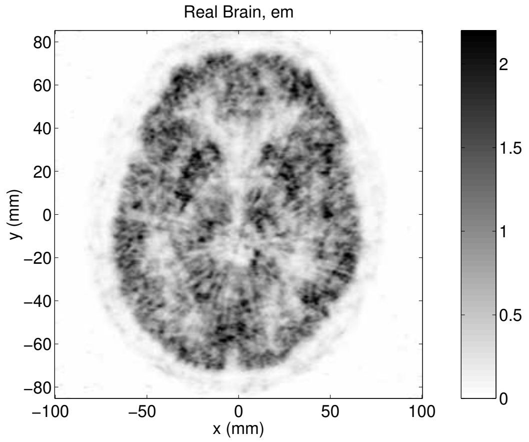

Figure 3.39: Reconstructed with the EM algorithm

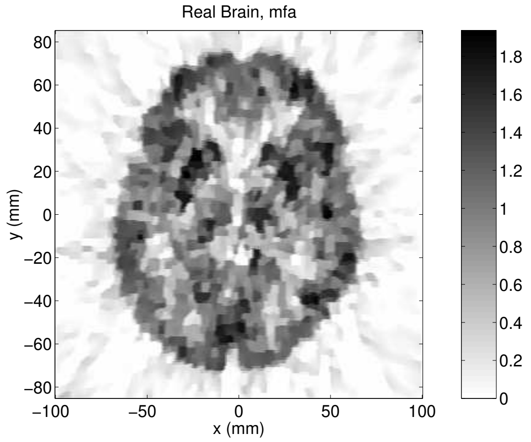

Figure 3.40: Reconstructed using the MRF model with MFA

3.3.4 Summary of Results

As a summary of the results it can be stated that the direct methods are best performing at low noise levels, but all of them result in poor performances with realistic noise levels, at 10^6-10^7 counts. Among the iterative methods the EM and the MRF-MFA give the most noise stable results, but the EM is also the most time consuming and the ART algorithm is much faster with only slightly poorer result. The MART algorithm is also not very stable at low or high image size, and the visual impression is not impressive. The problem with MRF-MFA is the parameter estimation, see Section5.5.2. In most scanners today Filtered Backprojection is used because the algorithm are the most stable.

Chapter 4

Three-Dimensional Transformation

The 3D Radon transform is not directly useful in 3D tomography, because the 3D Radon transform is a plane integral in space [Deans, 1983], and in tomography the observed data is in the form of line integrals. But it is possible to generalize the Radon transform to cover line integrals in 3D. A line in 3D cannot be described using a single normal equation as in Eq. 2.16 but in vector form

$$\boldsymbol{r} = \boldsymbol{r_0} + s\ \boldsymbol{e_s}, \quad s \in \mathbb{R} \tag{4.1}$$

where $\| \boldsymbol{e_s} \| = 1$ is the unity directional vector, $\boldsymbol{r_0}$ is the offset and s is the free parameter.

The unity vector can be defined by two angles (θ, ϕ) where θ is defined as the angle to the x-axis in the xy-plane and ϕ is the angle from this plane, Figure (4.1).

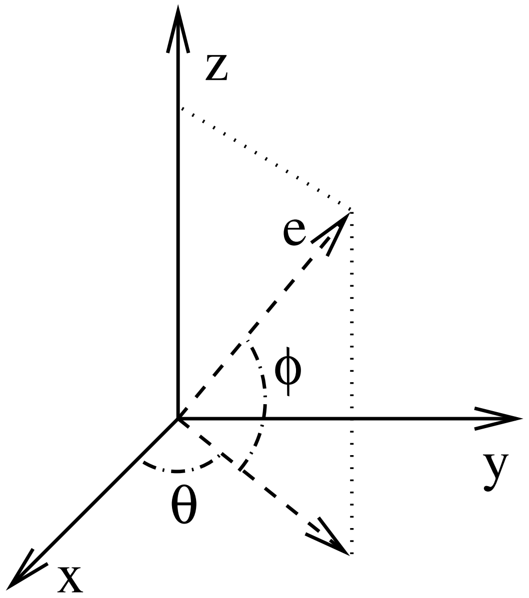

Figure 4.1: Definition of (θ, ϕ).

Defining $\boldsymbol{e_s}$ using these angles gives

$$\boldsymbol{e_s} = \begin{pmatrix} \cos\theta\cos\phi \\ \sin\theta\cos\phi \\ \sin\phi \end{pmatrix}, \theta \in [0; 2\pi[\text{ and } \phi \in \left[-\frac{\pi}{2}; \frac{\pi}{2}\right] \tag{4.2}$$

The chosen range of the angles covers the 3D space, but other definitions are possible because

$$\begin{aligned} \boldsymbol{e_s}(\theta, \phi + \pi) &= \boldsymbol{e_s}(\theta + \pi, -\phi) = -\boldsymbol{e_s}(\theta, \phi) && (4.3) \\ \boldsymbol{e_s}(-\theta, \phi) &= \boldsymbol{e_s}(2\pi - \theta, \phi) && (4.4) \\ \boldsymbol{e_s}(\theta, -\phi) &= \boldsymbol{e_s}(\theta, 2\pi - \phi) && (4.5) \end{aligned}$$

The offset $\boldsymbol{r_0}$ can be chosen to be the shortest distance from origin to the line like in 2D, meaning that $\boldsymbol{e_s} \cdot \boldsymbol{r_0} = 0$, and from this $\boldsymbol{r}_0$ can be defined by 2 orthogonal unity vectors $\boldsymbol{e}_u$ and $\boldsymbol{e}_v$

$$\boldsymbol{r_0} = u\,\boldsymbol{e_u} + v\,\boldsymbol{e_v} \tag{4.6}$$

where u and v are real numbers and $\boldsymbol{e_u} \cdot \boldsymbol{e_v} = 0$. This definition opens to many different definitions of $\boldsymbol{e_u}, \boldsymbol{e_v}$ [M. Defrise and Geissbuhler, 1990, Deans, 1983], since the vectors can be rotated around $\boldsymbol{e_s}$ arbitrarily. In [Schorr et al., 1983, Clack, 1992] $\boldsymbol{e_u}$ and $\boldsymbol{e_v}$ are defined as, called $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ in [Clack, 1992],

$$\boldsymbol{e_u} = \begin{pmatrix} -sin\theta \\ \cos\theta \\ 0 \end{pmatrix} \text{ and } \boldsymbol{e_v} = \begin{pmatrix} -\cos\theta\sin\phi \\ -\sin\theta\sin\phi \\ \cos\phi \end{pmatrix} \tag{4.7}$$

with the z coordinate of $\boldsymbol{e_u}$ equal to zero. Integrating along the line in Eq. 4.1, using Eq. 4.6, gives

$$\breve{g}(\theta, \phi, u, v) = \int_{-\infty}^{\infty} g(s\boldsymbol{e_s} + u\boldsymbol{e_u} + v\boldsymbol{e_v})\, ds \tag{4.8}$$

From Eq. 4.8 it can be seen that the 3D Radon transform of lines have 4 degrees of freedom eq. parameters. To obtain information of the 3D-line an important primitive is the transformed of a point source placed at $\boldsymbol{r_p}$

$$\begin{aligned} g_{point}(\boldsymbol{r}) &= \delta(\boldsymbol{r} - \boldsymbol{r_p}) && (4.9) \\ &\Downarrow \\ \breve{g}_{point}(\theta, \phi, u, v) &= \int_{-\infty}^{\infty} \delta(s\boldsymbol{e_s} + u\boldsymbol{e_u} + v\boldsymbol{e_v} - \boldsymbol{r_p})\, ds && (4.10) \\ &= \int_{-\infty}^{\infty} \delta((s - s_p)\boldsymbol{e_s} + (u - u_p)\boldsymbol{e_u} + (v - v_p)\boldsymbol{e_v})\, ds && (4.11) \\ &= \int_{-\infty}^{\infty} \delta(\tilde{s}\boldsymbol{e_s} + (u - u_p)\boldsymbol{e_u} + (v - v_p)\boldsymbol{e_v})\, d\tilde{s} && (4.12) \\ &= \delta(u - u_p)\delta(v - v_p) && (4.13) \end{aligned}$$

where $\boldsymbol{r_p} = s_p\boldsymbol{e_s} + u_p\boldsymbol{e_u} + v_p\boldsymbol{e_v}$ has been used. It should be noticed that $u_p = \boldsymbol{r_p} \cdot \boldsymbol{e_u}$ and $v_p = \boldsymbol{r_p} \cdot \boldsymbol{e_v}$.

To make a connection between 3D and 2D considering the special case where $\phi = 0$, $u = \rho$ and $v = z_0$

$$\breve{g}(\theta, 0, \rho, z_0) = \int_{-\infty}^{\infty} g(s\cos\theta - \rho\sin\theta, s\sin\theta + \rho\cos\theta, z_0)\, ds \tag{4.14}$$

this is a stack of 2D sinograms in the z direction, which can be seen from Eq. 2.17 rotated $-\frac{\pi}{2}$.

In PET scanners it is not possible to collect from the whole sphere, but only from a limited angle like with a Multi Ring PET scanner. The sphere can be a truncated ellipsoid parameterized by limiting the angle parameter $|\phi| \le \psi$. This gives rise to the limitation of the angle geometry Ω_ψ

$$\Omega_\psi = \{0 \le \theta < \pi \vee |\phi| \le \psi\} \tag{4.15}$$

where Eq. 4.3 is used to reduce the parameter space by a factor of two. This is not used in [Clack, 1992] and results in filters that differ with a factor of two by the ones presented in this thesis. The parameters u and v are only limited if the object is limited since $|\boldsymbol{r}| = \sqrt{x^2 + y^2 + z^2} = \sqrt{s^2 + u^2 + v^2}$. If the object has a maximum range of r_{max} meaning that

$$g(\boldsymbol{r}) = 0 \ \text{ if } \ |\boldsymbol{r}| \ge r_{max} \tag{4.16}$$

From this result it can be seen that the u and v are limited by r_{max}

$$-r_{max} < u < r_{max} \ \text{ and } \ -r_{max} < v < r_{max} \tag{4.17}$$

With only a limited angle geometry special considerations for the direct inversion methods must be made. In [Rogers et al., 1987] a 3 step scheme is described. First a crude volume is reconstructed by stacking a set axial slices reconstructed in 2D. The missing angles are estimated by calculating the forward projecting of the initial volume. After that a full angle reconstruction is applied, like Filtered Backprojection. In [Kinahan and Rogers, 1989] an implementation shows that an improvement of the volume is obtained, compared to 2D reconstruction.

4.1 3D Inversion of Line Transform

In this section the focus is on 3D versions of the direct inversion methods. The iterative methods will not be presented since the scheme is the same as for 2D, presented in section 3.2. It is assumed that the angle geometry satisfies Orlov's conditions [Orlov, 1975a, Orlov, 1975b].

4.1.1 3D Fourier Slice Reconstruction

The function $g(\mathbf{r})$ can also be reconstructed in 3D using Fourier techniques. Applying a 2D Fourier transform in the (u, v) parameters in Eq. 4.1

$$\begin{aligned}\breve{G}(\theta, \phi, \nu_u, \nu_v) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \breve{g}(\theta, \phi, u, v)\, e^{-j2\pi(u\nu_u + v\nu_v)}\, du dv &(4.18)\\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(s\boldsymbol{e_s} + u\boldsymbol{e_u} + v\boldsymbol{e_v})\, e^{-j2\pi(u\nu_u + v\nu_v)}\, ds\, du\, dv &(4.19)\end{aligned}$$

where $\boldsymbol{r} = s\boldsymbol{e_s} + u\boldsymbol{e_u} + v\boldsymbol{e_v}$. This indicates a close connection to the Fourier transform

$$G(\boldsymbol{\nu}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(\boldsymbol{r})\, e^{-j2\pi\boldsymbol{\nu}\cdot\boldsymbol{r}}\, d\boldsymbol{r} \tag{4.20}$$

$$g(\boldsymbol{r}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(\boldsymbol{\nu})\, e^{j2\pi\boldsymbol{\nu}\cdot\boldsymbol{r}}\, d\boldsymbol{\nu} \tag{4.21}$$

where $\boldsymbol{\nu} = \nu_s\boldsymbol{e_s} + \nu_u\boldsymbol{e_u} + \nu_v\boldsymbol{e_v}$ is the frequency vector. Using that $\boldsymbol{e_s}$, $\boldsymbol{e_u}$ and $\boldsymbol{e_v}$ are orthogonal gives

$$u\nu_u + v\nu_v = \boldsymbol{r}\cdot\boldsymbol{e_u}\nu_u + \boldsymbol{r}\cdot\boldsymbol{e_v}\nu_v \tag{4.22}$$

$$= \boldsymbol{r}\cdot(\nu_u\boldsymbol{e_u} + \nu_v\boldsymbol{e_v}) \tag{4.23}$$

from Eq. 4.18 it can be seen that

$$\check{G}(\theta,\phi,\nu_u,\nu_v) = G(\nu_u\boldsymbol{e_u} + \nu_v\boldsymbol{e_v}) \tag{4.24}$$

which is the Fourier slice theorem for line integrals in 3D, and the function $g(\boldsymbol{r})$ can be reconstructed by applying a two dimensional Fourier transformation to the sinogram for all (θ,ϕ) and after that a remapping of the spectrum to the inverse three dimensional Fourier transformation.

$$G(\nu_u\boldsymbol{e_u} + \nu_v\boldsymbol{e_v}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \check{g}(\theta,\phi,u,v)\, e^{-j2\pi(u\nu_u+v\nu_v)}\, dudv \tag{4.25}$$

$$g(\boldsymbol{r}) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(\boldsymbol{\nu})\, e^{j2\pi\boldsymbol{\nu}\cdot\boldsymbol{r}}\, d\boldsymbol{\nu} \tag{4.26}$$

However, the mapping of the spectrum raises a non-trivial problem, since $\nu_u\boldsymbol{e_u} + \nu_v\boldsymbol{e_v}$ is a four parameter description of the three dimensional space $\boldsymbol{\nu}$ and some sort of integration is needed to eliminate one free parameter.

4.1.2 3D Backprojection

To define algorithms like the 2D Filtered Backprojection and Filtering After Backprojection a definition of the 3D Backprojection operator is needed. The Backprojection operator is defined in [M. Defrise and Geissbuhler, 1990, Schorr et al., 1983] as

$$\begin{aligned}\tilde{g}(\boldsymbol{r}) &= \int_0^{\pi}\int_{-\psi}^{\psi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \check{g}(\theta,\phi,u,v)\delta(u-\boldsymbol{r}\cdot\boldsymbol{e_u})\delta(v-\boldsymbol{r}\cdot\boldsymbol{e_v})\, dudv\cos(\phi)d\phi d\theta \\ &= \int_0^{\pi}\int_{-\psi}^{\psi} \check{g}(\theta,\phi,\boldsymbol{r}\cdot\boldsymbol{e_u},\boldsymbol{r}\cdot\boldsymbol{e_v})\, \cos(\phi)\, d\phi\, d\theta \end{aligned} \tag{4.27}$$

which is an integration in all angles of the point source in $\boldsymbol{r}$, Eq. 4.13. Normally the Backprojection operator is presented in terms of angle geometry and independent of the actual implementation

$$\tilde{g}(\boldsymbol{r}) = \int_{\boldsymbol{\Omega}_\psi} \check{g}(\theta,\phi,\boldsymbol{r}\cdot\boldsymbol{e_u},\boldsymbol{r}\cdot\boldsymbol{e_v})\, d\boldsymbol{\Omega} \tag{4.28}$$

with $d\boldsymbol{\Omega} = \cos(\phi)\, d\phi\, d\theta$ and $\boldsymbol{\Omega}_\psi$ as the angle geometry used.

4.1.3 3D Filtered Backprojection

The filtered backprojection is a two step algorithm where the first step is a two dimensional filtering in the u and v parameter followed by a backprojection [Clack, 1992]. The 2d filtering can be expressed as a convolution

$$\tilde{\breve{g}}(\theta, \phi, u, v) \;=\; \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \breve{g}(\theta, \phi, u', v')\, h(\theta, \phi, u' - u, v' - v)\, du'\, dv' \qquad (4.29)$$

where $h(\theta, \phi, u, v)$ is the filter kernel. The next step is the backprojection step

$$g(\boldsymbol{r}) \;=\; \int_{\boldsymbol{\Omega}_\psi} \tilde{\breve{g}}(\theta, \phi, \boldsymbol{r} \cdot \boldsymbol{e_u}, \boldsymbol{r} \cdot \boldsymbol{e_v})\, d\boldsymbol{\Omega} \qquad (4.30)$$

One way to determine the valid filters is to examine the 3D Fourier transform of Eq. 4.30

$$\begin{aligned}
G(\boldsymbol{\nu}) \;&=\; \iiint_{-\infty}^{\infty} \Big(\int_{\boldsymbol{\Omega}_\psi} \tilde{\breve{g}}(\theta, \phi, \boldsymbol{r} \cdot \boldsymbol{e_u}, \boldsymbol{r} \cdot \boldsymbol{e_v})\, d\boldsymbol{\Omega}\Big)\, e^{-j2\pi \boldsymbol{\nu}\cdot\boldsymbol{r}}\, d\boldsymbol{r} && (4.31)\\
&=\; \int_{\boldsymbol{\Omega}_\psi} \iiint_{-\infty}^{\infty} \tilde{\breve{g}}(\theta, \phi, \boldsymbol{r} \cdot \boldsymbol{e_u}, \boldsymbol{r} \cdot \boldsymbol{e_v})\, e^{-j2\pi \boldsymbol{\nu}\cdot\boldsymbol{r}}\, d\boldsymbol{r}\, d\boldsymbol{\Omega} \\
&=\; \int_{\boldsymbol{\Omega}_\psi} \iiint_{-\infty}^{\infty} \tilde{\breve{g}}(\theta, \phi, u, v)\, e^{-j2\pi(s\boldsymbol{e_s}\cdot\boldsymbol{\nu} + u\boldsymbol{e_u}\cdot\boldsymbol{\nu} + v\boldsymbol{e_v}\cdot\boldsymbol{\nu})}\, ds\, du\, dv\, d\boldsymbol{\Omega} \\
&=\; \int_{\boldsymbol{\Omega}_\psi} \int_{-\infty}^{\infty} \breve{G}(\theta, \phi, \boldsymbol{e_u}\cdot\boldsymbol{\nu}, \boldsymbol{e_v}\cdot\boldsymbol{\nu})\, H(\theta, \phi, \boldsymbol{e_u}\cdot\boldsymbol{\nu}, \boldsymbol{e_v}\cdot\boldsymbol{\nu})\, e^{-j2\pi s\boldsymbol{e_s}\cdot\boldsymbol{\nu}}\, ds\, d\boldsymbol{\Omega} \\
&=\; \int_{\boldsymbol{\Omega}_\psi} \breve{G}(\theta, \phi, \boldsymbol{e_u}\cdot\boldsymbol{\nu}, \boldsymbol{e_v}\cdot\boldsymbol{\nu})\, H(\theta, \phi, \boldsymbol{e_u}\cdot\boldsymbol{\nu}, \boldsymbol{e_v}\cdot\boldsymbol{\nu})\, \delta(\boldsymbol{e_s}\cdot\boldsymbol{\nu})\, d\boldsymbol{\Omega} \\
&=\; \int_{\boldsymbol{\Omega}_\psi} G(\nu_u\boldsymbol{e_u}\cdot\boldsymbol{\nu} + \nu_v\boldsymbol{e_v}\cdot\boldsymbol{\nu})\, H(\theta, \phi, \boldsymbol{e_u}\cdot\boldsymbol{\nu}, \boldsymbol{e_v}\cdot\boldsymbol{\nu})\, \delta(\boldsymbol{e_s}\cdot\boldsymbol{\nu})\, d\boldsymbol{\Omega} && (4.32)
\end{aligned}$$

using Eq. 4.24 and since $\boldsymbol{e_s}\cdot\boldsymbol{\nu} = 0$ (from $\delta(\boldsymbol{e_s}\cdot\boldsymbol{\nu})$)

$$\begin{aligned}
&=\; \int_{\boldsymbol{\Omega}_\psi} G(\boldsymbol{\nu})\, H(\theta, \phi, \boldsymbol{e_u}\cdot\boldsymbol{\nu}, \boldsymbol{e_v}\cdot\boldsymbol{\nu})\, \delta(\boldsymbol{e_s}\cdot\boldsymbol{\nu})\, d\boldsymbol{\Omega} && (4.33)\\
&=\; G(\boldsymbol{\nu}) \int_{\boldsymbol{\Omega}_\psi} H(\theta, \phi, \boldsymbol{e_u}\cdot\boldsymbol{\nu}, \boldsymbol{e_v}\cdot\boldsymbol{\nu})\, \delta(\boldsymbol{e_s}\cdot\boldsymbol{\nu})\, d\boldsymbol{\Omega} && (4.34)
\end{aligned}$$

where it is used that $\int_{-\infty}^{\infty} e^{-j2\pi xy} dy = \delta(x)$ and

$$H(\theta, \phi, \nu_u, \nu_v) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} h(\theta, \phi, u, v) e^{-j2\pi(u\nu_u + v\nu_v)}\, du\, dv \qquad (4.35)$$

This leads to the following filter criteria for $H(\theta, \phi, \nu_u, \nu_v)$

$$1 = \int_{\boldsymbol{\Omega}_\psi} H(\theta, \phi, \boldsymbol{e_u} \cdot \boldsymbol{\nu}, \boldsymbol{e_v} \cdot \boldsymbol{\nu})\, \delta(\boldsymbol{e_s} \cdot \boldsymbol{\nu})\, d\boldsymbol{\Omega} \tag{4.36}$$

Many filters satisfy this condition, but some considerations can limit the number of valid filters. One type of filters is called factorizable filters [Defrise et al., 1989] and is factored as

$$H(\theta, \phi, \nu_u, \nu_v) = D(\theta, \phi)\, W(\nu_u, \nu_v) \tag{4.37}$$

these have equivalent filters in the Filtering after Backprojection algorithm [Clack, 1992]. If $D(\theta, \phi) = 1$ the filter W can be calculated as

$$W(\nu_u, \nu_v) = \frac{1}{\int_{\boldsymbol{\Omega}_\psi} \delta(\boldsymbol{e_s} \cdot \boldsymbol{\nu})\, d\boldsymbol{\Omega}} \tag{4.38}$$

under the assumption that $\boldsymbol{\nu} \cdot \boldsymbol{e_s} = 0$. In the following derivation $\boldsymbol{\nu}$ will be expressed in polar coordinates

$$\boldsymbol{\nu} = \begin{pmatrix} \nu_x \\ \nu_y \\ \nu_z \end{pmatrix} = |\boldsymbol{\nu}| \begin{pmatrix} \cos\theta' \cos\phi' \\ \sin\theta' \cos\phi' \\ \sin\phi' \end{pmatrix} \tag{4.39}$$

Note that new angles are used marked with $'$. $\delta(\boldsymbol{e_s} \cdot \boldsymbol{\nu})$ can then be expressed using Eq. 4.2 as

$$\begin{aligned} \delta(\boldsymbol{e_s} \cdot \boldsymbol{\nu}) &= \delta(|\boldsymbol{\nu}|(\cos\theta\cos\phi\cos\theta'\cos\phi' + \sin\theta\cos\phi\sin\theta'\cos\phi' + \sin\phi\sin\phi')) \\ &= \frac{1}{|\boldsymbol{\nu}|}\delta(\cos\phi\cos\phi'\sin(\phi - \phi') + \sin\phi\sin\phi') \end{aligned} \tag{4.40}$$

The backprojection of $\delta(\boldsymbol{e_s} \cdot \boldsymbol{\nu})$ can be calculated as

$$\int_{\boldsymbol{\Omega}_\psi} \delta(\boldsymbol{e_s} \cdot \boldsymbol{\nu})\, d\boldsymbol{\Omega} = \int_{\boldsymbol{\Omega}_\psi} \frac{1}{|\boldsymbol{\nu}|} \delta(\cos\phi\cos\phi'\sin(\phi - \phi') + \sin\phi\sin\phi')\, d\boldsymbol{\Omega} \tag{4.41}$$

$$= \frac{1}{|\boldsymbol{\nu}|} \int_0^\pi \int_{-\psi}^{\psi} \delta(\cos\phi\cos\phi'\cos(\theta - \theta') + \sin\phi\sin\phi')\, \cos\phi\, d\phi d\theta \tag{4.42}$$

$$= \frac{1}{|\boldsymbol{\nu}|} \int_{-\Psi}^{\Psi} \frac{\cos\phi}{|\cos\phi\cos\phi'\sin(\theta - \theta')|_{(*)}}\, d\phi \tag{4.43}$$

with $\Psi = \min(\psi, \arcsin(\cos\phi'))$ and $(*) = (\cos\phi\cos\phi'\cos(\phi - \phi') + \sin\phi\sin\phi' = 0)$

$$= \frac{1}{|\boldsymbol{\nu}|} \int_{-\Psi}^{\Psi} \frac{\cos\phi}{|\cos^2\phi' - \sin^2\phi|}\, d\phi \tag{4.44}$$

$$= \frac{1}{|\boldsymbol{\nu}|} \left[\arcsin\left(\frac{\sin\phi}{\cos\phi'}\right)\right]_{-\Psi}^{\Psi} \tag{4.45}$$

$$= \frac{2}{|\boldsymbol{\nu}|} \arcsin\left(\frac{\sin\Psi}{\cos\phi'}\right) \tag{4.46}$$

expressing Eq. 4.46 in (ν_s, ν_u, ν_v) using $\nu_s = 0$ since $\boldsymbol{\nu} \cdot \boldsymbol{e_s} = 0$. This results in

$$
\begin{aligned}
|\boldsymbol{\nu}| &= \sqrt{\nu_u^2 + \nu_v^2} && (4.47)\\
\cos\phi' &= \frac{1}{|\boldsymbol{\nu}|}\sqrt{\nu_u^2 + \nu_v^2 \sin^2\phi} = \frac{1}{|\boldsymbol{\nu}|}\sqrt{\nu_x^2 + \nu_y^2} && (4.48)
\end{aligned}
$$

The filter $H(\theta, \phi, \nu_u, \nu_v)$ can be expressed as

$$
H(\theta, \phi, \nu_u, \nu_v) = \begin{cases} \frac{\sqrt{\nu_u^2+\nu_v^2}}{\pi} & \text{if } \sin\psi > \cos\phi' \\ \frac{\sqrt{\nu_u^2+\nu_v^2}}{2\arcsin\left(\frac{\sin\psi}{\cos\phi'}\right)} & \text{if } \sin\psi \leq \cos\phi' \end{cases} \tag{4.49}
$$

For the geometry $\Omega_{\frac{\pi}{2}}$ the filter is

$$
H(\theta, \phi, \nu_u, \nu_v) = \frac{1}{\pi}\sqrt{\nu_u^2 + \nu_v^2} \tag{4.50}
$$

Other filters with different $D(\theta, \phi)$, see [Schorr et al., 1983], can change the filter to match different noise properties. Another type of extra filters that can change the original filter noise properties is called Null filters [Clack, 1992] which satisfy

$$
0 = \int_{\boldsymbol{\Omega}_\psi} N(\theta, \phi, \boldsymbol{e_u} \cdot \boldsymbol{\nu}, \boldsymbol{e_v} \cdot \boldsymbol{\nu})\, \delta(\boldsymbol{e_s} \cdot \boldsymbol{\nu})\, d\boldsymbol{\Omega} \tag{4.51}
$$

and an example of a null filter can found in [Clack, 1992].

4.1.4 3D Filtering After Backprojection

As in two dimensions the backprojection operator can be applied before the filtering [Townsend and T. Spinks, 1989]. With certain filter constrains it can be shown that this algorithm is mathematically equivalent to Filtered Backprojection [Clack, 1992].

The filter conditions can be found by taking the Fourier transform of both sides in Eq.

4.28

$$\tilde{G}(\boldsymbol{\nu}) = \iiint\limits_{-\infty}^{\infty} \int\limits_{\boldsymbol{\Omega}_\psi} \breve{g}(\theta, \phi, \boldsymbol{r}\cdot\boldsymbol{e_u}, \boldsymbol{r}\cdot\boldsymbol{e_v})\, d\boldsymbol{\Omega}\, e^{-j2\pi\boldsymbol{\nu}\cdot\boldsymbol{r}}\, d\boldsymbol{r} \tag{4.52}$$

$$= \iiint\limits_{-\infty}^{\infty} \int\limits_{\boldsymbol{\Omega}_\psi} \int_{-\infty}^{\infty} g(s\boldsymbol{e_s} + (\boldsymbol{r}\cdot\boldsymbol{e_u})\boldsymbol{e_u} + (\boldsymbol{r}\cdot\boldsymbol{e_v})\boldsymbol{e_v})\, ds\, d\boldsymbol{\Omega}\, e^{-j2\pi\boldsymbol{\nu}\cdot\boldsymbol{r}}\, d\boldsymbol{r} \tag{4.53}$$

$$= \iiint\limits_{-\infty}^{\infty} \int\limits_{\boldsymbol{\Omega}_\psi} \int_{-\infty}^{\infty} g(s\boldsymbol{e_s} + \boldsymbol{r})\, ds\, d\boldsymbol{\Omega}\, e^{-j2\pi\boldsymbol{\nu}\cdot\boldsymbol{r}}\, d\boldsymbol{r} \tag{4.54}$$

$$= \int\limits_{\boldsymbol{\Omega}_\psi} \int_{-\infty}^{\infty} \iiint\limits_{-\infty}^{\infty} g(s\boldsymbol{e_s} + \boldsymbol{r})\, e^{-j2\pi\boldsymbol{\nu}\cdot\boldsymbol{r}}\, d\boldsymbol{r}\, ds\, d\boldsymbol{\Omega} \tag{4.55}$$

$$= \int\limits_{\boldsymbol{\Omega}_\psi} \int_{-\infty}^{\infty} \iiint\limits_{-\infty}^{\infty} g(\tilde{\boldsymbol{r}})\, e^{-j2\pi\boldsymbol{\nu}\cdot(\tilde{\boldsymbol{r}} - s\boldsymbol{e_s})}\, d\tilde{\boldsymbol{r}}\, ds\, d\boldsymbol{\Omega} \tag{4.56}$$

$$= \int\limits_{\boldsymbol{\Omega}_\psi} \int_{-\infty}^{\infty} G(\boldsymbol{\nu})\, e^{j2\pi s\boldsymbol{\nu}\cdot\boldsymbol{e_s}}\, ds\, d\boldsymbol{\Omega} \tag{4.57}$$

$$= G(\boldsymbol{\nu}) \int\limits_{\boldsymbol{\Omega}_\psi} \delta(\boldsymbol{\nu}\cdot\boldsymbol{e_s})\, d\boldsymbol{\Omega} \equiv \frac{G(\boldsymbol{\nu})}{H_{fab}(\boldsymbol{\nu})} \tag{4.58}$$

using $\delta(\boldsymbol{\nu}\cdot\boldsymbol{e_s}) = \int_{-\infty}^{\infty} e^{j2\pi s\boldsymbol{\nu}\cdot\boldsymbol{e_s}}\, ds$ and the filter $H_{fab}(\boldsymbol{\nu})$ must satisfy

$$H_{fab}(\boldsymbol{\nu}) = \frac{1}{\int\limits_{\boldsymbol{\Omega}_\psi} \delta(\boldsymbol{\nu}\cdot\boldsymbol{e_s})\, d\boldsymbol{\Omega}} \tag{4.59}$$

Using the results from Eq. 4.46 with $\cos\phi' = \frac{1}{|\boldsymbol{\nu}|}\sqrt{\nu_x^2 + \nu_y^2}$

$$H(\boldsymbol{\nu}) = \begin{cases} \frac{|\boldsymbol{\nu}|}{\pi} & \text{if } \sin\psi > \cos\phi' \\ \frac{|\boldsymbol{\nu}|}{2\arcsin\left(\frac{\sin\psi}{\cos\phi'}\right)} & \text{if } \sin\psi \le \cos\phi' \end{cases} \tag{4.60}$$

and for full angle geometry

$$H(\boldsymbol{\nu}) = \frac{|\boldsymbol{\nu}|}{\pi} \tag{4.61}$$

4.2 Experimental 3D Results

The results presented in this section only cover full angle geometry computed with a modified version of the software package [Toft, 1996], with added extra functionality. The tested algorithms are Filtered Backprojection, Art, Mart and EM. Due to the increasing demand in computer power in 3D compared to 2D the tests are limited to noise free experiments.

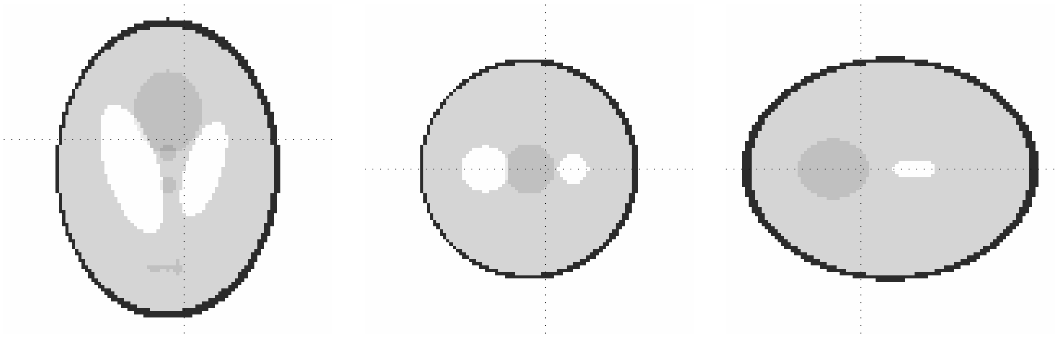

Figure 4.2: 3D version of Shepp-Logan phantom: Axial, Coronal and Sagittal slices.

For the tests a 3D version of the Shepp-Logan phantom is generated, Figure (4.2), so that all of the phantom is inside a square cube $[-1:1] \times [-1:1] \times [-1:1]$.

The algorithms are tested for the influence of the sampling parameters, the sinogram is sampled with parameters $\Delta v = \Delta u = 0.04$, $N_u = N_v = 101$, $N_\theta = N_\phi = 60$. The image sampling parameters are symmetric and in the range $\Delta x = \Delta y = \Delta z \in [0.01 : 0.04]$ and with corresponding size $N_x = N_y = N_z \in [51 : 201]$. The results can be seen in Figure (4.3) and Figure (4.4)

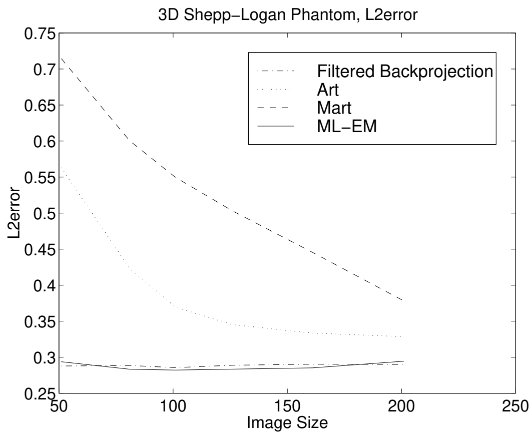

Figure 4.3: The L_2 error as function of image size, for the 3D Shepp-Logan phantom.

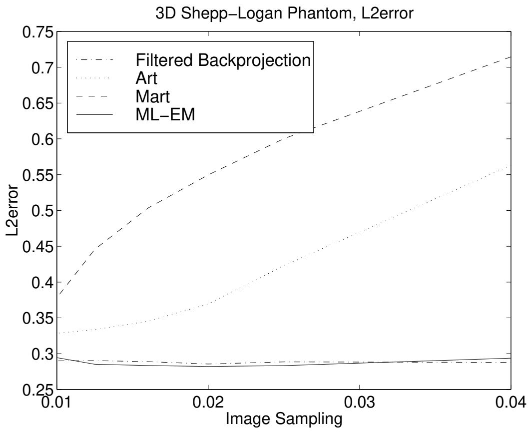

Figure 4.4: The L_2 error as function of image sampling, for the 3D Shepp-Logan phantom.

The results of the different reconstructing methods can be seen in Figure (4.5) through Figure (4.8) with sampling parameters $N_x = N_y = N_z = 101$ and $\Delta x = \Delta y = \Delta z = 0.02$.

As indicated by the L_2 errors the Filtered Backprojection and the EM performs best with the Filtered Backprojection as the visually sharpest. The ART and MART contain more artifacts, which spoils the results. The equal performance of the Filtered Backprojection and the EM in 3D, unlike in 2D, is probably due to a better estimation of the Radon transformation matrix.

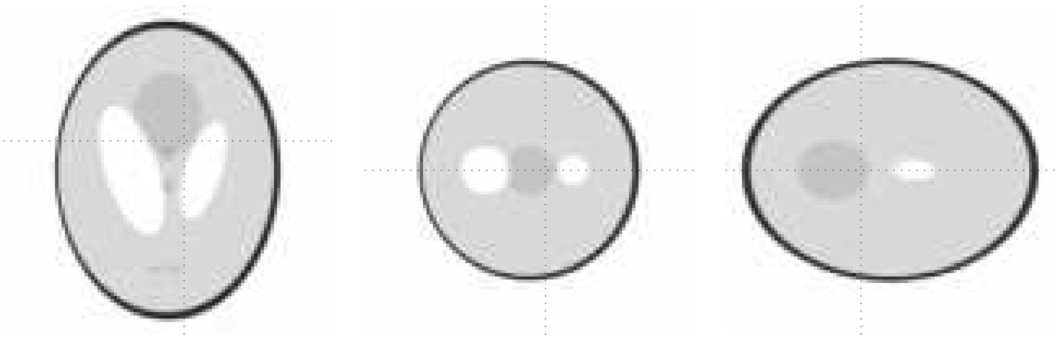

Figure 4.5: 3D version of Shepp-Logan phantom, reconstructed with Filtered Backprojection: Axial, Coronal and Sagittal slices

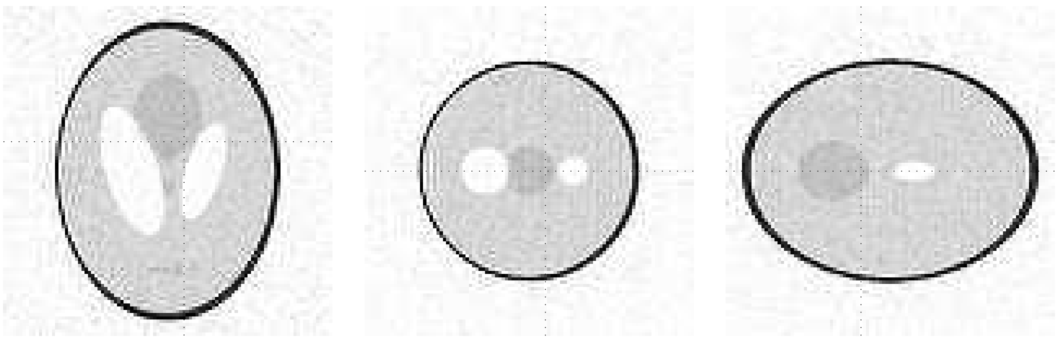

Figure 4.6: 3D version of Shepp-Logan phantom, reconstructed with ART: Axial, Coronal and Sagittal slices

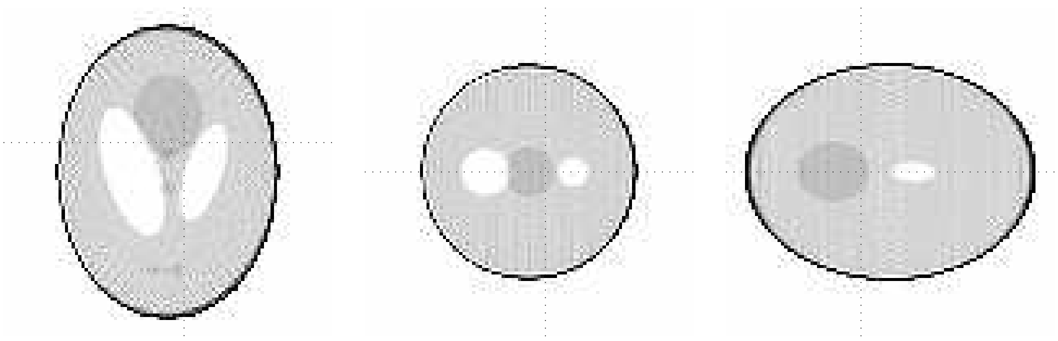

Figure 4.7: 3D version of Shepp-Logan phantom, reconstructed with MART: Axial, Coronal and Sagittal slices

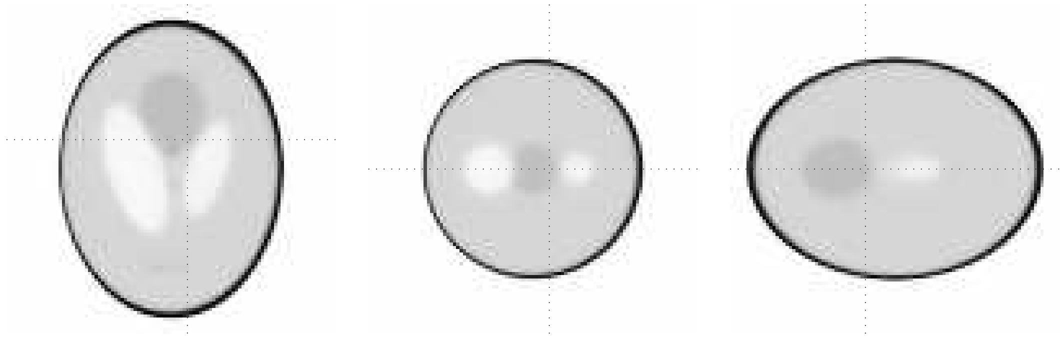

Figure 4.8: 3D version of Shepp-Logan phantom, reconstructed with EM: Axial, Coronal and Sagittal slices

Chapter 5

Bayesian Modeling

The interest in using statistical methods in image reconstruction and image restoration in a Bayesian framework is extensive [Li, 1995, Mumcuoglu et al., 1996, Chen et al., 1990, Zerubia and Chellappa, 1993, Johnson et al., 1991, Geman and Geman, 1984]. In the reconstruction of PET images within the Bayesian framework it is possible to incorporate additional anatomical information from an external source like MR images with the goal of improving the reconstruction [Lipinski et al., 1997, Zhou et al., 1994, Gindi et al., 1995, Ouyang et al., 1994]. This is the background of the work in this chapter and chapter 6.

5.1 Basics of Bayesian Modeling

The basic idea in Bayesian Modeling is to consider all the processes of the imaging system as stochastic processes. This means that both the source (un-degraded) signal and the processes of the imaging system are stochastic processes. The Bayes formula can then be used to obtain the distribution $P(\mathbf{u}|\mathbf{d})$ of the reconstructed signal $\mathbf{u}$, conditioned on the observed degraded signal $\mathbf{d}$:

$$P(\mathbf{u}|\mathbf{d}) = \frac{P(\mathbf{d}|\mathbf{u})P(\mathbf{u})}{P(\mathbf{d})} \propto P(\mathbf{d}|\mathbf{u})P(\mathbf{u}) \tag{5.1}$$

This conditional distribution is the product of the distribution of the imaging system process, also called the observation model: $P(\mathbf{d}|\mathbf{u}) \equiv P(\mathbf{u} \rightarrow \mathbf{d})$, and the *prior* distribution of the reconstructed signal $P(\mathbf{u})$. $P(\mathbf{u}|\mathbf{d})$ of Eq. 5.1 is referred to as the *posterior* distribution. Under a given observation, $\mathbf{d}$, $P(\mathbf{d})$ is constant.

The observation model, $P(\mathbf{d} \rightarrow \mathbf{u})$, includes all external influence on the signal e.g. physical incorrectness in the recording system. This could be several effects like the intrinsic blurring of PET and the Radon transform, in the sense that the observed signal is the sinogram of the PET image.

A useful estimate of the reconstructed signal is given by the *Maximum A Posterior* estimate, e.g. the estimate with the highest probability.

$$\mathbf{u}_{MAP} = \arg\max_{\mathbf{u}} P(\mathbf{u}|\mathbf{d}) = \arg\max_{\mathbf{u}} P(\mathbf{d}|\mathbf{u})P(\mathbf{u}) \tag{5.2}$$

Several methods to find the map estimate exist; in this thesis different annealing schedules to obtain this are examined in section 5.4.

The Prior model of the system is in this context a Markov Random Field Model containing both the intensity values and edge labels, called line-processes by Geman and Geman [Geman and Geman, 1984]. In this model it is possible to incorporate extra information about the desired signal from additional sources. In PET it could be information extracted from MR images, like information on the anatomical edges or direct preprocessing from the reconstructed signal itself. These properties are discussed in chapter 6.

5.2 Markov Random Fields

In this section a description of the prior model, the Markov Random Field (MRF), is given, and some of the properties are analyzed. In [Li, 1995, Johnson et al., 1991, Chen et al., 1990, Geman and Geman, 1984] a more general description of MRF is given. In this thesis MRF's of 2 and 3 dimensions are considered.

The Markovian behavior of the model is expressed as the individual element in $\mathbf{u}$ is only dependent on a neighborhood of elements. The neighborhood is defined as (not) including the element itself and if element $u_{j'}$ is a neighbor to u_j, u_j is always a neighbor to $u_{j'}$. For a 2 dimensional regular lattice the smallest symmetric neighborhood system includes 4 neighbors and is called a 4-neighborhood system. The next neighbor system includes 8 neighbors and so on. A neighborhood can be split up in smaller parts called cliques, and these are used in the mathematical definition.

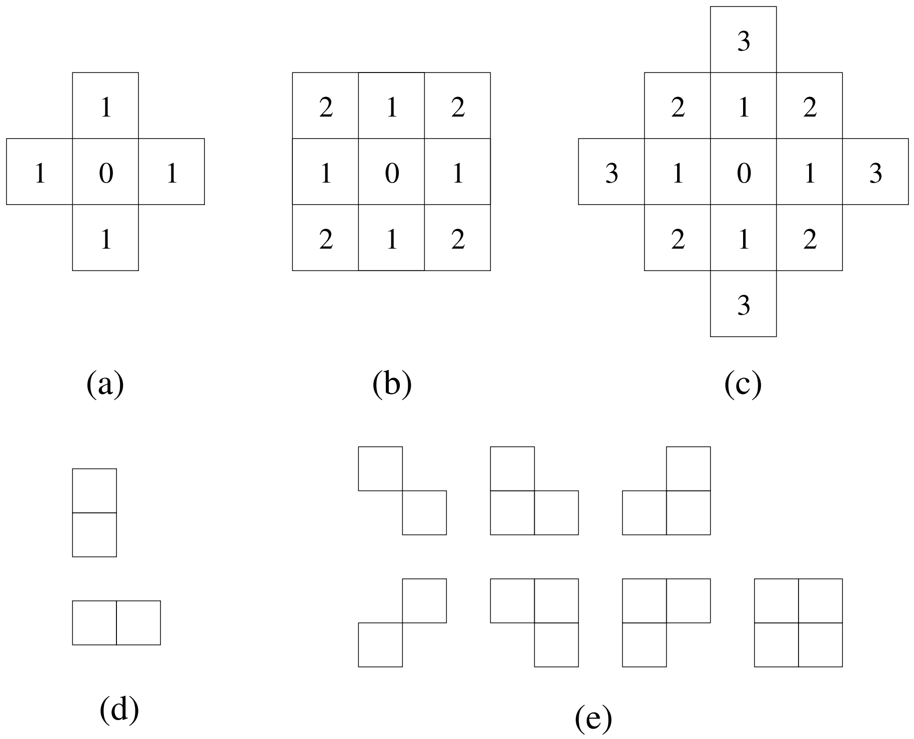

Figure 5.1: Two dimensional neighborhood systems and their corresponding cliques

The three smallest neighborhood systems are shown in Figure (5.1) (a), (b) and (c). The cliques for the 4-neighborhood system are shown in (d), which is only cliques along the axis. The cliques for the 8-neighborhood system (b) are both Figure (5.1) (d) and (e) and contain two diagonal cliques with 2 elements and 4 cliques with 3 elements and one single clique with 4 elements.

In the following only a 4-neighborhood system is considered in two dimensions and a 6-neighborhood system in three dimensions.

The basic Markov Random Field model used in this section is a model with edge elements (also called line-elements or line-processes, e.g. in [Geman and Geman, 1984]), and the edge elements have influence on the individual cliques in a 4-neighborhood system in 2 dimension Figure (5.2) and in the 6-neighborhood system in 3D Figure (5.3).

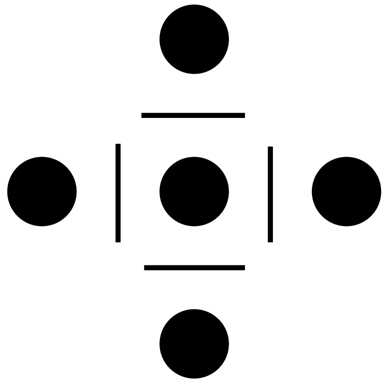

Figure 5.2: 2D 4-neighborhood system with edge elements

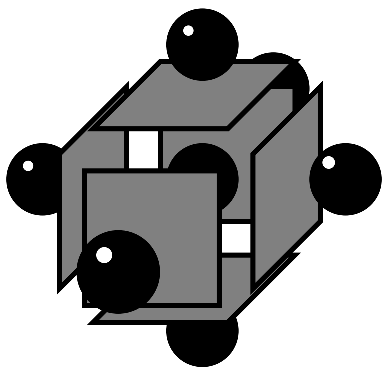

Figure 5.3: 3D 6-neighborhood system with edge elements

The edge elements are labeled binary as { Edge, Non-Edge}, implemented as $\{+1, -1\}$. In 3D the edges are introduced as p_j, q_j and r_j connected to edges in the x, y and z direction. The edge element p_j has the label { Edge} if the signal elements u_j and u_{j+dx} are independent. This corresponds to eliminating the horizontal clique between u_j and u_{j+dx} and gives the possibility to incorporate discontinuities in the image.

Using that the MRF can be written in the term of a non-causal Gibbs Random Field, gives rise to a mathematical expression of the Markov Random Field by defining an energy function for the field [Li, 1995, Hopfield and Tank, 1985]. Denoting all the cliques $\mathbb{C}$ the energy function is

$$E_{cliques} = \sum_{c \in \mathbb{C}} \sum_{j \in \mathbb{C}_c} U_c(\mathbf{u}) \tag{5.3}$$

where $U_c(\mathbf{u})$ is the clique potential depending both on model parameter and the edge elements.

Similar considerations are to be formed for the edge elements and an energy function for the combined the Markov Random Field can be expressed. The coupling between Markov Random Field and Gibbs Random Fields gives the prior distribution a Gibbs form[1],

$$P(\mathbf{u}, \mathbf{p}, \mathbf{q}, \mathbf{r} \,|\, \theta) = Z_1^{-1} \exp\left(-E_{\text{prior}}(\mathbf{u}, \mathbf{p}, \mathbf{q}, \mathbf{r} | \theta)\right), \tag{5.4}$$

where $E_{\text{prior}}(\mathbf{u}, \mathbf{p}, \mathbf{q}, \mathbf{r}|\theta)$ is the energy or cost function for the MRF model, θ are the model parameters and Z_1^{-1} is a normalization constant.

[1]A distribution in the form $P(f) = Z^{-1} \exp\left(-E(f)/T\right)$, where $E(f)$ is a cost-function (bounded from below), T is a parameter and Z a normalization constant, is in Gibbs form.

5.2.1 2D Model

The prior function can be more or less complicated; therefore first a simple 2D model is presented and examined and then this model is expanded. The 2D model can have the following outline for a chosen part of the image

$u_{j-dx+dy}$	$p_{j-dx+dy}$	u_{j+dy}	p_{j+dy}	$u_{j+dx+dy}$
q_{j-dx}		q_j		q_{j+dx}
u_{j-dx}	p_{j-dx}	u_j	p_j	u_{j+dx}
$q_{j-dx-dy}$		q_{j-dy}		$q_{j+dx-dy}$
$u_{j-dx-dy}$	$p_{j-dx-dy}$	u_{j-dy}	p_{j-dy}	$u_{j+dx-dy}$

The smoothness prior in the image is normally implemented as a penalty on the gradients within the defined cliques. The gradient at pixel i along the x-axis is normally defined as $u'_j = (u_{j+dx} - u_j)$, but higher order approximations can be used, and similarly for the y direction. If a constant gradient in the image is wanted the limitation are to be applied on the second order derivative.

For the simple 4-neighborhood system the smoothness prior is implemented as a cost based on the absolute value of the gradient, normally squared here without edge information

$$E_{smooth} = \sum_{j}^{J-1}(u_j - u_{j+dx})^2 + \sum_{j}^{J-1}(u_j - u_{j+dy})^2 \tag{5.5}$$

but others are possible, see overview in [Li, 1995].

The effect of the edge elements is to turn on or off the individual smooth cost. For simplicity the model is first derived without dependence between the edges.

In all the model a simple smoothness between the neighboring intensity value is used, unless an edge-element is active and allows a discontinuity. All through this text a simple second order smoothness is used.

5.2.1.1 Independent Edge Elements.

For independent edges the corresponding label is only dependent on the related intensity value. This assumption means that each dimension is treated independently, but to make the transition as easy as possible a 2D case is examined. The prior energy $E_{main}(\mathbf{u}, p, q|\theta)$ is defined as follows:

$$\begin{aligned} E_{\text{main}}(\mathbf{u},\mathbf{p},\mathbf{q}\,|\,\theta) &= \frac{\nu}{2}\sum_{j}^{J-1}(1-p_j)(u_j - u_{j+dx})^2 + \frac{\nu}{2}\sum_{j}^{J-1}(1-q_j)(u_j - u_{j+dy})^2 \\ &+ \mu^p\sum_{j}^{J-1} p_j + \mu^q \sum_{j}^{J-1} q_j \end{aligned} \tag{5.6}$$

where the model parameters: $\theta = \{\nu, \mu^p, \mu^q\}$ are all positive and $(p_j, q_j) \in \{+1, -1\}$. The prior in Eq. 5.6 is called a weak membrane [Zerubia and Chellappa, 1993].

The first two terms are the smoothness prior in each of the two dimensions. The two last terms control the number of active edges in each of the directions. It is obvious that the energy has a lower boundary at $-(\sum_j \mu^p + \sum_j \mu^q)$ and normal values between $-(\sum_j \mu^p + \sum_j \mu^q)$ and $(\sum_j \mu^p + \sum_j \mu^q)$. Normally the parameters μ^p and μ^q are equal, which gives a simpler model.

A simple explanation of the controlling terms, also called chemical potentials, can be given as follows looking at a single edge element p_j. An edge element is labeled active if the energy is lowered.

$$\begin{aligned} E_{\text{main}}(p_j \text{ active}) &< E_{\text{main}}(p_j \text{ not active}) && (5.7)\\ &\Downarrow \\ \mu^p &< \nu(u_j - u_{j+dx})^2 - \mu^p && (5.8)\\ &\Updownarrow \\ K &> \sqrt{\frac{2\mu^p}{\nu}} && (5.9)\end{aligned}$$

where $K = |u_j - u_{j+dx}|$ is the difference between the two neighboring pixels.

By controlling the parameters μ^p, ν it is possible to control the threshold to make an edge active.

This is illustrated by a small example. A small test image of 3 uniform squares can be seen in Figure (5.4) and the same added Gaussian white noise in Figure (5.5).

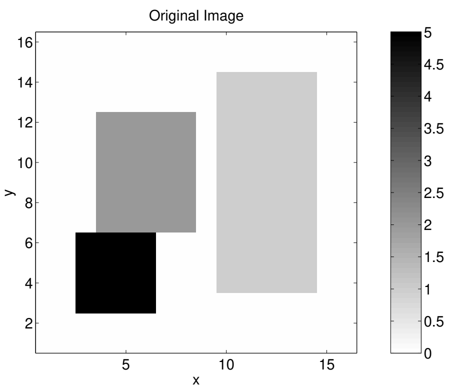

Figure 5.4: Small test image

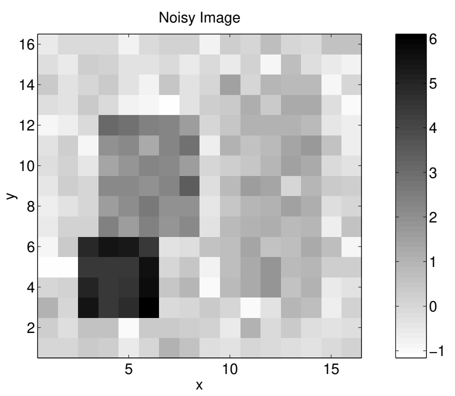

Figure 5.5: Small test image with added white Gaussian noise

The edges are then chosen to obtain lowest energy for variable μ and fixed $\nu = 1$. The number of active edges can be seen in Figure (5.7) and for the non-noisy image all edges have disappeared with $\mu > 12.5$. The contribution from the smoothing terms to the energy can be seen in Figure (5.6).

As expected for the noise free image the edges turn off in steps for rising μ. The steps are placed at the $\mu = (0.5, 2.0, 4.5, 12.5)$ (using Eq. 5.9) as expected. The steps are smoothed out for the noisy image, but it is still possible to determine their existence.

5.2.1.2 Local Connected Edges, Closing Curves

To obtain more closed edge curves several possible solutions have been proposed, see [Zerubia and Chellappa, 1993, Li, 1995, Hertz et al., 1991, Nadabar and Jain, 1996] and [Geman and Geman, 1984]. All of those methods use local connections between the edge elements. Most of those prefer long straight lines connecting each dimension independently

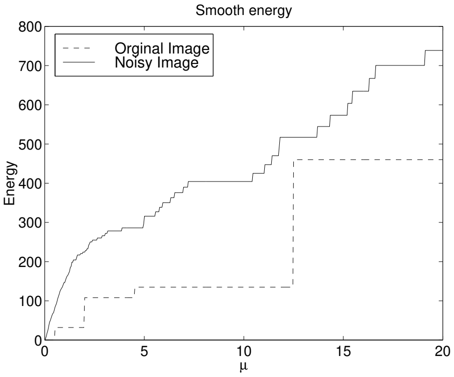

Figure 5.6: Only Smooth Energy as function of μ

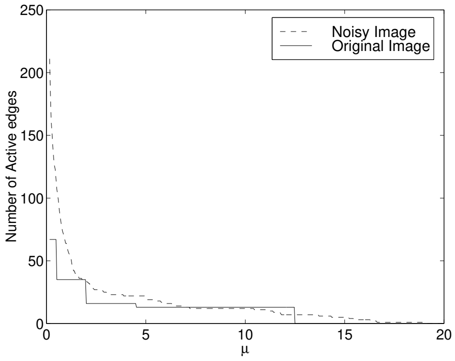

Figure 5.7: Number of active edges as function of μ

[Zerubia and Chellappa, 1993]. In this context only a simple connection term is presented which allows turning edge curves and coupling between the dimensions.

One way to introduce the connection between the local edges is to look at a *cell* in the image.

$$\begin{array}{ccc} u_{j+dy} & p_{j+dy} & u_{j+dx+dy} \\ q_j & & q_{j+dx} \\ u_j & p_j & u_{j+dx} \end{array}$$

Within a cell only a single edge curve is wanted, ending curves or curves splitting up is undesired, meaning 0, 2 or 4 active edges within a cell is desirable, Figure (5.8).

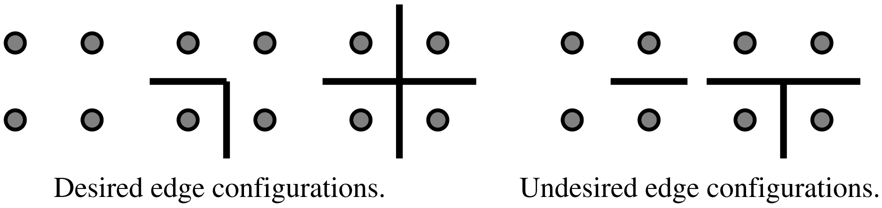

Figure 5.8: Possible edge curve configurations. Active edges are marked with a black line.

It is common to use more complicated structures [Li, 1995, Jeng et al., 1993] to implement the possibility to make a line turn and weight the different curves unevenly; straight lines are preferred to curved lines, but closed curved lines are preferred to broken lines. This extra connection between the edges can be implemented as only one extra term in the energy function if each of the desired configurations is treated equally [Hertz, 1989].

The energy function has the following form:

$$\begin{aligned} E_{\text{prior}}(\mathbf{u},\mathbf{p},\mathbf{q}\,|\,\theta) &= \frac{\nu}{2}\sum_j^J(1-p_j)(u_j-u_{j+dx})^2+\frac{\nu}{2}\sum_j^J(1-q_j)(u_j-u_{j+dy})^2 \\ &+ \mu^p\sum_j^J p_j+\mu^q\sum_j^J q_j-\lambda\sum_j p_jp_{j+dy}q_jq_{j+dx} \end{aligned} \tag{5.10}$$

The extra term can only have the binary values $\{-\lambda, \lambda\}$. Together with μ^p and μ^q it is possible to control the number of edges within the cell, but the parameter values can not be chosen independently. To obtain clarity in the following set $\mu = \mu^p = \mu^q$. With the new connection term the cost of creating an active edge is modeled by both μ and λ.

One way of looking at the parameter dependence is to look at the energy as a function of the number of active edges within one cell. When studying the interaction between the two parameters it is easy to see that it is the ratio which is interesting.

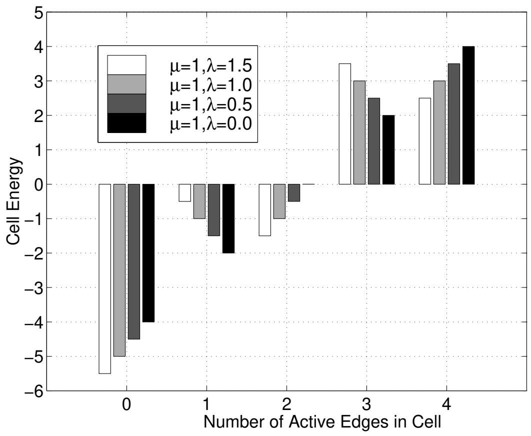

Figure 5.9: Difference in cost function as function of number of active edges within the cell,for $\mu = 1$ and $\lambda = \{0.0, 0.5, 1.0, 1.5\}$

A simple explanation of the controlling terms can again be given as follows, looking at a single edge element p_j. An edge element is labeled active if this gives the lowest energy.

$$\begin{aligned}
E_{\text{prior}}(p_j \text{ active}) &< E_{\text{prior}}(p_j \text{ not active}) && (5.11)\\
&\Downarrow \\
\mu^p - \lambda(p_{j+dy}q_j q_{j+dx} + p_{j-dy}q_{j-dy}q_{j+dx-dy}) &< \nu(u_j - u_{j+dx})^2 - \mu^p && (5.12)\\
&\quad +\lambda(p_{j+dy}q_j q_{j+dx} + p_{j-dy}q_{j-dy}q_{j+dx-dy}) \\
&\Updownarrow \\
\mu^p - \lambda\Gamma_j^p &< \nu(u_j - u_{j+dx})^2 - \mu^p + \lambda\Gamma_j^p && (5.13)\\
K^2 &> \frac{2}{\nu}(\mu^p - \lambda\Gamma_j^p) && (5.14)
\end{aligned}$$

where $\Gamma_j^p = p_{j+dy}q_j q_{j+dx} + p_{j-dy}q_{j-dy}q_{j+dx-dy}$ and $K = |u_j - u_{j+dx}|$ is the difference between the two neighboring pixels. Edge elements can only have the values -1 or 1 and therefore Γ_j^p can only have 3 values $-2, 0, 2$.

$$\Gamma_j^p = \begin{cases} +2 & \text{, if both neighbor cells want an active edge.} \\ 0 & \text{, if one neighbor cells wants an active edge and the other an inactive.} \\ -2 & \text{, if both neighbor cells want an inactive edge.} \end{cases}$$

This gives the following K values

$$
\begin{aligned}
\Gamma_j^p = 2 \quad &: \quad K^- &= &\begin{cases} \sqrt{\frac{2\mu^p}{v}\left[1 - 2\frac{\lambda}{\mu^p}\right]} & , \text{for } \lambda \leq \frac{\mu^p}{2} \\ 0 & , \text{for } \lambda \geq \frac{\mu^p}{2} \end{cases} \\
\Gamma_j^p = 0 \quad &: \quad K^0 &= &\sqrt{\frac{2\mu^p}{v}} \\
\Gamma_j^p = -2 \quad &: \quad K^+ &= &\sqrt{\frac{2\mu^p}{v}\left[1 + 2\frac{\lambda}{\mu^p}\right]}.
\end{aligned} \tag{5.15}
$$

These derived thresholds for creating edges are almost similar to the ones derived in [Zerubia and Chellappa, 1993] in spite of the different neighbor connectivity. Setting λ to high values could make it difficult to control the creation of edges since $K^- = 0$. This make the activation independent of the intensity values, and similar considerations in the other dimension.

To illustrate the effect of the extra λ term consider a small 4×3 image with only two non-zero elements a, b

$$
\boxed{\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & a & b & 0 \\ 0 & 0 & 0 & 0 \end{array}}
$$

For the lowest energy state and if $a > b$ only a few combinations of edge labeling are possible

$$
\underset{(a)}{\boxed{\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & a & b & 0 \\ 0 & 0 & 0 & 0 \end{array}}}
\quad
\underset{(b)}{\boxed{\begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & \boxed{a} & b & 0 \\ 0 & 0 & 0 & 0 \end{array}}}
\quad
\underset{(c)}{\boxed{\begin{array}{ccc} 0 & 0 \quad 0 & 0 \\ 0 & \boxed{a \quad b} & 0 \\ 0 & 0 \quad 0 & 0 \end{array}}}
\quad
\underset{(d)}{\boxed{\begin{array}{ccc} 0 & 0 \quad 0 & 0 \\ 0 & \boxed{a}\boxed{b} & 0 \\ 0 & 0 \quad 0 & 0 \end{array}}}
$$

the energy states of $(a), (b), (c)$ and (d) is dependent on the parameters ν, μ and λ

$$
\begin{aligned}
E_{(a)} &= \nu[3a^2 + 3b^2 + (a-b)^2] - 17\mu - 6\lambda && (5.16) \\
E_{(b)} &= \nu[3b^2] - 9\mu - 6\lambda && (5.17) \\
E_{(c)} &= \nu[(a-b)^2] - 5\mu - 6\lambda && (5.18) \\
E_{(d)} &= -3\mu - 2\lambda && (5.19)
\end{aligned}
$$

and it can be seen by examining the energy differences that the influence of the λ term is only on $E_{(d)}$. In Figure (5.10) and Figure (5.11) the different energy states are marked where they have lowest energy. The effect is that for μ in the range $[0 : 0.45[$ configuration (d) is changed to (c) or (b) if λ is set high enough.

5.2.1.3 Removing Close Edges Lines

The previous model has some tendency to make multiple parallel edge lines at a rising border across several pixels, Figure (5.12).

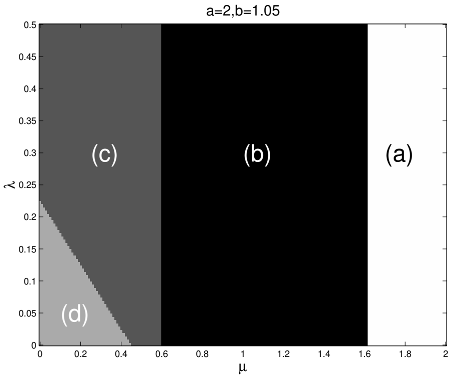

Figure 5.10: Configurations for $a = 2$ and $b = 1.05$ for variant μ and λ

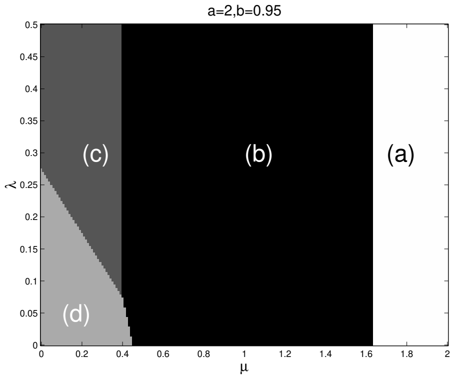

Figure 5.11: Configurations for $a = 2$ and $b = 0.95$ for variant μ and λ

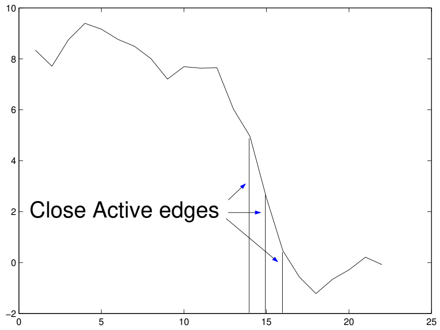

Figure 5.12: Close edges created.

This can (probably) be removed by an extra term controlling the interaction between the edges across a pixel in each dimension.

The energy function will then have the following form:

$$
\begin{aligned}
E_{\text{prior}}(\mathbf{u}, \mathbf{p}, \mathbf{q} \,|\, \theta) \;=\;& \frac{\nu}{2}\sum_j^J (1-p_j)(u_j - u_{j+dx})^2 + \frac{\nu}{2}\sum_j^J (1-q_j)(u_j - u_{j+dy})^2 \\
+\;& \sum_j^J \mu^p p_j + \sum_j^J \mu^q q_j - \sum_j \lambda p_j p_{j+dy} q_j q_{j+dx} \\
+\;& \alpha \sum_j \frac{1}{4}(1+p_j)(1+p_{j+dx}) + \alpha \sum_j \frac{1}{4}(1+q_j)(1+q_{j+dy}) \quad (5.20)
\end{aligned}
$$

The extra term gives an extra cost if, and only if, two neighboring edges across a pixel is active.

p_j	p_{j+dx}	$\frac{1}{4}(1+p_j)(1+p_{j+dx})$
-1	-1	0
-1	1	0
1	-1	0
1	1	1

$$
\begin{aligned}
E_{\text{prior}}(p_j \text{ active}) &< E_{\text{prior}}(p_j \text{ not active}) && (5.21)\\
&\Downarrow \\
\mu^p - \lambda(p_{j+dy}q_jq_{j+dx} + p_{j-dy}q_{j-dy}q_{j+dx-dy}) &< \nu(u_j - u_{j+dx})^2 - \mu^p && (5.22)\\
+\alpha\frac{1}{2}(1+p_{j+dx}) + \alpha\frac{1}{2}(1+p_{j-dx}) & \quad +\lambda(p_{j+dy}q_jq_{j+dx} + p_{j-dy}q_{j-dy}q_{j+dx-dy})\\
&\Updownarrow \\
\mu^p - \lambda\Gamma_j^p + \alpha\Lambda_j^p &< \nu(u_j - u_{j+dx})^2 - \mu^p + \lambda\Gamma_j^p && (5.23)\\
K^2 &> \frac{2}{\nu}(\mu^p + \frac{\alpha}{2}\Lambda_j^p - \lambda\Gamma_j^p) && (5.24)
\end{aligned}
$$

where $\Lambda_j^p = 1 + \frac{1}{2}(p_{j+dx} + p_{j-dx})$, $\Gamma_j^p = p_{j+dy}q_jq_{j+dx} + p_{j-dy}q_{j-dy}q_{j+dx-dy}$ and $K = |u_j - u_{j+dx}|$ is the difference between the two neighboring pixels. Edge elements can only have the values -1 or 1 and therefore Λ_i^p can only have 3 values $0, 1, 2$.

$$
\Lambda_j^p = \begin{cases} 2 & \text{, if both neighbor edges are active.}\\ 1 & \text{, if the neighbor edges are active and inactive.}\\ 0 & \text{, if both neighbor edges are inactive.} \end{cases}
$$

This extra term can be seen as a modification of μ_j^p which is as follows with $\lambda = 0$:

$$
\begin{aligned}
\Lambda_j^p = 2 &: K_2 = \sqrt{\tfrac{2}{\nu}(\mu^p + \alpha)}\\
\Lambda_j^p = 1 &: K_1 = \sqrt{\tfrac{2}{\nu}(\mu^p + \tfrac{\alpha}{2})} && (5.25)\\
\Lambda_j^p = 0 &: K_0 = \sqrt{\tfrac{2\mu^p}{v}}
\end{aligned}
$$

As it can be seen in Eq. 5.25, changing the parameter α controls the threshold to create active edges.

One of the drawbacks of this extra edge control is that the minimum size of the allowed objects is increased from 1×1 to 2×2. Another problem arising in PET images is that larger regions of gray matter connected by a thin track can be separated.

5.2.1.4 Spatially Dependent Parameters

The MRF model can be designed to have a more flexible behavior by making the parameters spatially dependent [Hansen et al., 1996]. The dependency can be controlled by extra prior perhaps from another observed signal.

$$
\begin{aligned}
E_{\text{prior}}(\mathbf{u}, \mathbf{p}, \mathbf{q}, \mathbf{r} \,|\, \theta) &= \frac{\nu}{2}\sum_j (1-p_j)(u_j - u_{j+dx})^2 + \frac{\nu}{2}\sum_j (1-q_j)(u_j - u_{j+dy})^2\\
&+ \sum_j \mu_j^p p_j + \sum_j \mu_j^q q_j - \sum_j \lambda_j p_j p_{j+dy} q_j q_{j+dx} && (5.26)\\
&+ \sum_j \frac{\alpha_j}{4}(1+p_j)(1+p_{j+dx}) + \sum_j \frac{\alpha_j}{4}(1+q_j)(1+q_{j+dy})
\end{aligned}
$$

This introduces many free parameters and this can be a problem at the point of estimating the parameters.

One simple way to reduce the parameter space can be done as follows: $\mu_j^p = \mu^p(1 - K_j)$,$\mu_j^q = \mu^p(1 - K_j)$,$\lambda_j = \lambda(1 - K_j)$,..., where $K_j \in [0:1]$.

Using the spatial dependency in this manner it it possible to incorporate anatomical priors from a MR scan in reconstruction/restoration of a PET scan.

5.2.2 3D Model

The previously described model can be extended to three dimensions, by making edge connection terms like in Eq. 5.10 for each dimension possible, xy, xz, yz.

This gives the following energy function

$$\begin{aligned} E_{\text{prior}}(\mathbf{u}, \mathbf{p}, \mathbf{q}, \mathbf{r} \,|\, \theta) &= \frac{\nu}{2}\sum_j (1 - p_j)(u_j - u_{j+dx})^2 + \frac{\nu}{2}\sum_j (1 - q_j)(u_j - u_{j+dy})^2 \\ &+ \frac{\nu}{2}\sum_j (1 - r_j)(u_j - u_{j+dz})^2 \\ &+ \sum_j \mu_j^p p_j + \sum_j \mu_j^q q_j + \sum_j \mu_j^r r_j - \sum_j \lambda_j^{xy} p_j p_{j+dy} q_j q_{j+dx} \\ &- \sum_j \lambda_j^{xz} p_j p_{j+dz} r_j r_{j+dz} - \sum_j \lambda_j^{yz} q_j q_{j+dz} r_j r_{j+dy} \\ &+ \sum_j \frac{\alpha_j}{4}(1 + p_j)(1 + p_{j+dx}) + \sum_j \frac{\alpha_j}{4}(1 + q_j)(1 + q_{j+dy}) \\ &+ \sum_j \frac{\alpha_j}{4}(1 + r_j)(1 + r_{j+dz}) \end{aligned} \tag{5.27}$$

The model can be expressed in a more simple form by using a vector description

$$E_{\text{prior}}(\mathbf{u}, \mathbf{p}, \mathbf{q}, \mathbf{r} \,|\, \theta) = \mathbf{u}^T \Sigma_{MRF} \mathbf{u} + E_{MRF-edge-control} \tag{5.28}$$

where $E_{MRF-edge-control}$ are the terms controlling the generation of the active edges. Σ_{MRF} is a symmetric band structured matrix describing the neighbor connectivity based on the active cliques.

This is the MRF model used in the rest of this thesis with parameters set to $\lambda_j^{xy} = \lambda_j^{xz} = \lambda_j^{yz} = \lambda_j$ and $\mu_j^p = \mu_j^q = \mu_j^r = \mu_j$.

5.3 Observation Model - Degeneration Process

The observation model: $P(\mathbf{d}|\mathbf{u}) \equiv P(\mathbf{u} \to \mathbf{d})$ in Eq. 5.1 is the model describing how the observed data is created from $\mathbf{u}$. In general a non-linear transformation of $\mathbf{u}$ is examined, called $\vartheta(\cdot)$ and describing the mapping $\mathcal{R}^J \xrightarrow{\vartheta} \mathcal{R}^I$.

The distribution may be calculated as

$$p(\mathbf{d}|\mathbf{u}) = p_{noise}(\mathbf{d} - \vartheta(\mathbf{u})) \tag{5.29}$$

where $p_{noise}(\mathbf{e})$ is the noise distribution of the residual $\mathbf{e} = \mathbf{d} - \vartheta(\mathbf{u})$.

In the following only linear transformation $\varphi(\cdot)$ is considered. The linear transformations can be the forward Radon transformation combined with blurring used to model the forward PET model for reconstruction, $\varphi(\mathbf{u}) = \mathbf{RBu}$ where $\mathbf{B}$ is the blurring and $\mathbf{R}$ the

Radon transform. Or the linear transformation can be a blurring transformation $\mathbf{B}$ to use in image enhancement. In the first case the observed signal is the sinogram, in the second a blurred observation of the original image, each of them observed with noise. In the simplest case the transformation is just the identity transformation and is useful for investigating the system model.

5.3.1 Gaussian Noise

First we consider the linear transformation $\varphi(\cdot)$ and the noise is additive spatially independent, white, Gaussian noise.

We look at the observation model for each point d_i of $\mathbf{d}$.

$$d_i = \varphi_i(\mathbf{u}) + e_i \tag{5.30}$$

where $\varphi_i(\mathbf{u}) = \sum_j \varphi_{ij} u_j$ and $e_i \in \mathcal{N}(0, \sigma_i^2)$ are independent for each d_i.

Inserting in Eq. 5.29 gives

$$\begin{aligned} P(d_i|\mathbf{u}) &= \int \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{e_i^2}{2\sigma_i^2}\right) \delta(d_i - \varphi_i(\mathbf{u}) + e_i) de_i && (5.31) \\ &= \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{(d_i - \varphi_i(\mathbf{u}))^2}{2\sigma_i^2}\right) && (5.32) \end{aligned}$$

Since each of the $P(d_i|\mathbf{u})$ is independent

$$\begin{aligned} P(\mathbf{d}|\mathbf{u}) &= \prod_i p(d_i|\mathbf{u}) && (5.33) \\ &= \prod_i \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\sum_i \frac{(d_i - \varphi_i(\mathbf{u}))^2}{2\sigma_i^2}\right) && (5.34) \\ &= Z_2 \exp\left(-\sum_i \frac{(d_i - \varphi_i(\mathbf{u}))^2}{2\sigma_i^2}\right) && (5.35) \end{aligned}$$

where $Z_2 = \prod_i \frac{1}{\sqrt{2\pi\sigma_i^2}}$. This is already in Gibbs form with

$$E_{Gaussnoise} = \sum_i \frac{(d_i - \varphi_i(\mathbf{u}))^2}{2\sigma_i^2} \tag{5.36}$$

or expressed in vector form

$$E_{Gaussnoise} = [\mathbf{d} - \varphi(\mathbf{u})]^T \Sigma_{Gauss} [\mathbf{d} - \varphi(\mathbf{u})] \tag{5.37}$$

where $\Sigma_{Gauss} = diag(\frac{1}{2\sigma_1^2}, \ldots, \frac{1}{2\sigma_i^2}, \ldots, \frac{1}{2\sigma_I^2})$

5.3.2 Poisson Based Observed Data

Like for the EM the likelihood $P(\mathbf{d}|\mathbf{u})$ can be written as

$$P(\mathbf{d}|\mathbf{u}) = \prod_{i=1}^{I} \frac{(d_i^*)^{d_i}}{d_i!} e^{-d_i^*} \tag{5.38}$$

with d_i^* being the mean value of the Poisson process generating d_i. d_i^* is determinant by the linear transformation

$$\mathbf{d}^* = \varphi(\mathbf{u}) \tag{5.39}$$

Using Poisson distribution in map estimation can give some problems since the variance is dependent on the observed value. In [Rao, 1973] a square root transformation is used to obtain approximately independent variance. Using the transformation $\sqrt{d_i + k}$, with k as a constant to stabilize the transformation, the distribution can be approximated with a Gaussian with mean $\sqrt{d_i^* + k}$ and variance $\frac{1}{4}$ in first order approximation. For the parameter setting $k = 0$ the likelihood function gives

$$P(\mathbf{d}|\mathbf{u}) \propto \exp\left[-2\sum_{i=0}^{I}(\sqrt{d_i} - \sqrt{d_i^*})^2\right] \tag{5.40}$$

In [Johnson et al., 1991] Eq. 5.40 is transformed into the domain of the desired image by introducing a latent image, which is the particular Poisson sample generation of the observed data. This method, called Iterated Conditional Average (ICA), can be expressed as a combination of the ML-EM algorithm and a Mean Field Approximation.

For both Gaussian and Poisson based observation systems the distribution can then be described in Gibbs form

$$P(\mathbf{d}|\mathbf{u}) \propto \exp\left[-E_{observation}\right] \tag{5.41}$$

with $E_{observation}$ corresponding to the actual observation model.

5.4 System Model - Finding MAP Estimate

The model of the total system can now be expressed as

$$P(\mathbf{u}|\mathbf{d}) \propto \exp\left[-(E_{observation} + E_{prior})\right] \tag{5.42}$$

where the normalization constant can be ignored and the map estimate of Eq. 5.42 corresponds to finding the minimum of the energy function.

The energy function usually has many local minimums, and methods based only on gradient decent do not find the optimal solution. Two methods are often used to deal with the local minimum problem: random search and annealing. In random search methods, like the Metropolis algorithm [Metropolis et al., 1953], the new configuration does not always make an energy descent, but occasional energy increase is allowed, and this strategy hopefully helps to get out of a local minimum and into the global minimum.

In annealing an extra parameter is introduced T, called the temperature, defining a new posterior probability function

$$P_T(\mathbf{u}|\mathbf{d}) \propto (P(\mathbf{u}|\mathbf{d}))^{\frac{1}{T}}, \quad T > 0 \tag{5.43}$$

For high values of T the energy landscape is convex and local methods can be used to find the maximum of $P_T(\mathbf{u}|\mathbf{d})$ and then lowering the temperature while following the maximum could help to find the global maximum. It should be noticed that for $T \to \infty$, $P_T(\mathbf{u}|\mathbf{d})$ is a uniform distribution and for $T \to 0$, $P_T(\mathbf{u}|\mathbf{d})$ is concentrated on the peaks of $P(\mathbf{u}|\mathbf{d})$.

The annealing schedules can be divided into two parts: deterministic, like Mean Field annealing [Peterson and Söderberg, 1989], and stochastic. One of the most common stochastic annealing schedules is simulated annealing [Geman and Geman, 1984].

Several different methods to find the map estimate, i.e. the minimum of the energy function, exist. Local methods like gradient descent will not be discussed in this context but only global methods.

5.4.1 Metropolis Sampling

In the original Metropolis algorithm [Metropolis et al., 1953] the idea is to generate a sequence of estimates, each new estimate generated close to one before. The new estimate is accepted if the energy is lowered, and if the energy is increased the new state is accepted with the probability $\exp(-\Delta E/T)$, where ΔE is the change in energy, all at a fixed temperature. The algorithm can be simplified to

---

initialize $\mathbf{f}$
repeat
  get random $i \in I$
  generate f_i' close to f_i
  $\Delta E = E(\mathbf{f_i}') - E(\mathbf{f_i})$
  if $\Delta E < 0$ **then**
    $f_i \leftarrow f_i'$
  else
    $P = \exp(-\Delta E/T)$
    if $random[0;1[< P$ **then**
      $f_i \leftarrow f_i'$
    end if
  end if
until equilibrium

This algorithm can be applied to work within an annealing schedule. In the case of the two dimensional MRF in Eq. 5.10 and Gaussian based noise in the observation model Eq. 5.36 the algorithm step can be as follows. The generating step is easy to implement for independent edge elements, the edge element is toggled and the energy difference for $p_j : -1 \to 1$ can be calculated as

$$\Delta E(p_j : -1 \to 1) = 2(\mu^p - \lambda\Gamma_j^p) - \nu(u_j - u_{j+dx})^2 \tag{5.44}$$

with $\Delta E(p_j : 1 \to -1) = -\Delta E(p_j : -1 \to 1)$ and similarly for q_j. For the image point u_j the new estimate is u_j'. Defining $\Delta u_j = u_j' - u_j$

$$\begin{aligned}\Delta E(u_j : u_j \to u_j') = \sum_i \frac{1}{2\sigma_i^2}[\varphi_i(\mathbf{1}_j\Delta u_j)](\varphi_i(\mathbf{1}_j\Delta u_j) + 2[\varphi_i(\mathbf{u}) - d_i]) \\ +\nu(1-p_j)\Delta u_j(u_j' + u_j - u_{j-dx} - u_{j+dx}) \\ +\nu(1-q_j)\Delta u_j(u_j' + u_j - u_{j-dy} - u_{j+dy})\end{aligned} \tag{5.45}$$

where $\mathbf{1}_j$ is a vector with zero elements except element j which is set to 1. This is a very time consuming using sequential update off all images and edge values, but speedup can be achieved by having parallel update. Simulated annealing can be applied by using a sequence of decreasing temperatures where the Metropolis algorithm is used for each temperature.

5.4.2 Gibbs Sampling - Simulated Annealing

The Gibbs sampler [Geman and Geman, 1984] generates the new estimate based on the conditional distribution and not on the energy change like in the Metropolis algorithm Section 5.4.1. The updating scheme can both be sequential and parallel. The basis is to calculate a new estimate for a fixed temperature and then lower the temperature.

```
initialize f and T
repeat
  for all i do {Can be evaluated in parallel}
    generate sample f'_i from p(f_i|f_k, k ∈ N_neighborhood)
  end for
  f_i ← f'_i
  Decrease T
until equilibrium, T → 0
```

Again in the case of the two dimensional MRF in Eq. 5.10 and Gaussian based noise in the observation model Eq. 5.36 algorithm step can be as follows. The posterior distribution $P_T(\mathbf{u}|\mathbf{d})$ in Eq. 5.43 is a Gauss distribution for the signal values $\mathbf{u}$ and with mean value $\bar{\mathbf{u}}$ and variance $\Sigma_{\mathbf{u}}$, which is a diagonal matrix. For the individual elements u_j the conditional posterior distribution term can be expressed as

$$P_T(u_j|\mathbf{u}/\{u_j\}, \mathbf{d}, \mathbf{p}, \mathbf{q}, \mathbf{r}) = \frac{1}{Z_{u_j}} \exp(-\frac{1}{T} E(u_j|\mathbf{u}/\{u_j\}, \mathbf{d}, \mathbf{p}, \mathbf{q}, \mathbf{r})) \tag{5.46}$$

where

$$\begin{aligned} E(u_j|\mathbf{u}/\{u_j\}, \mathbf{d}, \mathbf{p}, \mathbf{q}, \mathbf{r}))E_{u_j} &= E_{obs+prior} && (5.47)\\ &= E_{u_j} + \text{terms independent of } u_i && (5.48) \end{aligned}$$

and

$$\begin{aligned} E_{u_j} &= \sum_i \frac{(d_i - \varphi_i(\mathbf{u}))^2}{2\sigma_i^2} + \frac{\nu}{2}\sum_j (1-p_j)(u_j - u_{j+dx})^2 \\ &+ \frac{\nu}{2}\sum_j (1-q_j)(u_j - u_{j+dy})^2 && (5.49) \end{aligned}$$

The normalization constant Z_{u_j} can be calculated using

$$1 = \int P_T(u_j|\mathbf{u}/\{u_j\}, \mathbf{d}, \mathbf{p}, \mathbf{q}, \mathbf{r})\, du_j \tag{5.50}$$

It can be seen that E_{u_j} is a quadric form like $x = Ax^2 + Bx + C$ and this gives

$$\begin{aligned} \bar{x} &= \frac{\int x \, \exp(-Ax^2 - Bx - C)\, dx}{\int \exp(-Ax^2 - Bx - C)\, dx} && (5.51)\\ &= \frac{-B}{2A} && (5.52) \end{aligned}$$

This is also the MAP-solution for x because the global minimum of $Ax^2 + Bx + C$ is $x = \frac{-B}{2A}$. The variance can be found to

$$Var(x) = \frac{1}{2A} \tag{5.53}$$

Using the quadric form inserting in Eq. 5.49 and using Eq. 5.50 gives

$$\begin{aligned}\bar{u}_j &= \frac{1}{\kappa}\Bigg[\sum_i \frac{1}{\sigma_i^2}\varphi_{ji}(d_i - \varphi_i(\mathbf{u})) + (\sum_i \frac{1}{\sigma_i^2}\varphi_{ji}^2)u_j \qquad (5.54)\\ &\quad +\nu[(1-p_j)u_{j+dx} + (1-p_{j-dx})u_{j-dx} + (1-q_j)u_{j+dy} + (1-q_{j-dy})u_{j-dy}]\Bigg]\end{aligned}$$

which is independent of the temperature and variance

$$Var(u_j) = \frac{1}{T}\ \kappa \tag{5.55}$$

where

$$\kappa = \sum_i \frac{1}{\sigma_i^2}\varphi_{ji}^2 + \nu(4 - p_j - p_{j-dx} - q_j - q_{j-dy}) \tag{5.56}$$

The new estimate can then be sampled from $N(\bar{u}_j, Var(u_j))$. For the edge element p_j the conditional posterior distribution can be formulated as

$$P_T(p_j|\mathbf{u},\mathbf{d},\mathbf{p}/\{p_j\},\mathbf{q},\mathbf{r}) = \frac{1}{Z_{p_j}}\exp(-\frac{1}{T}E(p_j|\mathbf{u},\mathbf{d},\mathbf{p}/\{p_j\},\mathbf{q},\mathbf{r})) \tag{5.57}$$

where

$$\begin{aligned}E(p_j|\mathbf{u},\mathbf{d},\mathbf{p}/\{p_j\},\mathbf{q},\mathbf{r})) &= E_{obs+prior} &(5.58)\\ &= E_{p_j} + \text{terms independent of } p_j &(5.59)\end{aligned}$$

and

$$\begin{aligned}E_{p_j} &= \frac{\nu}{2}(1-p_j)(u_j - u_{j+dx})^2 + \mu_j^p p_j &(5.60)\\ &- \lambda p_j p_{j+dy} q_j q_{j+dx} - \lambda p_{j-dy} p_j q_{j-dy} q_{j+dx-dy} &\\ & &(5.61)\end{aligned}$$

Again the normalization constant can be found using

$$1 = P_T(p_j = +1|\mathbf{u},\mathbf{d},\mathbf{p}/\{p_j\},\mathbf{q},\mathbf{r}) + P_T(p_j = -1|\mathbf{u},\mathbf{d},\mathbf{p}/\{p_j\},\mathbf{q},\mathbf{r}) \tag{5.62}$$

The probability $P_T(p_j = +1) = P_T(p_j = +1|\mathbf{u},\mathbf{d},\mathbf{p}/\{p_j\},\mathbf{q},\mathbf{r})$ can be calculated as

$$P_T(p_j = +1) = \frac{1}{1 + \exp(-\frac{1}{T}[\nu(u_j - u_{j+dx})^2 - 2\mu^p + 2\lambda\Gamma_j^p])} \tag{5.63}$$

with $\Gamma_j^p = p_{j+dy}q_j q_{j+dx} + p_{j-dy}q_{j-dy}q_{j+dx-dy}$. The new estimate is $p_j = +1$ if

$$P_T(p_j = +1) < \text{Random number } p \in [0:1[\tag{5.64}$$

and $p_j = -1$ otherwise, and similarly for q_j. For high temperatures $P_T(p_j = +1) \approx \frac{1}{2}$ and the edge elements will often flip, a chaotic system. And for $T \to 0$ the edge label will be determined by the sign of $[\nu(u_j - u_{j+dx})^2 - 2\mu^p + 2\lambda\Gamma_j^p]$, a deterministic system. The annealing schedule will be discussed in 5.4.4.

5.4.3 Mean Field Annealing

The basic idea behind the Mean Field Approximation is to replace the single element with its mean value conditioned on the neighboring elements [Peterson and Söderberg, 1989]. For high temperatures the system is chaotic and for Gaussian shape posterior distribution the mean value of the image elements represents the map estimate. The binary labels -1 and $+1$ of the edge elements are replaced with a continued value in the interval $[-1;+1]$ ending with binary at temperature $T \to 0$. To ensure binary edge elements the a measure called *saturation* is defined in [Peterson and Söderberg, 1989], here in two dimensions

$$\Sigma_{saturation} = \frac{1}{J}\sum_j p_j^2 + \frac{1}{J}\sum_j q_j^2 \tag{5.65}$$

as it must satisfy $\Sigma_{saturation} = 1$, but normally accepted if $\Sigma_{saturation} > 0.99$. The algorithm can then be expressed as

---

initialize $\mathbf{f}$ and T
repeat
  for all i **do** {Can be evaluated in parallel}
    calculate mean value f_i' from $p(f_i|f_k, k \in \mathbb{N}_{neighborhood})$
  end for
  $\mathbf{f}_i \leftarrow \mathbf{f}_i'$
  Decrease T
  calculate saturation $\Sigma_{saturation}$
until $\Sigma_{saturation} > 0.99$

The mean value of the image elements can calculated like in Eq. 5.54

$$\begin{aligned}\bar{u}_j' &= \frac{1}{\kappa_j}\Bigg[\sum_i \frac{1}{\sigma_i^2}\varphi_{ji}(d_i - \varphi_i(\bar{\mathbf{u}})) + (\sum_i \frac{1}{\sigma_i^2}\varphi_{ji}^2)\bar{u}_j \\ &\quad +\nu[(1-p_j)\bar{u}_{j+dx} + (1-p_{j-dx})\bar{u}_{j-dx} + (1-q_j)\bar{u}_{j+dy} + (1-q_{j-dy})\bar{u}_{j-dy}]\Bigg]\end{aligned} \tag{5.66}$$

with

$$\kappa_j = \sum_i \frac{1}{\sigma_i^2}\varphi_{ji}^2 + \nu(4 - p_j - p_{j-dx} - q_j - q_{j-dy}) \tag{5.67}$$

The mean value of the edge elements can be calculated as

$$\begin{aligned}\bar{p}_j' &= \frac{+1\cdot P_T(p_j=+1|\mathbf{u},\mathbf{d},\mathbf{p}/\{p_j\},\mathbf{q},\mathbf{r}) - 1\cdot P_T(p_j=-1|\mathbf{u},\mathbf{d},\mathbf{p}/\{p_j\},\mathbf{q},\mathbf{r})}{P_T(p_j=+1|\mathbf{u},\mathbf{d},\mathbf{p}/\{p_j\},\mathbf{q},\mathbf{r}) + P_T(p_j=-1|\mathbf{u},\mathbf{d},\mathbf{p}/\{p_j\},\mathbf{q},\mathbf{r})} \\ &= \tanh(\frac{1}{T}[\nu(u_j - u_{j+dx})^2 - 2\mu^p + 2\lambda\Gamma_j^p])\end{aligned} \tag{5.68}$$

where all the mean value expressions $\bar{f}$ on the right hand sides are omitted.

5.4.4 Annealing Schedules

In the literature several different annealing schedules are proposed [Geman and Geman, 1984, Peterson and Söderberg, 1989, Li, 1995]. In [Geman and Geman, 1984] it is proved that simulated annealing converges to the global minimum if the decreasing sequence of temperatures satisfy

$$T^{(k)} \to 0 \text{ for } k \to \infty \tag{5.69}$$

and

$$T^{(k)} > \frac{N\Delta}{\log(k)} \text{ for } k \geq k_0, \text{ with } k_0 \text{ as an integer } \geq 2 \tag{5.70}$$

where $\Delta = \max_f E(f) - \min_f E(f)$ and N is the total number of elements in the system. But as mentioned in [Geman and Geman, 1984] $N\Delta$ is too big for any practical purpose and they use

$$T^{(k)} = \frac{C}{\log(1+k)} \quad , 1 \leq k \leq K \tag{5.71}$$

with values of C at 3.0 to 4.0 which corresponds to a starting temperature around $T^{(0)} \approx 5$. For annealing in [Li, 1995] the temperature schedule used is

$$T^{(k)} = \kappa^{(k)} T^{(k-1)} \tag{5.72}$$

where

$$\kappa^{(k)} = \begin{cases} 0.9 & , \text{ if } \Sigma_{saturation} < 0.9 \\ 0.95 & , \text{ if } 0.9 < \Sigma_{saturation} < 0.95 \\ 0.99 & , \text{ if } 0.95 < \Sigma_{saturation} \end{cases} \tag{5.73}$$

which is based on the proposed schedule in [Peterson and Söderberg, 1989] which uses

$$T^{(k)} = 0.9\, T^{(k-1)} \tag{5.74}$$

The next problem is to determine the initial temperature $T^{(0)}$ which has to be higher that the critical temperature T_c for which the system changes phase from chaotic to deterministic. In [Peterson and Söderberg, 1989] the critical temperature is estimated for the Graph Partition (GP) case and for the Traveling Salesman Problem (TSP) using an estimate of the largest and smallest eigenvalues of the system.

5.4.4.1 Critical Temperature

To estimate the critical temperature a one dimensional Markov field is examined using mean field approximation. The system is stabilized at a high temperature such that $\bar{p}_j$ is in the linear region of tanh. The stability is examined by looking at small fluctuations of $\bar{p}_j$ around the fix point. The system is more complicated than in [Peterson and Söderberg, 1989] since the influence of the fluctuations is indirect through the image values $\bar{\mathbf{u}}$. The mean value will be omitted in the following. The fluctuation for edge element p_j, δp_j can be expressed using first order Taylor expansion

$$\delta p'_j = \sum_l \frac{\partial p_j}{\partial p_l} \delta p_l \tag{5.75}$$

where $\frac{\partial p_j}{\partial p_l}$ is calculated in the stable point. The edge update is

$$p_j = \tanh(\frac{1}{T}[\nu(u_j - u_{j+dx})^2 - 2\mu^p]) \tag{5.76}$$

and this leads to

$$\frac{\partial p_j}{\partial p_l} = (1+p_j^2)\frac{1}{T}[2\nu(u_j - u_{j+dx})](\frac{\partial u_j}{\partial p_l} - \frac{\partial u_{j+dx}}{\partial p_l}) \tag{5.77}$$

The next step is to evaluate $\frac{\partial u_j}{\partial p_l}$ using a Gaussian observation model with a unity transformation ($\mathbf{u} = \varphi(\mathbf{u})$) and spatially independent variance $\sigma_i^2 = \sigma^2$. In the 1D case the image update is

$$u_j \quad = \quad \frac{1}{\kappa_j}\left[\frac{1}{\sigma^2}d_i + \nu[(1-p_j)u_{j+dx} + (1-p_{j-dx})u_{j-dx}]\right]$$

with

$$\kappa_j = \frac{1}{\sigma^2} + \nu(2 - p_j - p_{j-dx}) \tag{5.78}$$

it can be seen that u_j is only dependent on p_j and p_{j-dx} and this leads to $\frac{\partial p_j}{\partial p_l}$ being nonzero for $l = \{j-dx, j, j+dx\}$. Using

$$\frac{\partial u_j}{\partial p_j} = \nu\frac{1}{\kappa_j}[u_j - u_{j+dx}] \tag{5.79}$$

and

$$\frac{\partial u_j}{\partial p_{j-dx}} = \nu\frac{1}{\kappa_j}[u_j - u_{j-dx}] \tag{5.80}$$

gives

$$\delta p'_j = m_{j,j-dx}\delta p_{j-dx} + m_{j,j}\delta p_j + m_{j,j+dx}\delta p_{j+dx} \tag{5.81}$$

with

$$\begin{aligned}
m_{j,j-dx} &= (1+p_j^2)\frac{1}{T}[2\nu(u_j - u_{j+dx})](\nu\frac{1}{\kappa_j}[u_j - u_{j-dx}]) && (5.82)\\
m_{j,j} &= (1+p_j^2)\frac{1}{T}[2\nu(u_j - u_{j+dx})](\frac{\nu}{\kappa_j}[u_j - u_{j-dx}] - \frac{\nu}{\kappa_{j+dx}}[u_{j+dx} - u_j]) && (5.83)\\
m_{j,j+dx} &= (1+p_j^2)\frac{1}{T}[2\nu(u_j - u_{j+dx})](\nu\frac{1}{\kappa_{j+dx}}[u_{j+2dx} - u_{j+dx}]) && (5.84)
\end{aligned}$$

In matrix notation this can be expressed as

$$\delta\mathbf{p}' = \mathbf{M}\delta\mathbf{p} \tag{5.85}$$

where $\mathbf{M}$ is a tridiagonal matrix with elements described above. The stability is dominated by the eigenvector that corresponds to the numerically largest eigenvalue.

Simplified for high temperatures $p_j \approx 0$ and assuming that $|u_j - u_{j+dx}| \approx \sqrt{\frac{2\mu}{\nu}}$ the highest eigenvalue can be limited by

$$\max|eigenvalue| \leq \| \mathbf{M} \|_\infty \tag{5.86}$$

and using the approximations given above gives

$$\| \mathbf{M} \|_\infty \approx \frac{1}{T} 8\mu \frac{2\nu\sigma^2}{1+2\nu\sigma^2} \tag{5.87}$$

and since the eigenvalue must be less that 1 the critical temperature is

$$1D: \, T_c = 8\mu \frac{2\nu\sigma^2}{1+2\nu\sigma^2} \tag{5.88}$$

It should be noticed that these approximations are very crude since the result is independent of the observed data. A more time consuming approximation can be to evaluate $|u_j - u_{j+dx}|$ based on the observed data still using $p_j \approx 0$. The two dimensional expansion of this approximation, without connection between the edge elements, results in extra diagonals in the matrix $\mathbf{M}$ due to indirect correlation of the edge elements through the image elements and results in

$$2D: \, T_c = 16\mu \frac{2\nu\sigma^2}{1+2\nu\sigma^2} \tag{5.89}$$

The approximation is bad if the input data is very noisy since the approximation $|u_j - u_{j+dx}| \approx \sqrt{\frac{2\mu}{\nu}}$ is not valid and $|u_j - u_{j+dx}|$ needs to be estimated from $\mathbf{d}$, which again includes an inversion of the system matrix. Using connectivity between the edge elements increases the critical temperature in a complicated manner.

5.5 Experimental Results

In this section experimental results are presented using the different algorithms: Metropolis Sampling, Gibbs Sampling and Mean Field Annealing.

The inverse of the temperature $\beta = \frac{1}{T}$ is used, which is convenient since β does not get confused with T from the sampling of the sinogram, and β increases with increasing iteration number so that the presentation of data gets simplified.

5.5.1 Comparing MAP Algorithms

First the different algorithms are tested, used as reconstruction of PET images in 2D, see section 3.3. A small version of the Shepp-Logan phantom is used, Figure (5.13), with sampling parameters: $M = N = 101$ and $\Delta x = \Delta y = 0.02$ still covering the square $[-1;1] \times [-1;1]$. The sinogram is sampled with $R = 101$, $\Delta\rho = 0.02$ and $T = 100$ (number of θ samples), Figure (5.14).

In the following map-reconstruction the parameters $\nu = 10$ and $\omega = \frac{1}{\sigma^2} = 40$, which gives a reasonable balance between the observation model and the MRF prior model; they are not optimal but chosen to illustrate the different map algorithms. The optimal parameters are depended of the given problem and can be estimated using the Generalized Boltzmann Learning Rule [Kjems, 1998]. For the Shepp-Logan phantom the lowest step is

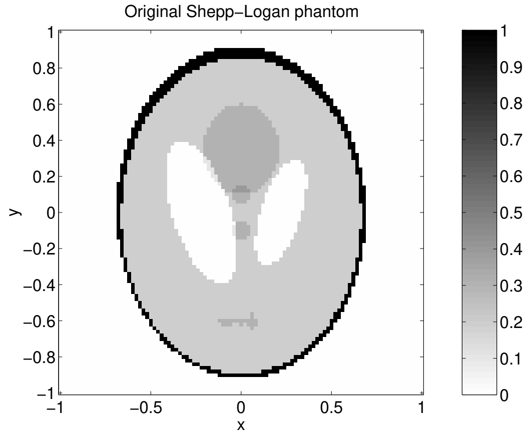

Figure 5.13: Original Shepp-Logan phantom.

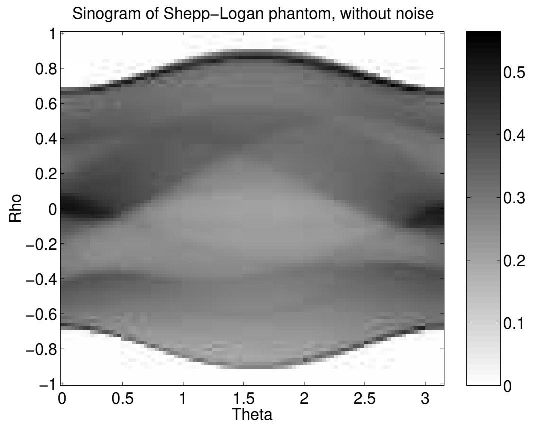

Figure 5.14: Sinogram of original Shepp-Logan phantom.

of the size $K = 0.1$ and using the model with independent edges, Eq. 5.9, gives $\mu \leq \frac{\nu}{2}K^2 = 0.05$, choosing $\mu = 0.03$ to ensure all active edges are estimated. The critical temperature (or the inverse β_c) can be estimated using Eq. 5.89 to $\beta_c = 6.25$ and the starting β must be chosen below β_c. As a very conservative guess the starting temperature is chosen to be $\beta_{start} = 0.02$.

As annealing schedule Eq. 5.72 is used with $\kappa = 0.99 \Rightarrow \kappa_\beta = 1.01$. The stop criterion for the annealing is either a maximum number of iteration or that the change in system energy, Eq. 5.42 is less that 10^{-4} per iteration.

5.5.1.1 Metropolis Sampling

An example of the result when using Metropolis sampling can be seen in Figure (5.15) with the corresponding edges in Figure (5.16) and the error measure being $L_2 = 0.33$. It can be seen that the smallest objects are not reconstructed due to the low sampling of the sinogram compared to the size of the objects. For evaluation of the annealing schedule Figure (5.18) is examined. In the first iterations high fluctuation occurs until the image elements reach an equilibrium. After the equilibrium is reached, still at high temperatures (low β), the system is in its chaotic phase where the edges are often flipped and the image elements still have large intensity changes until the critical temperature where the system stabilizes. The first small increase is at $\beta \approx 4$, but the major changes happen at $\beta \approx 100$, so the initial temperature can be lowered without loss of performance.

In Figure (5.19) the energy is shown as function of the increasing β and again it can be seen that in the first iterations the system is fluctuating but stabilized into a smooth decrease towards the minimum. It can be seen in Figure (5.20) that the L_2 norm follows the energy towards the minimum.

5.5.1.2 Gibbs Sampling

For the same example the Gibbs sampler is used ending in Figure (5.21) with $L_2 = 0.34$ and the edge configuration in Figure (5.22). To evaluate the annealing schedule Figure (5.24) is used, for low β the system is in its chaotic phase and the edges are evenly distributed between active and non-active edges. A small increase in the number of active

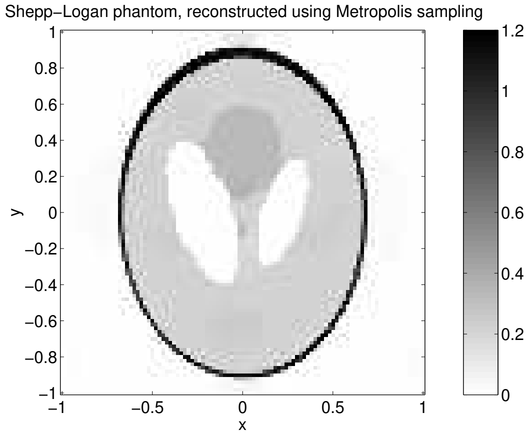

Figure 5.15: Shepp-Logan phantom reconstructed using Metropolis Sampling and $\mu = 0.03$

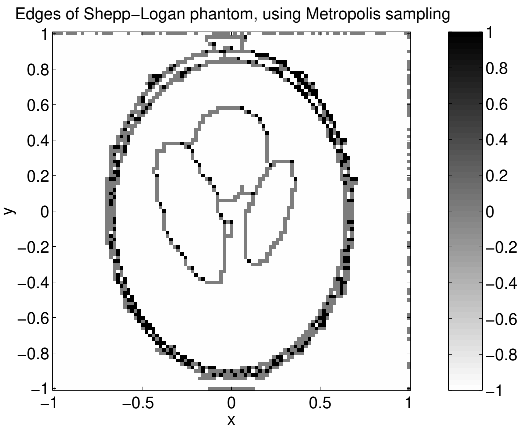

Figure 5.16: Estimated edges using Metropolis Sampling and $\mu = 0.03$

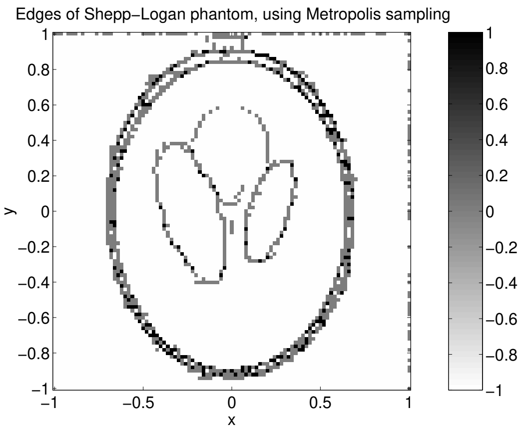

Figure 5.17: Estimated edges using Metropolis Sampling and $\mu = 0.10$

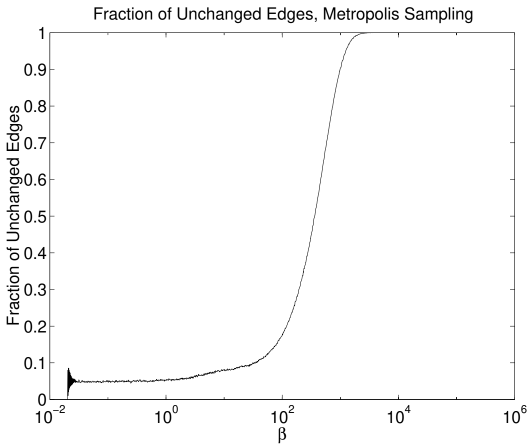

Figure 5.18: Fraction of unchanged edges as function of β in logarithmic scale for the Metropolis sampling

edges happen at $\beta \approx 4$ ending with a major decrease in the number of active edges that starts at temperature $\beta \approx 100$.

Again the L_2 error follows the energy decrease until $\beta \approx 1000$ where the L_2 error has a small increase while the energy still decreases. This is probably due to a mismatch between the MRF model and the real image or due to an incorrect observation model, but the increase in the L_2 is so small that it is negligible.

5.5.1.3 Mean Field Annealing

Using Mean Field Annealing the result gives $L_2 = 0.39$ and it can be seen in Figure (5.27) and Figure (5.28). In Figure (5.30) two critical temperatures can be estimated: the first at $\beta \approx 4$ and the second at $\beta \approx 100$. These are similar to the ones found for Metropolis sampling and Gibbs sampling.

Even though the the critical temperatures of the three algorithms are the same, the

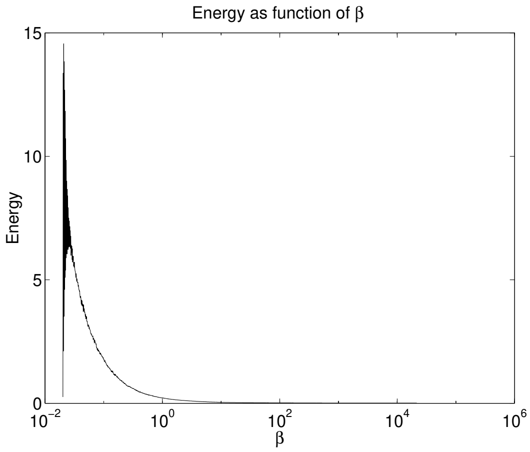

Figure 5.19: System energy as function of β in logarithmic scale for Metropolis sampling

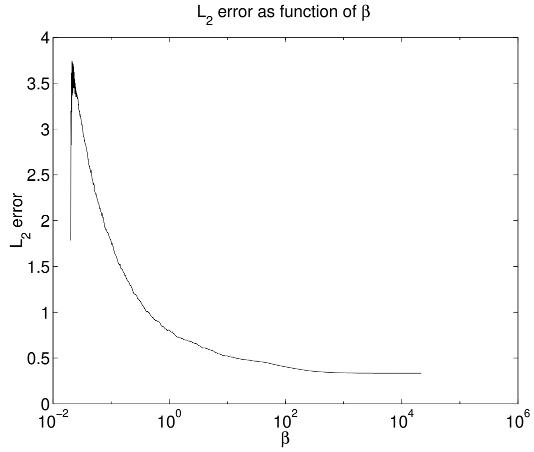

Figure 5.20: L_2 error as function of β in logarithmic scale for Metropolis sampling

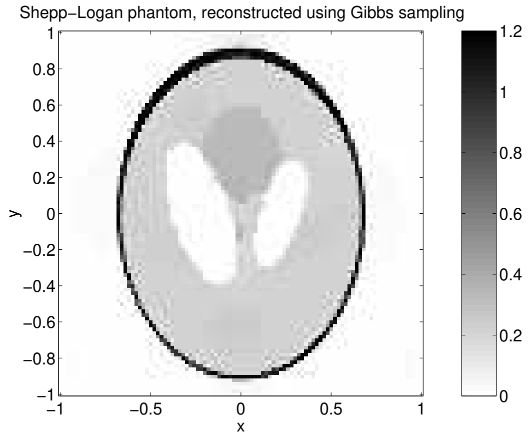

Figure 5.21: Shepp-Logan phantom reconstructed using Gibbs Sampling and $\mu = 0.03$

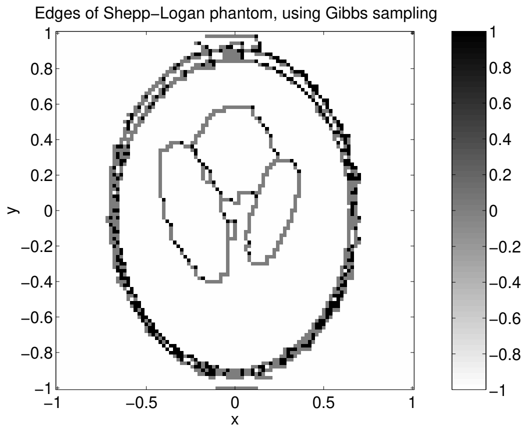

Figure 5.22: Estimated edges using Gibbs Sampling and $\mu = 0.03$

energy and the L_2 error acts very differently, Figure (5.31) and Figure (5.32). Three plateaus exist which correspond to different phases in the system: at the first plateau the system is in the chaotic phase with edge values around 0 and a very smooth image. The second plateau starts at $\beta \approx 4$ where the edges around the high intensity outer ellipsoid are activated while the rest of the image is still smoothed. After $\beta \approx 100$ the rest of the edges are activated and the image is stabilized. As for the Gibbs Sampler the L_2 error has a small increase while the energy still decreases at the last iterations, which supports the idea of a mismatch between the MRF model and the real image or is due to an incorrect observation model.

5.5.1.4 Performance Comparison

In the comparison of the performance of the different algorithms several aspects must be covered. First there is the visual impression and the norm error measure. The visual impression is that the algorithms result in images which are very alike. It seems that the Mean Field Annealing has fewer active edges and that they are closer to the true

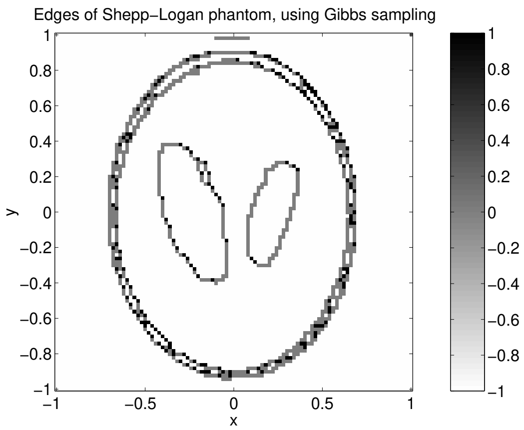

Figure 5.23: Estimated edges using Gibbs Sampling and $\mu = 0.10$

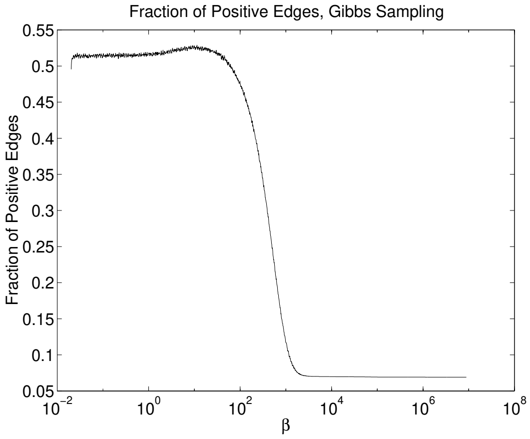

Figure 5.24: Fraction of positive edges as function of β in logarithmic scale for the Gibbs Sampler.

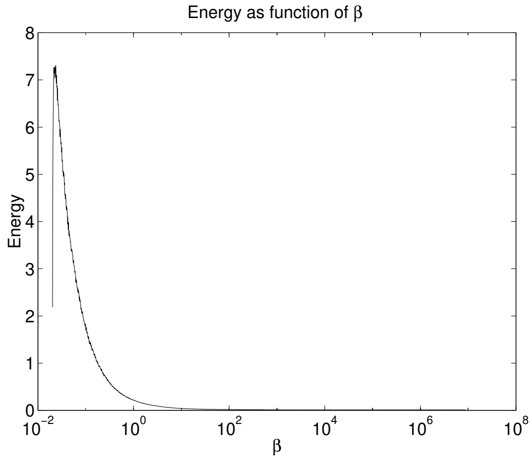

Figure 5.25: Energy as function of β in logarithmic scale for the Gibbs Sampler.

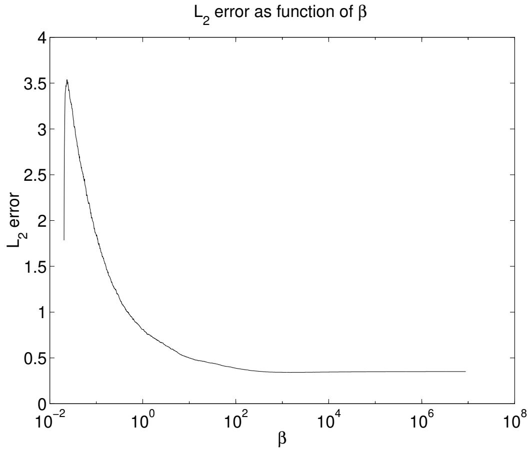

Figure 5.26: L_2 norm error as function of β in logarithmic scale for the Gibbs Sampler.

result; Figure (5.16), Figure (5.22) and Figure (5.28), but the L_2 norm gives a worse result. A closer look at the edges produced with Mean Field Annealing shows that the edges tend to be a little misplaced in that the small object with high intensity gets bigger. Another aspect is to compare the performance as function of the parameters. As a simple experiment the parameter μ is varied in the range from 0.01 to 1.0. Lower values of μ tend to make the system unstable since the limitation of creating of active edges is too small. The average performance result of 5 runs can be seen in Figure (5.33).

The Mean Field Annealing performs worse than the other two and this is again due to misplacement of the edges (see Figure (5.17), Figure (5.23) and Figure (5.29)) but the visual impression is still good. The difference between the Metropolis sampling and the Gibbs Sampler is within the errorbars of the L_2 measure. The last two aspects in the comparison are time consumption and stability. Time comparison is in the favor of the MFA algorithm for two reasons: First it is usually possible to start at lower temperature without worse performance and each iteration step is faster. Secondly normally fewer

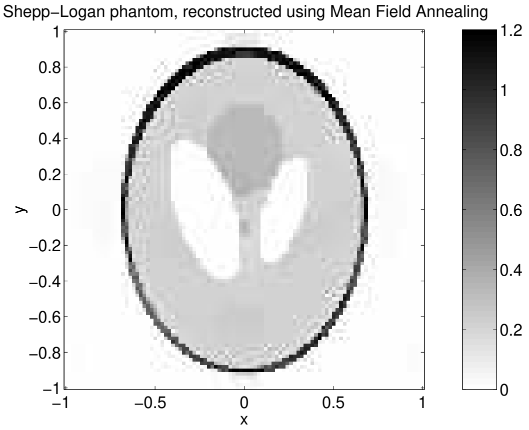

Figure 5.27: Shepp-Logan phantom reconstructed using Mean Field Annealing and $\mu = 0.03$

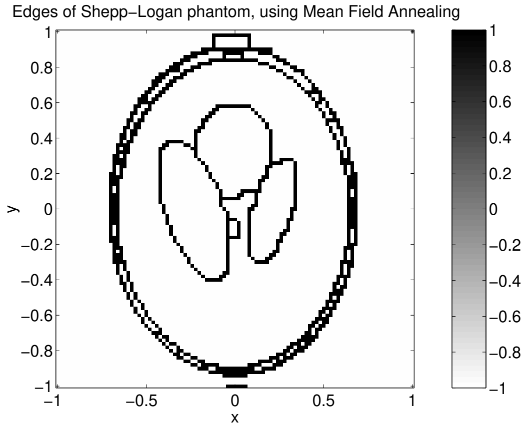

Figure 5.28: Estimated edges using Mean Field Annealing and $\mu = 0.03$

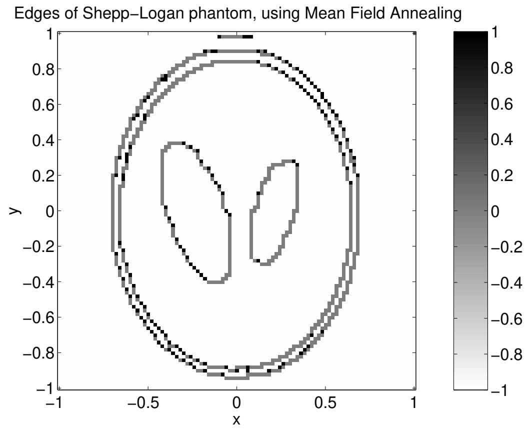

Figure 5.29: Estimated edges using Mean Field Annealing and $\mu = 0.10$

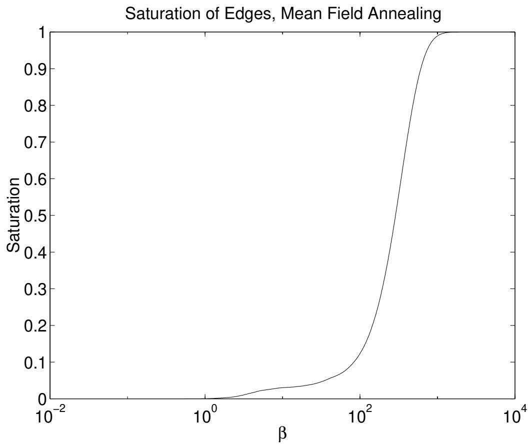

Figure 5.30: Saturation of edges as function of β in logarithmic scale.

iterations are needed. For the test results in Figure (5.33) the average run time on a PentiumII-233MHz is

Algorithm	L_2 error	time used (relative)
Metropolis Sampling	0.33	130 sec.
Gibbs Sampling	0.34	194 sec.
Mean Field Annealing	0.39	69 sec.

All these results include calculation of performance results, saving of temporary results and checking of divergence of the algorithms. If all these functions are minimized the time consumption is approximately $\frac{1}{3}$.

Regarding stability the Metropolis Algorithm results are the most varying. The algorithms are then tested with Gaussian white noise added to the sinogram, Figure (5.34). The image is now reconstructed with $\mu = 0.10$ and $\omega = 20$ and the corresponding results

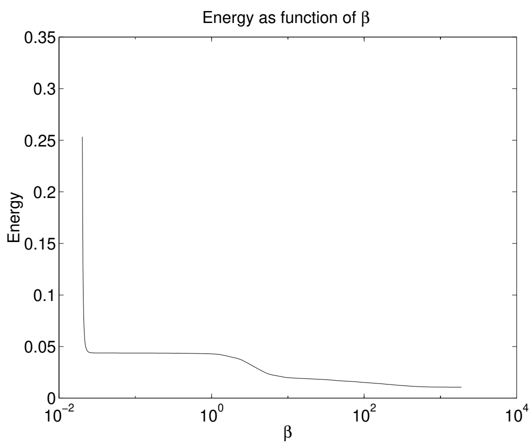

Figure 5.31: Energy as function of β in logarithmic scale for The Mean Field Annealing.

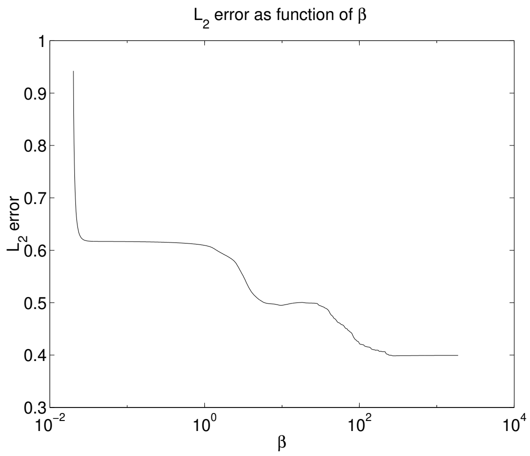

Figure 5.32: L_2 norm error as function of β in logarithmic scale for the Mean Field Annealing

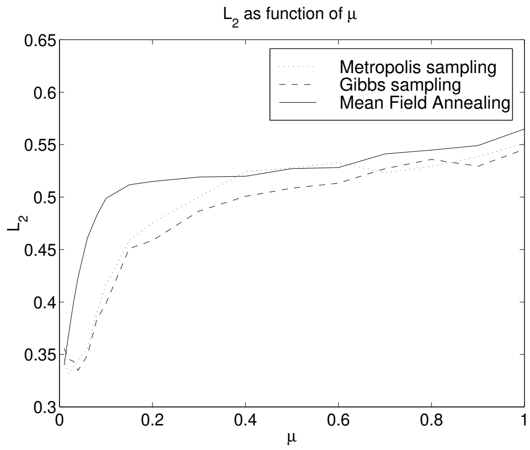

Figure 5.33: L_2 error as function of μ, average of 5 runs.

can be seen in Figures (5.35) to (5.40).

To obtain better performance for the Metropolis algorithm and for the Gibbs Sampler the annealing schedule is changed to obtain a higher number of iterations at the phase transitions; this is not needed for the Mean Field Annealing. The L_2 error and the time used for the shown images are

Algorithm	L_2 error	time used (relative)
Metropolis Sampling	0.75	2.9
Gibbs Sampling	0.85	4.2
Mean Field Annealing	0.57	1.0

Again the Mean Field Algorithm tends to have more smoothed images, but for noisy data the final result is better. By adjusting the parameters and the annealing schedule for the Metropolis algorithm and the Gibbs Sampler, they can perform just as well as the

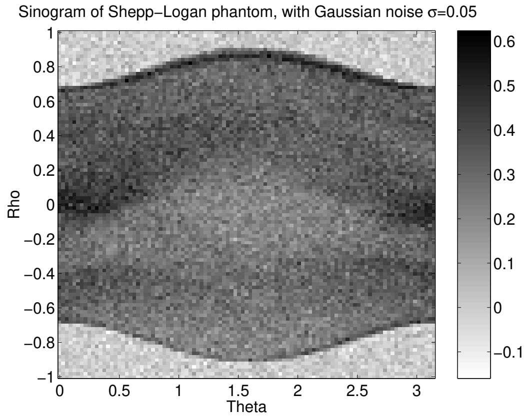

Figure 5.34: Sinogram of original Shepp-Logan phantom with added Gaussian noise.

Mean Field Annealing but with higher computational cost and therefore the Mean Field Annealing is used in section 3.3.

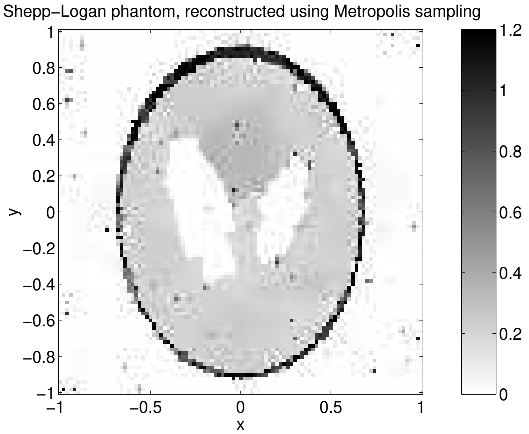

Figure 5.35: Noisy Shepp-Logan phantom reconstructed using Metropolis Sampling and $\mu = 0.10$

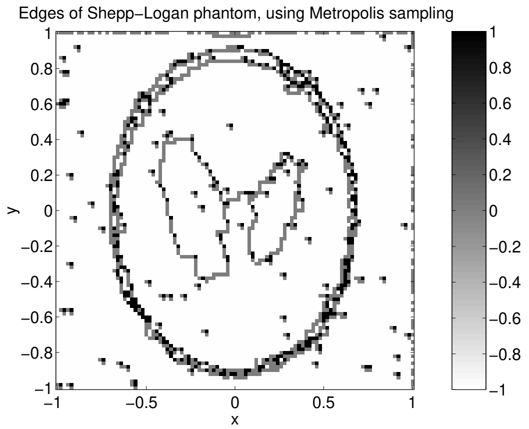

Figure 5.36: Estimated edges using Metropolis Sampling and $\mu = 0.10$

5.5.2 Stability of the Noise Parameter

In this subsection the mean field algorithm is tested for parameter stability, given different noise properties of the input data to match the reconstruction of PET images; Using $\omega = \frac{1}{\sigma}$ as the estimated noise parameter in the observation model, which is varied to find the optimal value for a given input sinogram. The input sinograms are generated using Poisson processes to match the physical properties of PET using different count rates as in Section 3.3.2. The results , expressed as the L_2 error norm, are presented in Figure (5.41) as function of ω and the total number of counts in the sinogram.

The area with $L_2 = 0$ (white) corresponds to an unstable system and no result is produced. The unstable area is characterized by high noise in the input data (few counts) and high reliance on the input data (high ω) which cannot be combined. For low values of

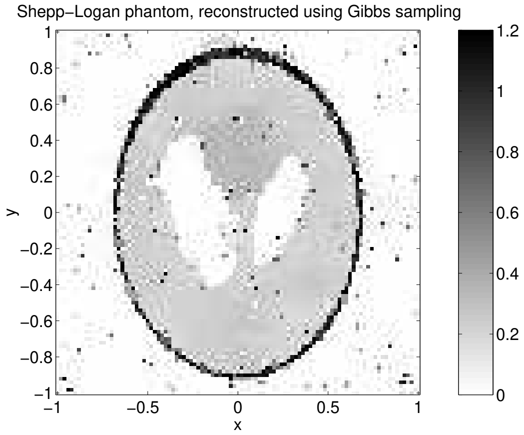

Figure 5.37: Noisy Shepp-Logan phantom reconstructed using Gibbs Sampling and $\mu = 0.10$

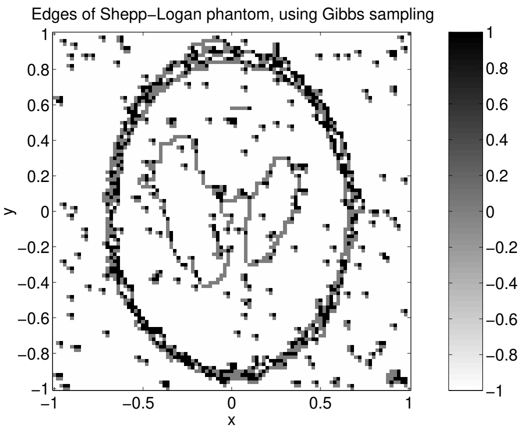

Figure 5.38: Estimated edges using Gibbs Sampling and $\mu = 0.10$

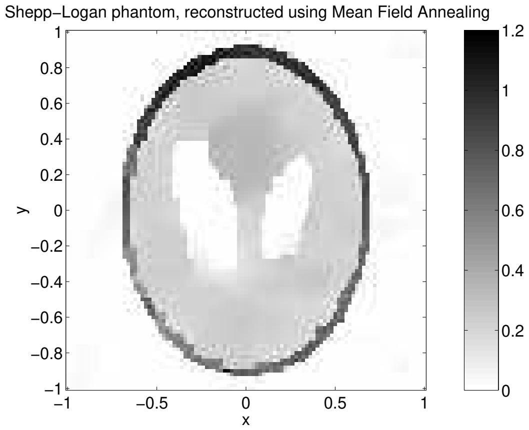

Figure 5.39: Noisy Shepp-Logan phantom reconstructed using Mean Field Annealing and $\mu = 0.10$

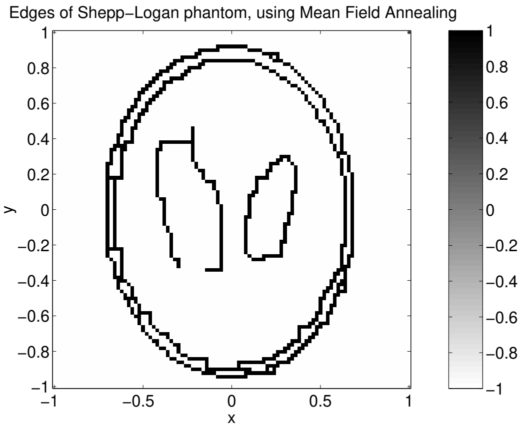

Figure 5.40: Estimated edges using Mean Field Annealing and $\mu = 0.10$

ω the outcome of the MFA algorithm is always far from the true image and very dominated by the smoothing in the MRF model and therefore the result is a smoothed version of the desired image. The optimal value of ω follows a square-root function of total number of counts, but the final result is not sensible to a small mismatch ($\pm 10\%$) of ω which does not influence the result.

5.5.3 Prior on Edges – Fixed Edges

The advantage of applying additional information about the actual edge configuration can be seen in Figure (5.42) where the L_2 error is at least halved. The results are comparable to the results in Section 3.3.2 and show that adding additional information in the reconstruction process of PET images increases the performance. This is also the background of the work in [Gindi et al., 1995, Ouyang et al., 1994].

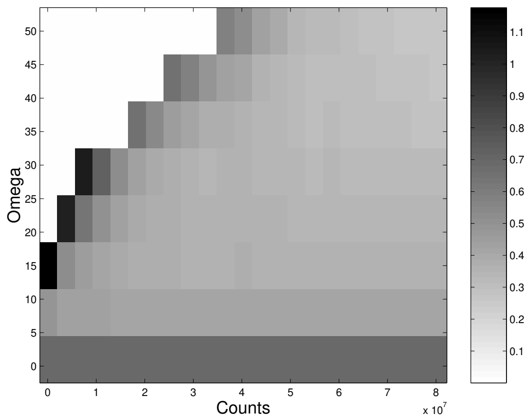

Figure 5.41: L_2 error as function of ω and counts in the sinogram. The white area corresponds to an unstable system and no result is produced.

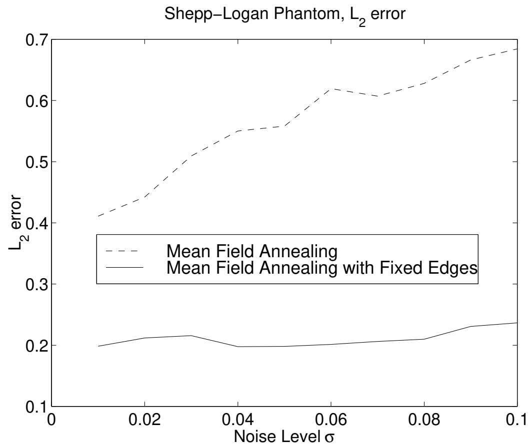

Figure 5.42: L_2 error as function of added Gaussian noise in the sinogram, σ

5.5.4 Restoration of Blurred Images

In this subsection a small example of deblurring is shown; results on PET images can be seen in section 6.2. The observation model is with a blurring kernel added with Gaussian noise.

The example is a $6 \times 6 \times 6$ cube with intensity value of 100 on a $16 \times 16 \times 16$ background, which is blurred with a Gaussian kernel and Gaussian noise is added. The blurring kernel is symmetric with standard deviation 1.5 and the standard deviation of the noise is 5. A slice of the observed signal is in Figure (5.43), and the corresponding restored slice is in Figure (5.44). The edges are fixed on the true edges. Profiles through the center of the cube can be seen in Figure (5.45).

The model is tested with varying size of the blurring kernel and for an increasing amount of noise. The resulting L_2 errors can be seen in Figure (5.46). As expected the algorithm performs well at low noise ratios and with small blurring kernel size. At high noise ratios and large blurring the estimation of the edges gets difficult and the results are

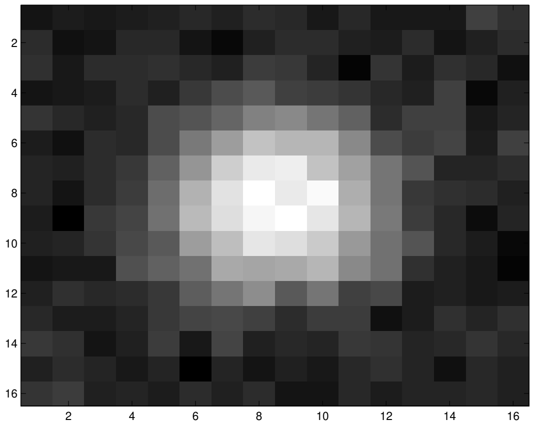

Figure 5.43: Slice of observed cube, blurred and noise are added.

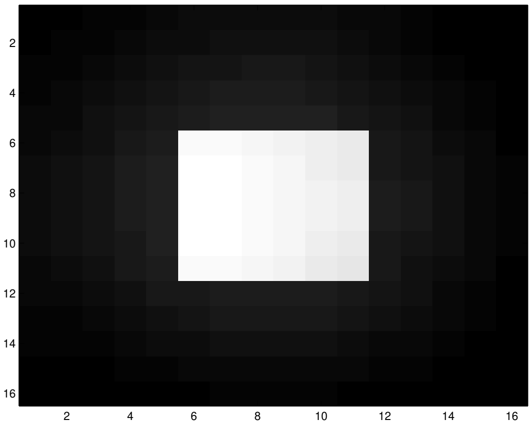

Figure 5.44: Slice of the restored image.

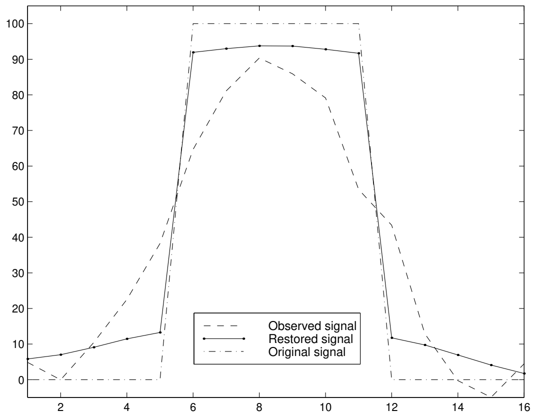

Figure 5.45: Profiles of the test example through the center of the observed and restored image.

poor. The number of iterations used can be seen in Figure (5.47) and illustrates the need for more iteration when the input data are highly degraded.

5.5.5 Summary

In this section some of the properties of the annealing MAP algorithms are exploited. An estimate of the highest critical temperature is given and the phases of the system model are examined. By an example it is shown that the critical temperatures of the system, system phase changes, are independent of the algorithm used to find the map estimate.

The system stability regarding the parameter choice is explored. The effect of parameter choice on the performance is investigated.

The advances of external edge information are illustrated.

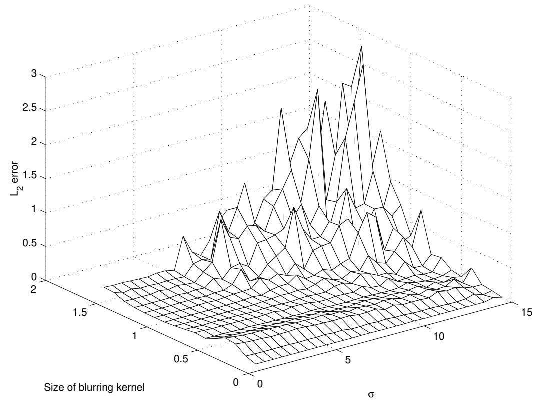

Figure 5.46: L_2 error as function of σ and the size of the blurring kernel

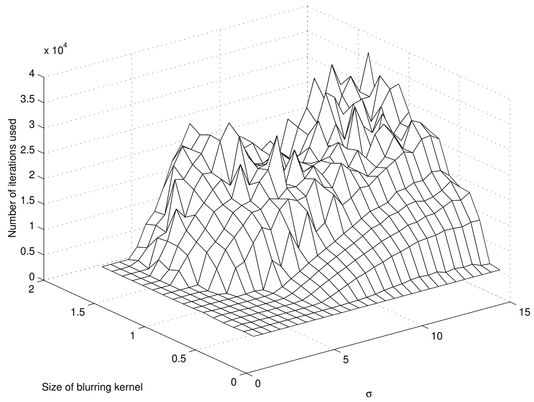

Figure 5.47: Number of iterations used for Mean Field Annealing as function of σ and the size of the blurring kernel

Chapter 6

Restoring Functional PET Images Using Anatomical MR Images

In this chapter a few methods to enhance edges in PET images are investigated with the purpose of obtaining higher resolution in the PET images. The first method extracts information about closed contours in the image using the image itself or using external information from MR images. In the other section a method to extract information from MR images is discussed. The direct method of using standard edge detection methods in the MR image does not give convenient results.

All of the results using both PET and MR images are very dependent on the alignment of the PET and MR image since a bad registration can introduce new errors in the restoration. In this section the alignment is done using the program package AIR, [Woods and Cherry, 1993], which result in a 6 parameter transformation determining translation and rotation.

6.1 Active Contour - Use of Snakes

In this section is shown how a "Snake" contour model may be used to produce efficient edge hints to the MRF model in section 5.2. There has been progress in the use of deformable models for edge and surface identification, see, e.g., [Cohen and Cohen, 1993, Storvik, 1994, Yezzi et al., 1997, Zhu and Yuille, 1996] both in 2D and in 3D. In this section strong edge hints for the Markov Random Field model through a Snake deformable model is presented. This is only in two dimensions but extension to three dimensions is possible where the active surface is called balloons. The Snake is defined to be a periodic set of N points in the visual field of an image: $\mathbf{r}_j = (x_j, y_j)$. Periodicity meaning $\mathbf{r}_{j+pN} = \mathbf{r}_j, p \in Z$. The Snake energy function consists of a form control part E_{form} and a match part E_{match}.

$$E = E_{\text{form}} + E_{\text{match}} \tag{6.1}$$

In a simple version the form control preserves total length,

$$E_{\text{form}} = \frac{a}{2}(d - d_0)^2 \tag{6.2}$$

$$d = \sum_j |\mathbf{r}_j - \mathbf{r}_{j+1}| \tag{6.3}$$

where d_0 is the initial Snake length and a is a parameter to control the balance between the two parts in the energy function. More complex form controls can be designed to preserve shape, corners, etc [Cootes et al., 1994, Storvik, 1994, Gunn and Nixon, 1997]. The Snake match energy is designed to ensure that the Snake points track edge contours in the image field. This effect may be obtained by letting the Snake points seek local maxima in the gradient energy map of the image, $G(\mathbf{r})$, evaluated from the observed image. The total Snake match energy is then given by,

$$E_{\text{match}} = -\sum_j G(\mathbf{r}_j). \tag{6.4}$$

Snake dynamics is established through gradient descent,

$$\frac{1}{\eta}\frac{\partial \mathbf{r}_j}{\partial t} = -\frac{\partial E}{\partial \mathbf{r}_j} = a(d-d_0)\frac{\partial d}{\partial \mathbf{r}_j} + \frac{\partial G(\mathbf{r}_j)}{\partial \mathbf{r}_j} \tag{6.5}$$

where η is a parameter to control the step size in the gradient direction. In numerous image processing applications quite detailed prior information can be devised. In e.g. brain scan reconstruction, detailed atlases are known describing the generic brain topography under various scanning modes. To illustrate the use of strong edge hints in brain scans, the head phantom Ph2 is used, shown again in Figure (6.1). The reconstruction is done from a noise corrupted image shown in Figure (6.2) and the inversion of the Radon transform is omitted. During the reconstruction process the edges are estimated, due to the interpretation of the image.

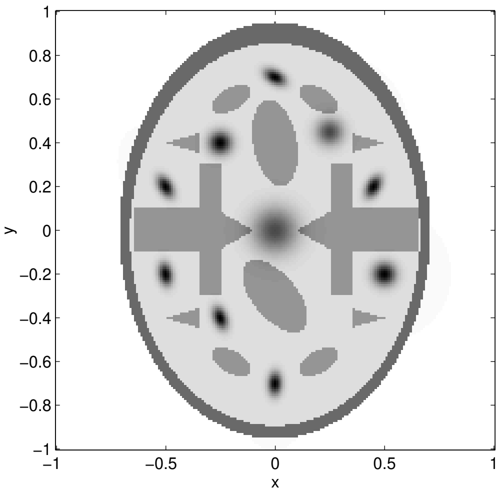

Figure 6.1: Original head phantom without noise.

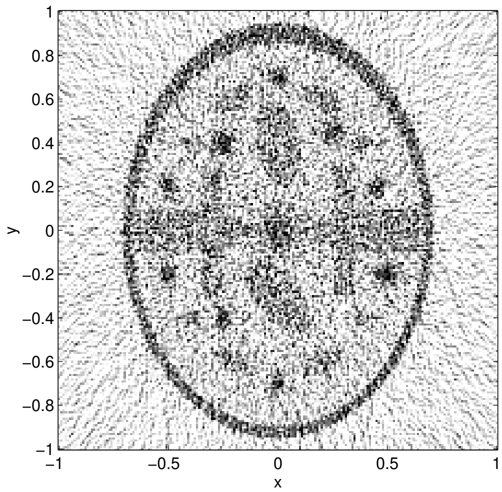

Figure 6.2: Noise corrupted head phantom shown in same color scale.

Snakes are initialized in seven generic positions, as shown in Figure (6.3). For this particular instance the Snake equilibrates in about 100 iterations as illustrated in Figure (6.4). The approach to equilibrium is quite sensitive to the topography of the image and careful control of η is necessary, Eq. 6.5. This problem may be relieved by use of a pseudo-second order search direction instead of the gradient of the edge energy map.

Subsequently, hints are created by modulating the chemical potentials μ^p, μ^q and λ. This modulation implies that edge units are strongly suppressed outside of the region

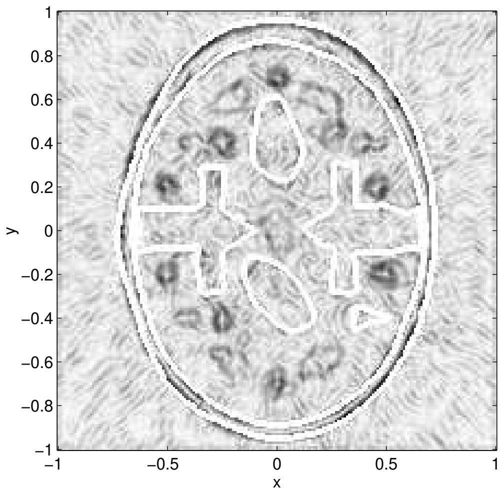

Figure 6.3: Edge energy map with initial snake positions shown in white.

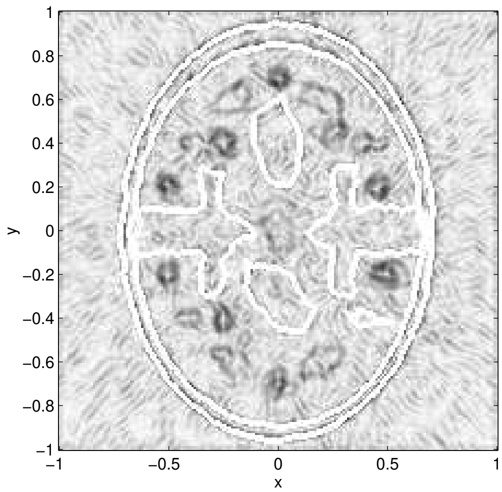

Figure 6.4: Edge energy map with equilibrated Snake positions shown in white.

suggested by the Snakes. The results of the Mean Field Annealing are presented in Figure (6.5) (without Snakes) and Figure (6.6) (using Snakes).

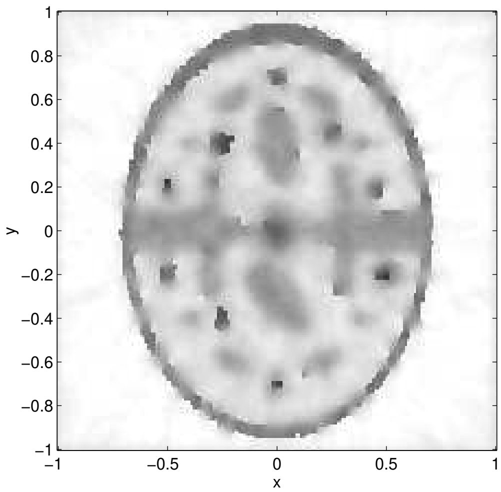

Figure 6.5: The restored head phantom without using Snakes edge priors to the Mean Field Annealing.

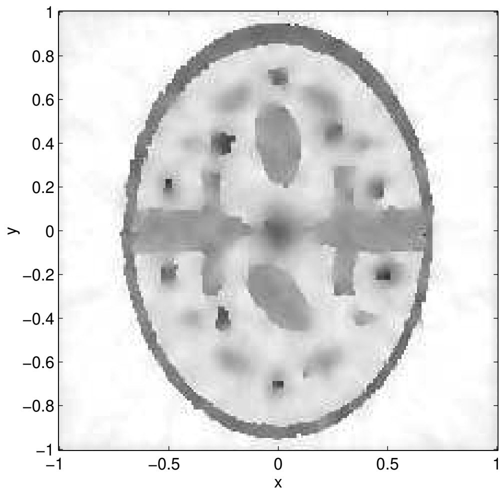

Figure 6.6: The restored head phantom using Snakes as edge priors to the Mean Field Annealing.

Note that the intensity levels inside the regions of the Snakes have been estimated more closely using Snakes. This might be of significant importance in, e.g., brain activation studies, in which the minute differences in brain activity between activated and resting states are investigated. Using Snakes and Mean Field Annealing enhances the Signal to Noise ratio (SNR) with 12.5 dB. Without Snakes the improvement is 11.3 dB. The improvement is due to a better estimation of edges.

In Figures (6.7) and (6.8) are shown the active edges (i.e., $p = +1$ or $q = +1$) without using Snakes and with Snakes respectively. As seen from Figure (6.8) it is possible to

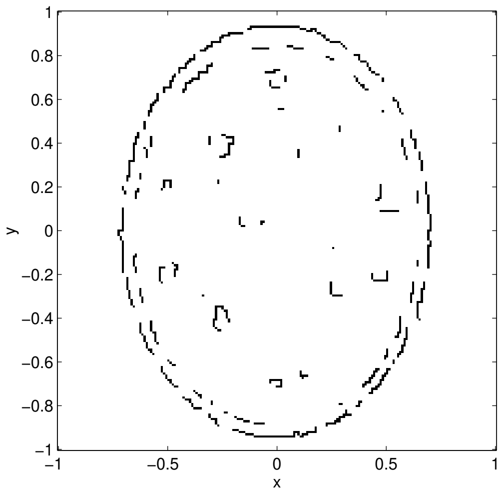

Figure 6.7: The estimated edges without using Snakes. Active edges are marked black.

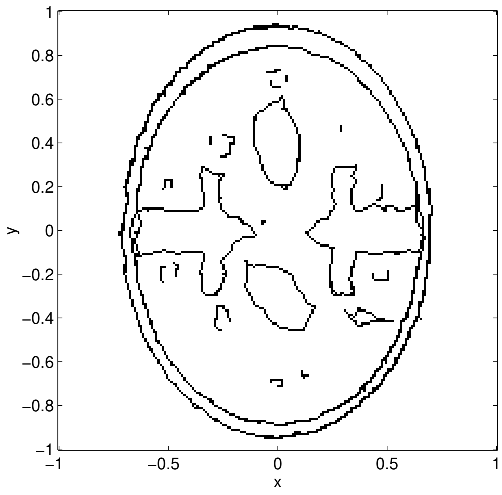

Figure 6.8: Estimated edges using Snakes as priors. Active edges are marked black.

incorporate strong priors in order to obtain closed edge contours. This can also be used to segment the image.

Another way of evaluating the results is to examine the histograms of the images. Figure (6.9) shows the histogram of the original noise free image and Figure (6.10) shows, in the same interval, the histogram of the noise corrupted image.

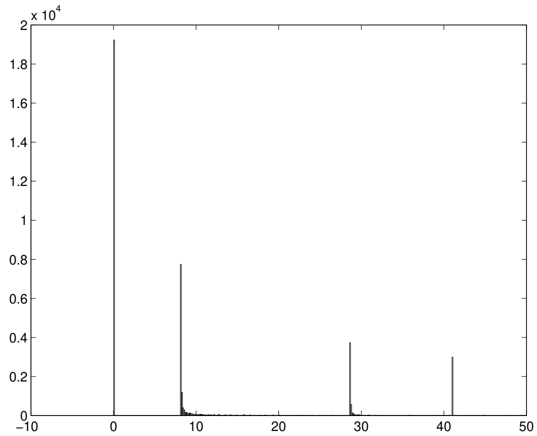

Figure 6.9: Histogram of original noise free head phantom.

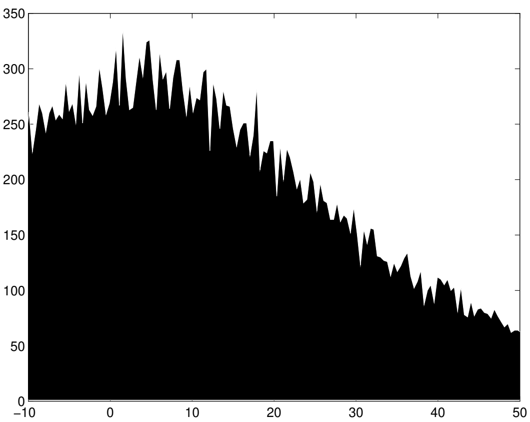

Figure 6.10: Histogram of the observed noise corrupted head phantom.

Without using Snakes the histogram after restoration can be seen in Figure (6.11). Finally Figure (6.12) shows the histogram corresponding to the restored image using Snakes as a prior. From the Figures it can be seen that the two higher levels (29 and 41) is better resolved using Snakes and compared to the noisy histogram shown in Figure (6.10) the result is far better.

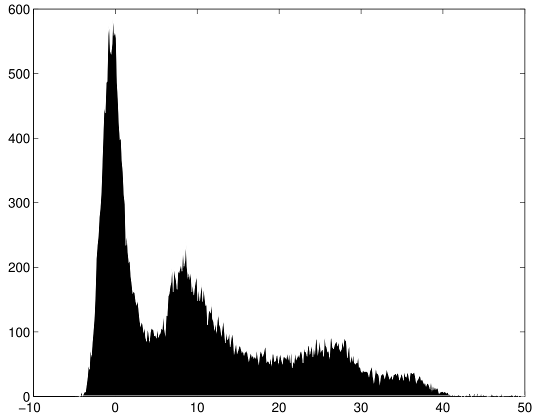

Figure 6.11: Histogram of the restored image without using Snakes.

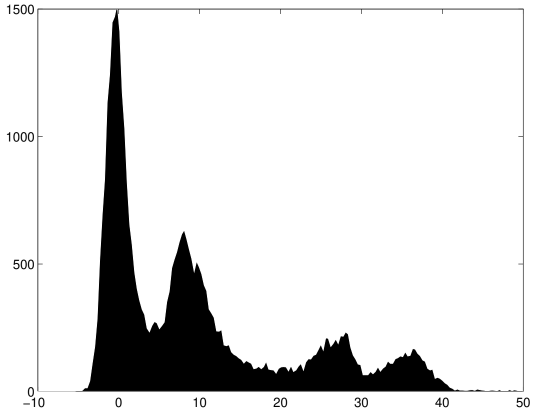

Figure 6.12: Histogram of the restored image using Snakes as priors.

6.1.1 Snakes Used on MR Images

In this section Snakes are used to find the edges between the gray and white matter and around the ventricles in the MR image. The start guesses are placed by hand, Figure (6.13), and after 500 iterations the placements in Figure (6.14) are found. This placement is not perfect and some errors will occur. The original PET image can be seen in Figure (6.15) and the edge enhanced image in Figure (6.16). The use of edge information can eliminate some of the smoothing, especially at the border of the image and by the ventricles. Profiles through a line of the images can be seen in Figure (6.17).

Figure 6.13: Initial Snake placement in a smoothed gradient of the MR image, shown in log scale.

Figure 6.14: Ending Snake placement in a smoothed gradient of the MR image, shown in log scale.

This is not convenient since the Snakes need manual placement for each snake, and the precision of the initial guess in some areas needs to be good. Instead of using snakes on each slide balloons could be used [Hill et al., 1993, Ohlsson et al., 1997] but still the

Figure 6.15: Original smoothed PET image.

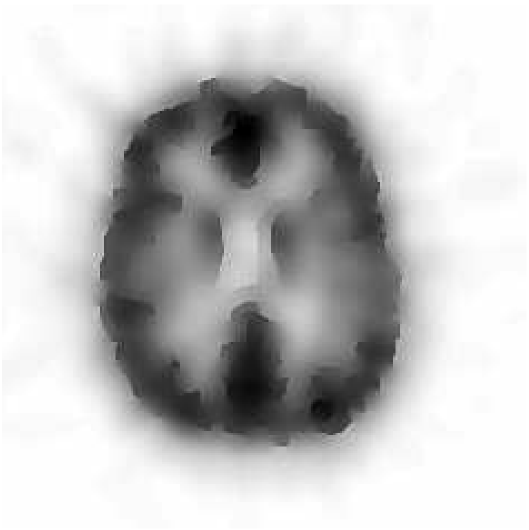

Figure 6.16: Edge enhanced PET image.

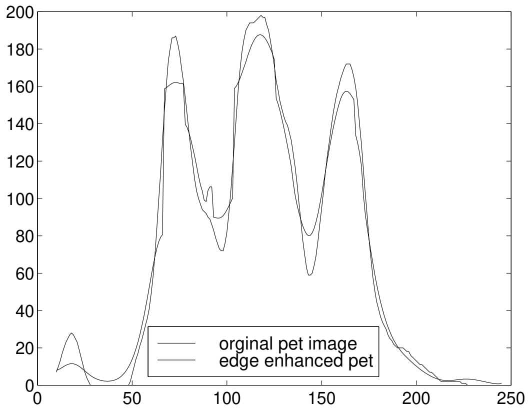

Figure 6.17: Profiles of original smoothed PET image and edge enhanced PET image.

initial guess of the placement needs to be precise.

6.2 Using Edges from Segmented MR Images

In this section edges generated from segmented MR images are used. The edges correspond to the anatomical boundaries in the MR image between the different tissue types. To obtain the desired edge information from the raw MR images shown in Figure (6.18), several steps are needed. Focusing on edges between different tissue types is based on the knowledge that tissue types in the brain have different activation levels. To obtain a tissue segmented brain several steps are necessary.

6.2.1 Pre-Processing MR Images

First a segmentation in brain/non-brain is done as shown in Figure (6.19). This is needed to obtain a better alignment between PET and MR and to improve segmentation/classifi-

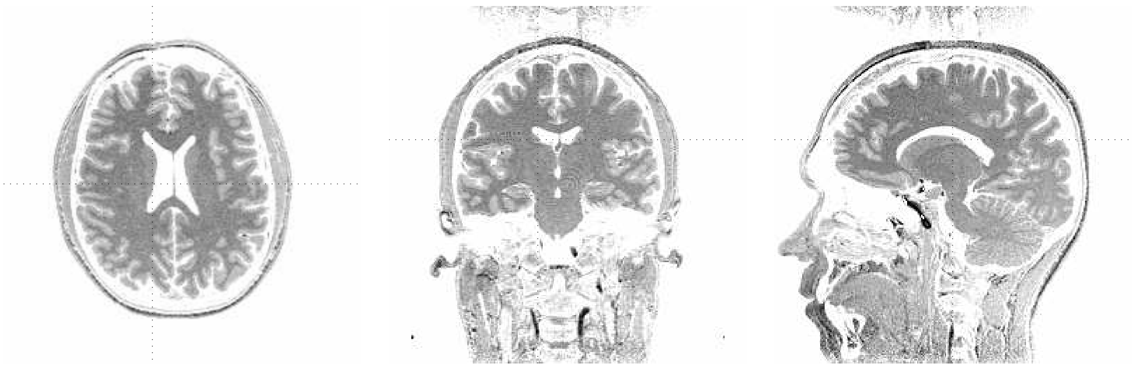

Figure 6.18: Slices of the Raw MR image: Axial, Coronal and Sagittal.

cation of the brain. The program used is developed during my stay at Brain Image Analysis Laboratory (BIAL), University of California, San Diego (UCSD), and a short description can be found in appendix A. Alignment is done using AIR, [Woods and Cherry, 1993], which is a 6 parameter transformation determining translation and rotation.

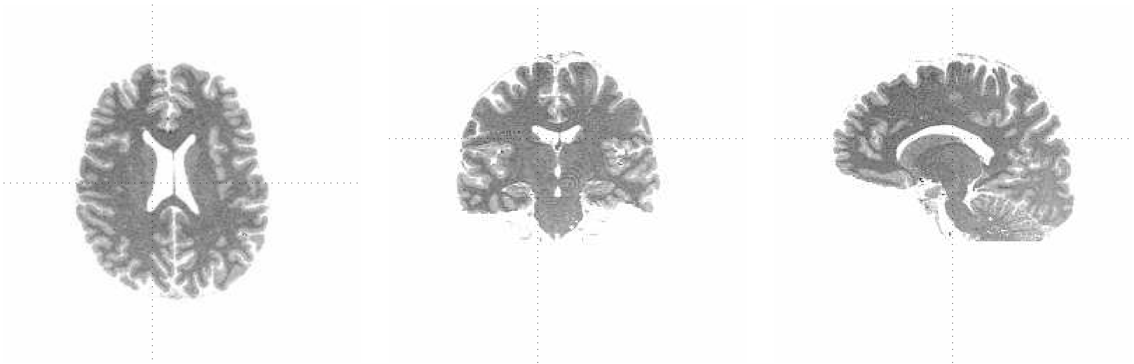

Figure 6.19: Only brain voxels of the MR image: Axial, Coronal and Sagittal slices.

Next step is a segmentation of the brain to obtain volumes of the different tissue types. In this example we segment into CSF, white matter, cortex gray matter and subcortical gray matter, Figure (6.20), using a Gaussian classification. To improve classification gray matter is separated into 2 types, but in the following these are merged again, because we have no information about any difference in PET activation in these 2 types of gray matter.

From the segmented MR image a set of edge volumes $(\mathbf{p}, \mathbf{q}, \mathbf{r})$ is created based only on gray matter and white matter, Figure (6.21).

6.2.2 Restoring PET image.

In Figure (6.22) slices of the resampled PET image are shown. The PET image is smoothed during the alignment and the resampling to MR resolution.

Using an observation model with a Gaussian blurring kernel set to 3 mm and with a noise level at 2 the result can be seen in Figure (6.23). The result came from running 100

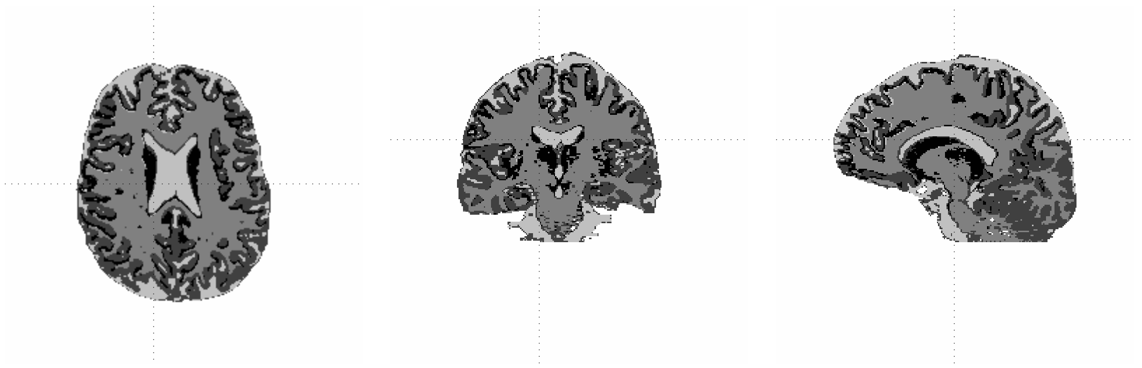

Figure 6.20: Slices of the segmented MR image, the two types of gray matter are the two darkest, then white matter and CSF.

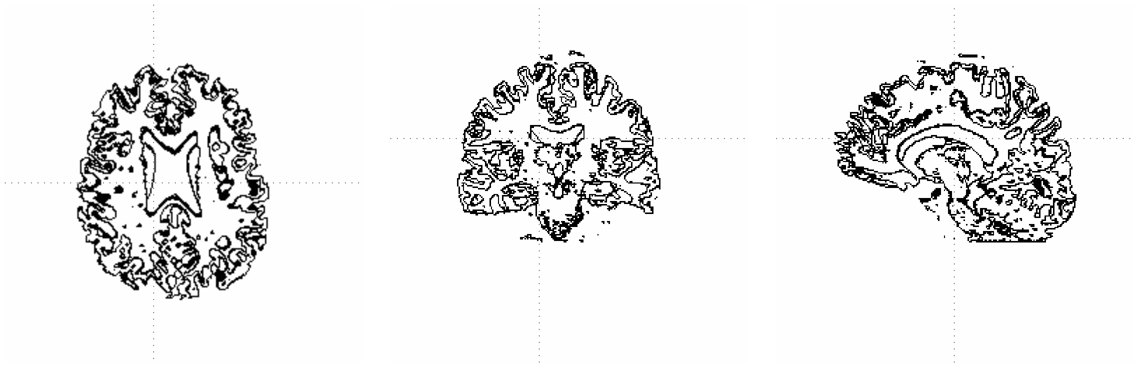

Figure 6.21: Slice of edges created from MR image, ignoring CSF and merging the two types of gray matter.

iterations with an image size of $190 \times 215 \times 176$ voxels using 220 Mb ram on a Silicon Graphics ONYX and less than 1 cpu hour. It was possible to preserve/enhance edges and avoid edge blurring on the outside of the brain.

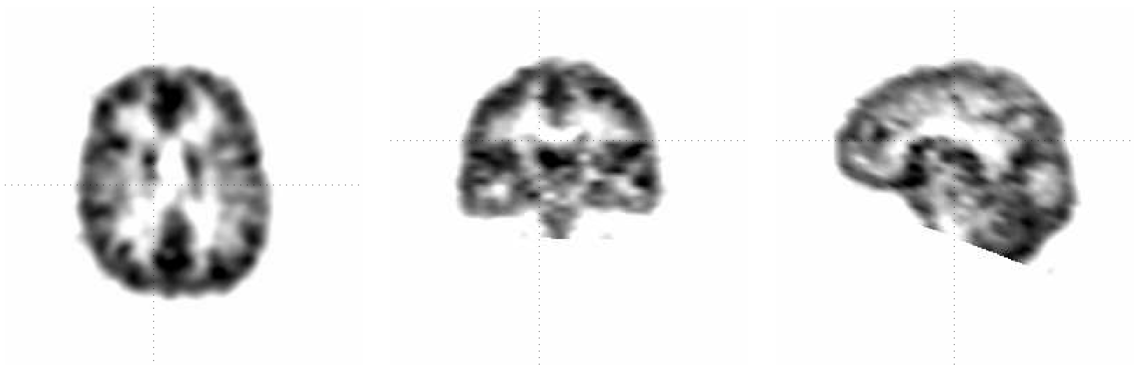

Figure 6.22: Slice of the original PET image, smooth due to resampling to MR resolution.

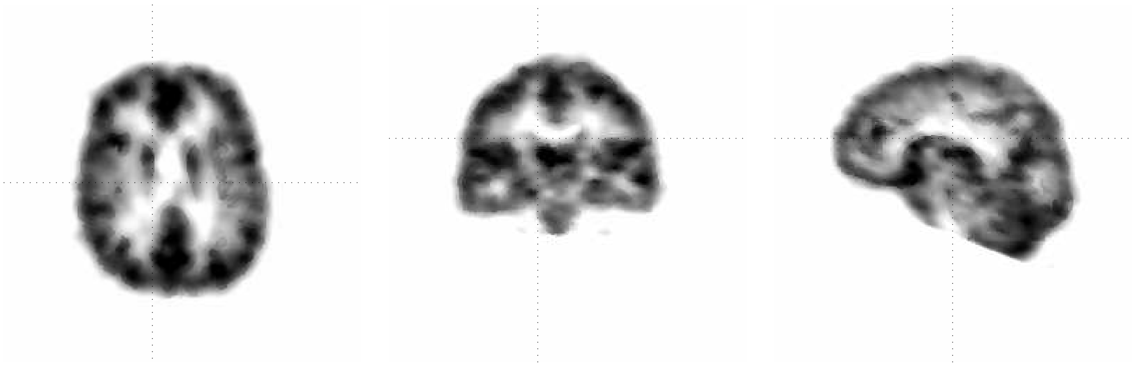

Figure 6.23: Slices of Restored PET image with a Gauss blurring on 3 mm and a noise level on 2.

6.3 Summary

In this chapter restoration of PET images using Snakes are examined both for extracting extra information from the PET image it self and from MR images. Edges generated by segmented MR images are used in restoration of a 3D PET image, as postprocessing of the reconstruction. The edges correspond to anatomical boundaries between the different tissue types.

Conclusion

In this thesis the mathematical background of PET scanning is outlined. Several standard two-dimensional reconstruction methods are explored including direct and iterative methods as well as a Bayesian reconstruction technique based on a Markov Random Field model. The different methods are tested with different sampling parameters and with different noise models and varying signal to noise ratios. The methods are used in reconstruction of real brain data. It is found that the MRF-MFA perform just as well as the ML-EM for both Poisson and Gaussian noise.

The basics of 3D reconstruction is described and the derivation of some of the standard inversion methods are presented. An initial examination of the methods is carried out.

An introduction to Markov Random Fields with discontinuities (edge elements) is given and some simple rules to control the behavior of the edges are derived. The system stability with respect to the parameter selection is analyzed. The effect of parameter choice on the performance is investigated.

Some of the properties of annealing MAP algorithms: Metropolis Sampling, the Gibbs Sampler and Mean Field Annealing, are exploited.

An estimate of the highest critical temperature is given and the phases of the system model are examined. By example it is shown that the critical temperatures of the system, i.e. the system phase changes, are independent of the algorithm used to find the map estimate.

It is furthermore shown that additional knowledge about edge information can improve the reconstruction of PET images, and a few methods to extract additional edge information are investigated.

In the context of restoration of PET images Snakes are examined for extracting extra information both from the PET image itself and from MR images. Edges generated by segmented MR images are used in restoration of a 3D PET image as postprocessing of the reconstruction. The edges correspond to anatomical boundaries between the different tissue types.

It is shown that incorporation of additional information in the reconstruction increase the performance. However, there are many aspects of PET-MR "Sensor Fusion", such as PET-MR alignment, that need further investigation.

Appendix A

Separation of Brain/Non-Brain in MR Images

This appendix contains a short technical description of the program I developed during my stay at Brain Image Analysis Laboratory (BIAL), University of California, San Diego (UCSD). The program creates a mask indicating the brain, using 2 3D MR scans. The method can also be used with only one MR image, but with poorer result.

A.1 Introduction

The goal of this program is not to make a perfect automated segmentation in brain/non-brain, but to make a program that can compete with manual methods in time and liability. Therefore user interaction is possible/needed during the different steps. In almost perfect scans no or only little user interaction is needed but always as verification. It is implemented in idl (v5.0) but this appendix does not have any details about the implementation.

In it basic form it uses 2 3D MR scan, a pd and t2 weighted, but other scan types can be used. The best results are obtained using scan types where the different tissue types are distinguishable.

A.2 Basic Relations.

The segmentations are constructed in several steps with each step as a simple operation. The steps are constructed to be continuous, but each can be repeated/redone if necessary without use of much extra time.

A slice of the original pd image can be seen in Figure (A.2).

The first step is a double threshold in both scans. The user determines a low and a high threshold for each scan, based on the one dimensional histograms of the scan and the two dimensional histogram (also called 'scatter plot'). The program estimates possible values based on the steepest slope and the steepest descent in the pd scan figure (A.3). The threshold values are symbolized with $pd_{min}, pd_{max}, t2_{min}$ and $t2_{max}$. The mask defines the area of the brain as 1 and background and tissue of no interest as 0.

$$mask_{i,j,k} = (pd_{min} \leq pd_{i,j,k} \leq pd_{max}) \text{ and } (t2_{min} \leq t2_{i,j,k} \leq t2_{max})$$

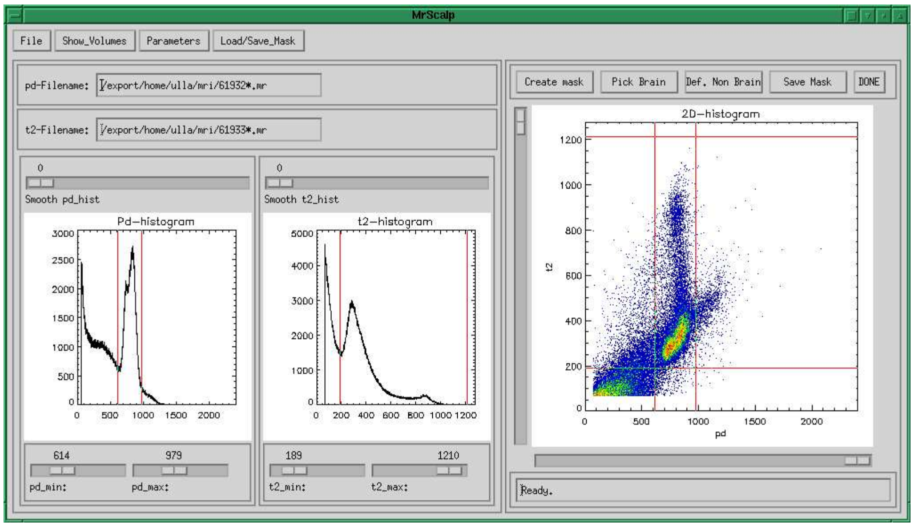

Figure A.1: Mrscalp main window.

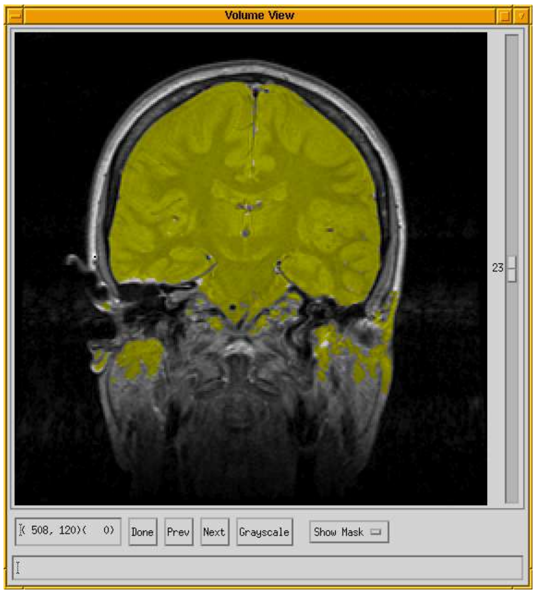

Figure A.2: Coronal slice through center of the brain.

The result of the first step can be a mask defining not only the brain, but also other areas outside the brain. These can be connected in 3d through a nerve from the brain to e.g. the eyeballs, figure (A.6).

This is not desired and another step is applied to remove these connections. The second step uses the morphological operation called 'opening'. The operator is added to remove small single voxels and voxels made of thin lines etc. It is implemented using the morphology operations: dilate and erode. A discussion of this topic is beyond the scope of this report. A suggested reference is: Haralick, Sternberg, and Zhuang, "Image Analysis Using Mathematical Morphology," IEEE Transactions on Pattern Analysis and Machine

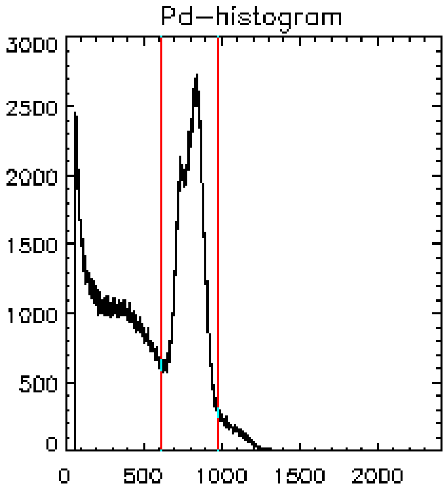

Figure A.3: Histogram of pd scan.

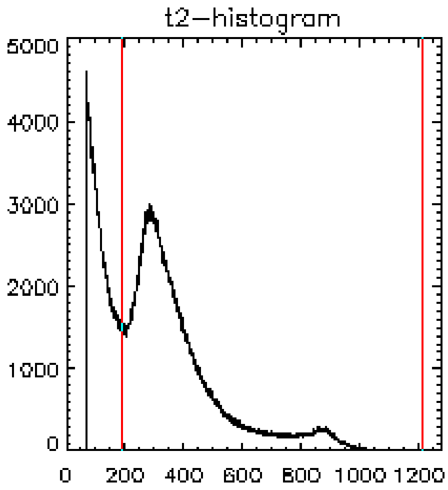

Figure A.4: Histogram of t2 scan.

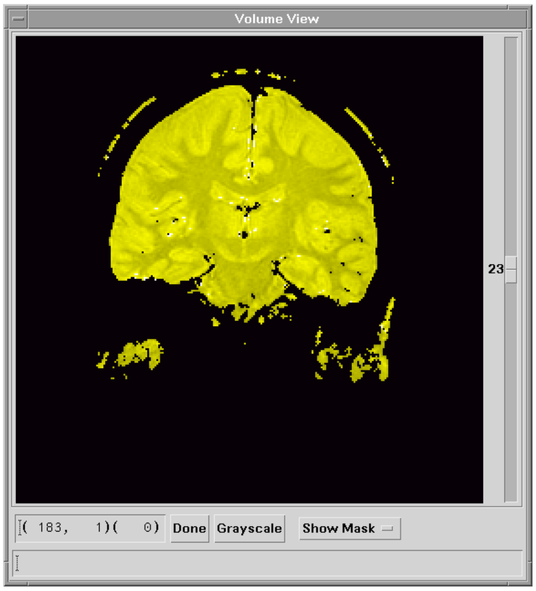

Figure A.5: Coronal Slice through center of the brain after thresholds are applied.

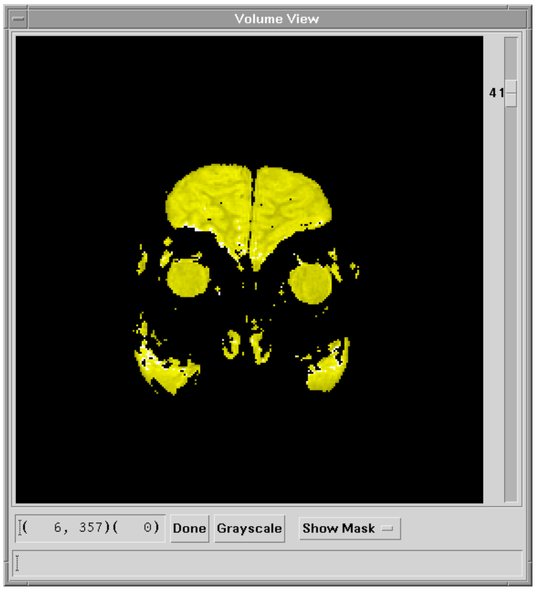

Figure A.6: Coronal Slice through front of the brain after thresholds are applied.

Intelligence, Vol. PAMI-9, No. 4, July, 1987, pp. 532-550. Different types of kernels are implemented and each of these gives different results, some with more side effects. This effect depends on the chosen type. It is possible to change type choosing 'parameter' → 'Img. Procs ... ' → 'ON'.

Some of the side effects can be small modulations of the edge of the brain, usually the edge will be more smooth.

The 'opening' operator is defined as:

$$mask = dilate(erode(mask, kernel), kernel)$$

where 'kernel' is implemented as one of these 6 types (1 is active):

Kernel	Shape	Side Effect	Figure
1 (3d)	000 010 000 $\uparrow$ 010 111 010 y 000 010 000 $x \rightarrow$ $z=0$ $z=1$ $z=2$	rounded edges in 3d	(A.7)
2 (2d)	000 010 000 $\uparrow$ 000 111 000 y 000 010 000 $x \rightarrow$ $z=0$ $z=1$ $z=2$	rounded edges in 2d	(A.8)
3 (3d)	010 111 010 $\uparrow$ 111 111 111 y 010 111 010 $x \rightarrow$ $z=0$ $z=1$ $z=2$	tends to make more squared edges	(A.9)
4 (2d)	000 111 000 $\uparrow$ 000 111 000 y 000 111 000 $x \rightarrow$ $z=0$ $z=1$ $z=2$	tends to make more squared edges	(A.10)
5 (3d)	11 11 11 $\uparrow$ 11 11 11 y 11 11 11 $x \rightarrow$ $z=0$ $z=1$ $z=2$	Is not symmetrical and tends to make more squared edges	(A.11)
6 (2d)	00 11 00 $\uparrow$ 00 11 00 y 00 11 00 $x \rightarrow$ $z=0$ $z=1$ $z=2$	Is not symmetrical and tends to make more squared edges	(A.12)

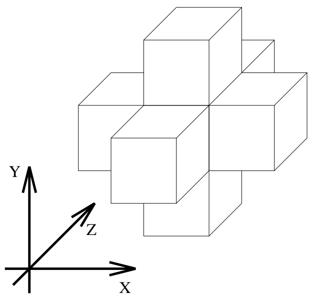

Figure A.7: Kernel type 1.

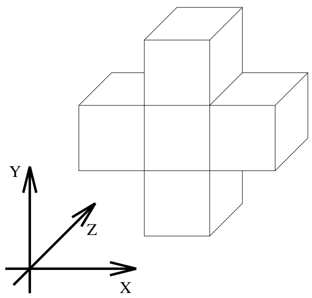

Figure A.8: Kernel type 2.

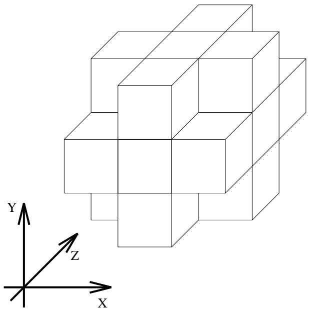

Figure A.9: Kernel type 3.

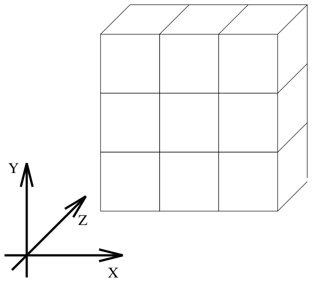

Figure A.10: Kernel type 4.

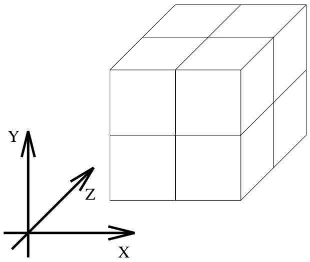

Figure A.11: Kernel type 5.

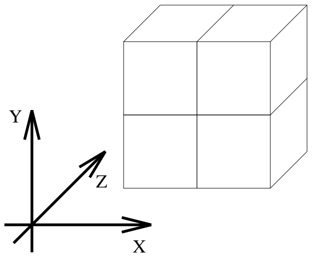

Figure A.12: Kernel type 6.

The third step is a 3d region growing with the purpose of selecting the brain as a connected region and at that time remove 'outside' defined regions. The region growing function needs a seed to grow from. This seed can be picked automatically or manually. If automatic pick fails to find a point in the brain, manual pick is required Figure (A.13). It uses the idl function *search3d*.

The fourth step is a step to fill in small holes in the brain. It use region growing as in step 3, but now the seed is outside the brain. This makes it possible to fill out holes within the brain if these holes are not connected to the outside ex, through a vein, and the intensity value in the pd-scan must be above a low threshold. The last restriction is to avoid including veins within the brain.

The final result can be seen in Figures (A.14) and (A.15).

A.3 Concluding Remarks

The semi automated segmentation program can speed up the process of manual segmentation, but the method still needs some trial and error in the process of determining the thresholds which can be trained. The method is now a part of the standard procedure in processing MR images at BIAL, UCSD [Jernigan et al.,]. Influence of inhomogeneities in the MR scan has a major impact on the performance and the MR scan has to be corrected first.

Figure A.13: Window for manually defining brain, note reverse colormap to indicate new behavior

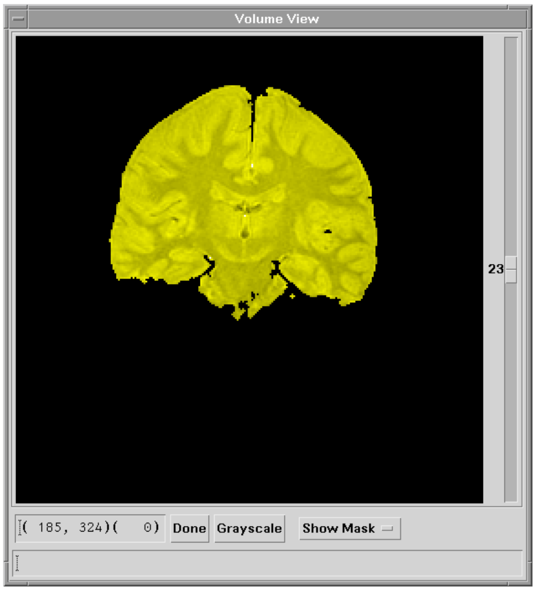

Figure A.14: Result after thresholds are applied.

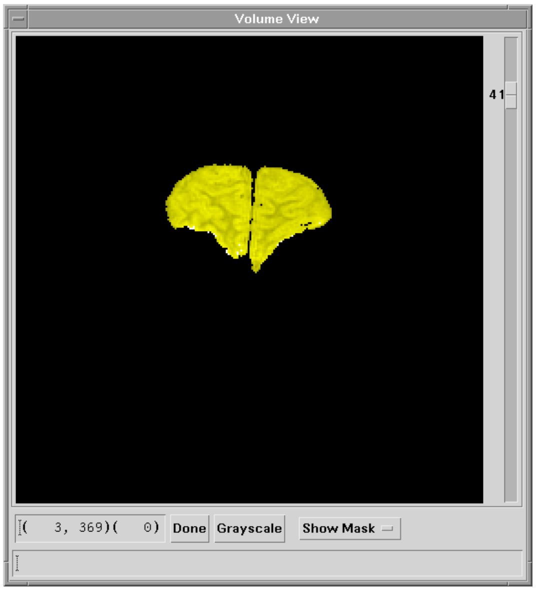

Figure A.15: Result after thresholds are applied.

Appendix B

Contribution to IIC95

This appendix contains the paper "Mean Field Reconstruction with Snaky Edge Hints" [Philipsen et al., 1996], presented at the 1995 Interdisciplinary Inversion Conference in Århus, Denmark.

Mean Field Reconstruction with Snaky Edge Hints

Peter Alshede Philipsen, Lars Kai Hansen and Peter Toft

CONNECT, Electronics Institute, build. 349
Technical University of Denmark, DK-2800 Lyngby, Denmark
email: pap,lkhansen,ptoft@ei.dtu.dk

1 Introduction

Reconstruction of imagery of a non-ideal imaging system is a fundamental aim of computer vision. Geman and Geman [Geman(1984)] introduced Metropolis sampling from Gibbs distributions as a simulation tool for visual reconstruction and showed that a *Simulated Annealing* strategy could improve the efficiency of the sampling process. The sampling process is implemented as a stochastic neural network with symmetric connections. Peterson and Anderson [Peterson(1987)] applied the *Mean Field* approximation, and observed substantial improvements in speed and performance.

In the next section the Bayesian approach to reconstruction is outlined and we study the so-called *Weak Membrane* model, as a model example. The Weak Membrane is a popular vehicle for piece wise smooth reconstruction and involves edge units (called line processes in [Geman(1984)]). Edge unit control has shown to be a major challenge in applications of the Weak Membrane. However, recently there has been substantial progress in use of deformable models for contour (2D) and surface (3D) modeling, see, e.g., [Cohen(1993), Chen(1994)]. In this contribution we suggest to combine the two approaches, in particular we show how a "Snake" contour model may be used to produce efficient edge hints to the Weak Membrane. Finally, section three contains experiments and concluding remarks.

2 Bayesian Visual Reconstruction

The basic idea is to consider both the source (un-degraded) signal and the processes of the imaging system as stochastic processes. The Bayes formula can then be used to obtain the distribution $P(V|d)$ of the reconstructed signal

V, conditioned on the observed degraded signal d:

$$P(V|d) = \frac{P(d|V)P(V)}{P(d)} \tag{1}$$

This conditional distribution is the product of the distribution of the imaging system process: $P(d|V) \equiv P(V \rightarrow d)$, and the *prior* distribution of the reconstructed signal $P(V)$. $P(V|d)$ of Equation (1) is referred to as the *posterior* distribution. A useful estimate of the reconstructed signal is given by the location of the mode of the posterior distribution, the so-called *Maximum A Posteriori* estimate.

2.1 The Weak Membrane Model

The *prior* distribution reflects our general insight into the objects being imaged, for example expressing that the image represents extended 3D structures etc. The Weak Membrane is a simple model for reconstruction of piece-wise continuous intensity surfaces (2D signals) from noisy observations [Geman(1984)]. In the lattice $(m, n) \in Z^2$ version the reconstructed surface is described by the intensity values $V_{m,n}$. The prior information formalizes the expectation that neighbor intensity values should be close, except when they are disconnected by the rare occasion of an active edge-unit. We introduce horizontal edges $h_{m,n}$ and vertical edges $v_{m,n}$. The *prior* probability distribution for a membrane reads:

$$P(V, h, v) = Z_1^{-1} \exp\left(-E_{\text{prior}}(y, h, v)\right) \tag{2}$$

where $E_{\text{prior}}(y, h, v)$ is the energy or cost function:

$$\begin{aligned} & E_{\text{prior}}(V, h, v) = \\ & \frac{\nu}{2}\left[\sum_{m,n}(1 - h_{m,n})(V_{m,n} - V_{m+1,n})^2 + (1 - v_{m,n})(V_{m,n} - V_{m,n+1})^2\right] \\ + & \sum_{m,n} \mu^h_{m,n} h_{m,n} + \sum_{m,n} \mu^v_{m,n} v_{m,n} - \sum_{m,n} \gamma_{m,n} h_{m,n} h_{m,n+1} v_{m,n} v_{m+1,n} \end{aligned} \tag{3}$$

and Z_1^{-1} is a normalization constant. This is a *Gibbs* distribution[1]. Note that the last terms in the cost function act as local *chemical potentials* for control of the number of active edge-units (i.e., $h_{m,n} = +1$).

[1] A distribution of the form $P(x) = Z^{-1} \exp(-E(x)/T)$, where $E(x)$ is a cost-function, bounded from below, and T is a parameter.

The degradation process produces the measurements ($V_{m,n} \to d_{m,n}$). In the Weak Membrane example we will for simplicity study addition of zero-mean, white Gaussian noise [2] with variance σ^2 :

$$P[d|V] = Z_2^{-1} \exp\left(-\frac{1}{2\sigma^2}\sum_{m,n}(V_{m,n} - d_{m,n})^2\right) \tag{4}$$

Using Bayes formula (1), we can combine Equations (2) and (4), to obtain the parameterized posterior distribution:

$$P[V, h, v|d] = Z^{-1} \exp(-E(V, h, v, d)) \tag{5}$$

where Z is a normalization constant and the energy function is given by

$$E(V, h, v, d) = \frac{1}{2\sigma^2}\sum_{m,n}(V_{m,n} - d_{m,n})^2 + E_{\text{prior}}(V, h, v). \tag{6}$$

2.2 Network design

Inspecting the posterior distribution we note that the three-component process (V, h, v) realizes a *Compound Random Markov Field.* This property ensures that the neural network implementation only involves neighbor connectivity. To enhance the sampling efficiency we use a simulated annealing strategy as recommended by Geman and Geman [Geman(1984)]. The scheme is implemented by introducing a *temperature* T in the Gibbs distribution and designing the sampler with a decreasing sequence of temperatures endinged at $T = 1$. The temperature dependent distribution reads:

$$P[V, h, v|d] = Z_T^{-1} \exp\left(-\frac{E[V, h, v, d]}{T}\right) \tag{7}$$

As investigated by Geman and Geman, sampling of this distribution is too slow for most applications. It is therefore recommended to invoke a deterministic approximation scheme to obtain the necessary averages [Blake(1989), Peterson(1987)]. The self-consistent Mean Field equations for the Weak Membrane model in the k'th iteration read (see also [Hertz(1991)] for an introduction):

$$\begin{aligned}
V_{m,n}^{k+1} &= \kappa^{-1}\left[\sigma^{-2} d_{m,n} + (1 - h_{m,n}^k)V_{m+1,n}^k + (1 - h_{m-1,n}^k)V_{m-1,n}^k \right. \\
&\quad \left. + (1 - v_{m,n}^k)V_{m,n+1}^k + (1 - v_{m,n-1}^k)V_{m,n-1}^k\right] \\
h_{m,n}^{k+1} &= \tanh\left[\beta\left(\frac{1}{2}(V_{m,n}^k - V_{m+1,n}^k)^2 - \mu_{m,n}^h - \gamma_{m,n}\,\Gamma_{m,n}^h(k)\right)\right] \\
v_{m,n}^{k+1} &= \tanh\left[\beta\left(\frac{1}{2}(V_{m,n}^k - V_{m,n+1}^k)^2 - \mu_{m,n}^v - \gamma_{m,n}\,\Gamma_{m,n}^v(k)\right)\right]
\end{aligned} \tag{8}$$

[2] It is possible to incorporate any other model of the degeneration process.

where $\beta = 1/T$ and

$$\begin{aligned}
\Gamma^h_{m,n}(k) &= h^k_{m,n+1} v^k_{m,n} v^k_{m+1,n} + h^k_{m,n-1} v^k_{m,n-1} v^k_{m+1,n-1} && (9)\\
\Gamma^v_{m,n}(k) &= v^k_{m+1,n} h^k_{m,n} h^k_{m,n+1} + v^k_{m-1,n} h^k_{m-1,n} h^k_{m-1,n+1} && (10)\\
\kappa &= \sigma^{-2} + 4 - h^k_{m,n} - h^k_{m-1,n} - v^k_{m,n} - v^k_{m,n-1} && (11)
\end{aligned}$$

We have used the same symbols for the averaged quantities as for the stochastic ones, but that should not lead to confusion since the two never occur in the same expression. The coupled equations are solved by straightforward iteration as shown in Equation (8), defining a recursive nearest neighbor connected *cellular neural network*.

2.3 Snake hints

There has been much recent progress in the use of deformable models for edge and surface identification, see, e.g., [Cohen(1993)]. In this presentation we suggest to produce strong edge hints for the Weak Membrane through a Snake deformable model. The Snake is defined to be a periodic set of N points in the visual field of an image: $\mathbf{r}_j = (x_j, y_j)$. Periodicity meaning $\mathbf{r}_{j+pN} = \mathbf{r}_j, p \in Z$. The Snake energy function consists of a form control part E_{form} and a match part E_{match}.

$$E = E_{\text{form}} + E_{\text{match}} \tag{12}$$

In this simple version the form control preserves total length,

$$\begin{aligned}
E_{\text{form}} &= \frac{a}{2}(d - d_0)^2 && (13)\\
d &= \sum_j |\mathbf{r}_j - \mathbf{r}_{j+1}| && (14)
\end{aligned}$$

where d_0 is the initial Snake length and a is a parameter to control the balance between the two parts in the energy function. More complex form controls can be designed that preserve shape, corners, etc. The Snake match energy is designed to ensure that the Snake points track edge contours in the image field. This effect may be obtained by letting the Snake points seek local maxima in the gradient energy map of the image, $G(\mathbf{r})$, evaluated from the observed image. The total Snake match energy is then given by,

$$E_{\text{match}} = -\sum_j G(\mathbf{r}_j). \tag{15}$$

Snake dynamics is established through gradient descent,

$$\frac{1}{\eta}\frac{\partial \mathbf{r}_j}{\partial t} = -\frac{\partial E}{\partial \mathbf{r}_j} = a(d - d_0)\frac{\partial d}{\partial \mathbf{r}_j} + \frac{\partial G(\mathbf{r}_j)}{\partial \mathbf{r}_j} \tag{16}$$

where η is a parameter to control the step size in the gradient direction.

3 Experiments and concluding remarks

In numerous image processing applications quite detailed prior information can be devised. In, e.g., brain scan reconstruction detailed atlases are known describing the generic brain topography under various scanning modes. To illustrate the potential of using strong edge hints in brain scans, we want to reconstruct a head phantom shown in Figure 1. This is done from a noise corrupted image shown in Figure 2. During the reconstruction process we also want to estimate edges, due to the interpretation of the image.

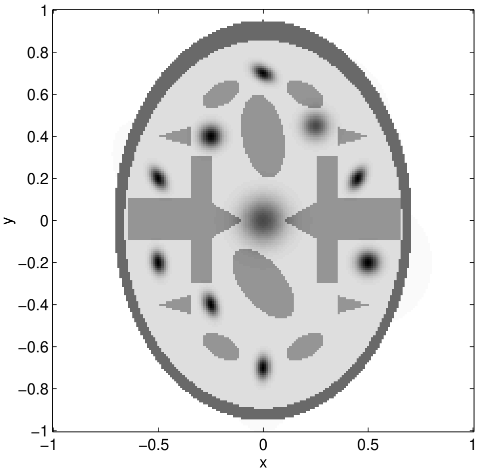

Figure 1: Original head phantom without noise.

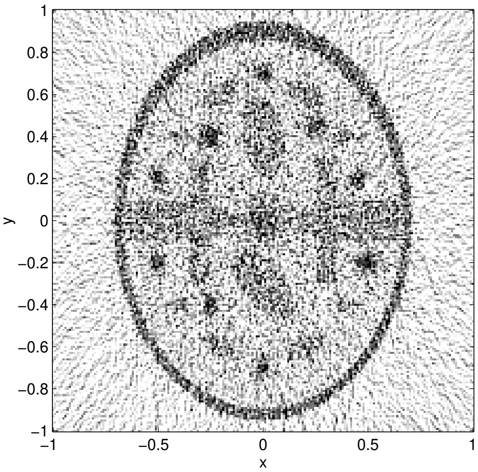

Figure 2: Noise corrupted head phantom shown in same color scale.

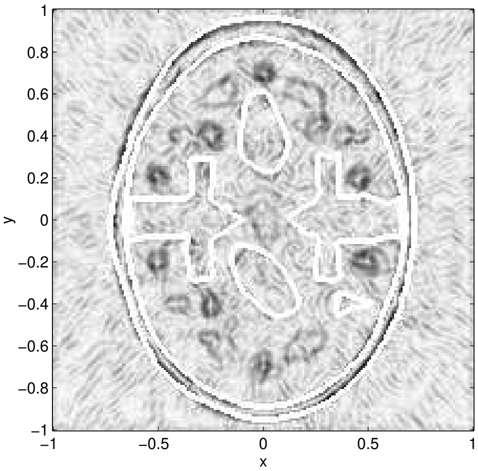

Figure 3: Edge energy map with initial snake positions shown in white.

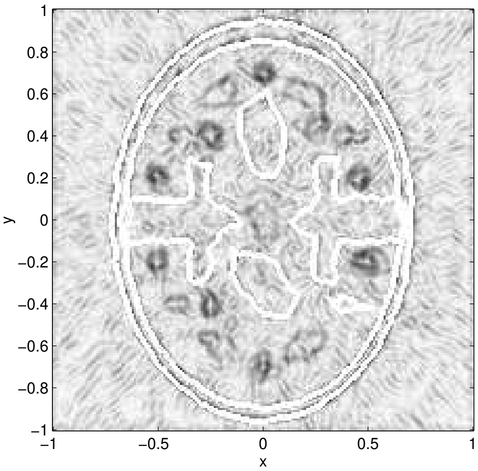

Figure 4: Edge energy map with equilibrated Snake positions shown in white.

Snakes are initialized in seven generic positions, as shown in Figure 3. For this particular instance the Snake equilibrates in about 100 iterations as illustrated in Figure 4. The approach to equilibrium is quite sensitive to the topography of the image and careful control of η is necessary, Equation (16). This problem may be relieved by use of a pseudo-second order search direction instead of the gradient of the edge energy map.

Subsequently, hints are created by modulating the chemical potentials $\mu^h_{m,n}$, $\mu^v_{m,n}$ and $\gamma_{m,n}$. This modulation implies that edge units are strongly suppressed outside of the region suggested by the Snakes. Finally, we relax the cellular network as defined in Equation (8). The result of the cellular net is presented in Figures 5 (without Snakes) and 6 (using Snakes).

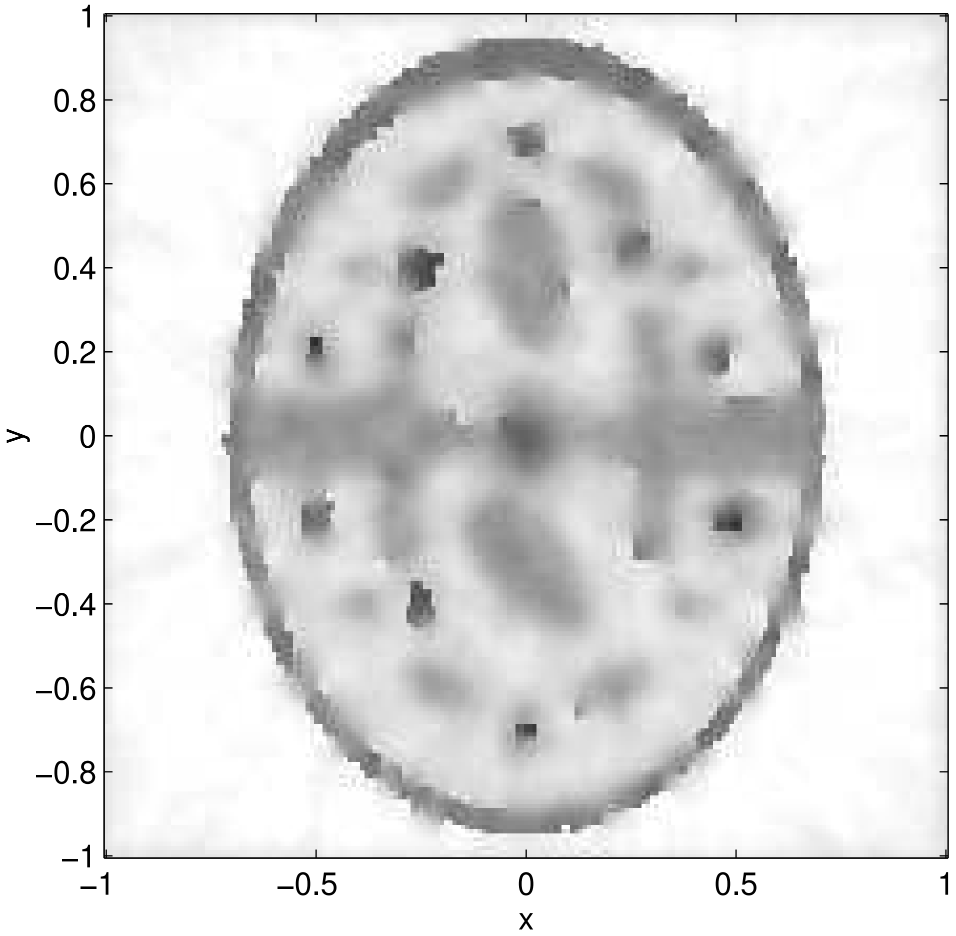

Figure 5: The restored head phantom without using Snakes edge priors to the Mean Field Annealing.

Note that the intensity levels inside the regions of the Snakes have been estimated more closely using Snakes. This might be of significant importance in, e.g., brain activation studies, in which the minute differences in brain activity between activated and resting states are investigated. Using Snakes and Mean Field Annealing enhances the Signal to Noise ratio (SNR) with 12.5 dB. Without Snakes the improvement is 11.3 dB. The improvement is due to a better estimation of edges.

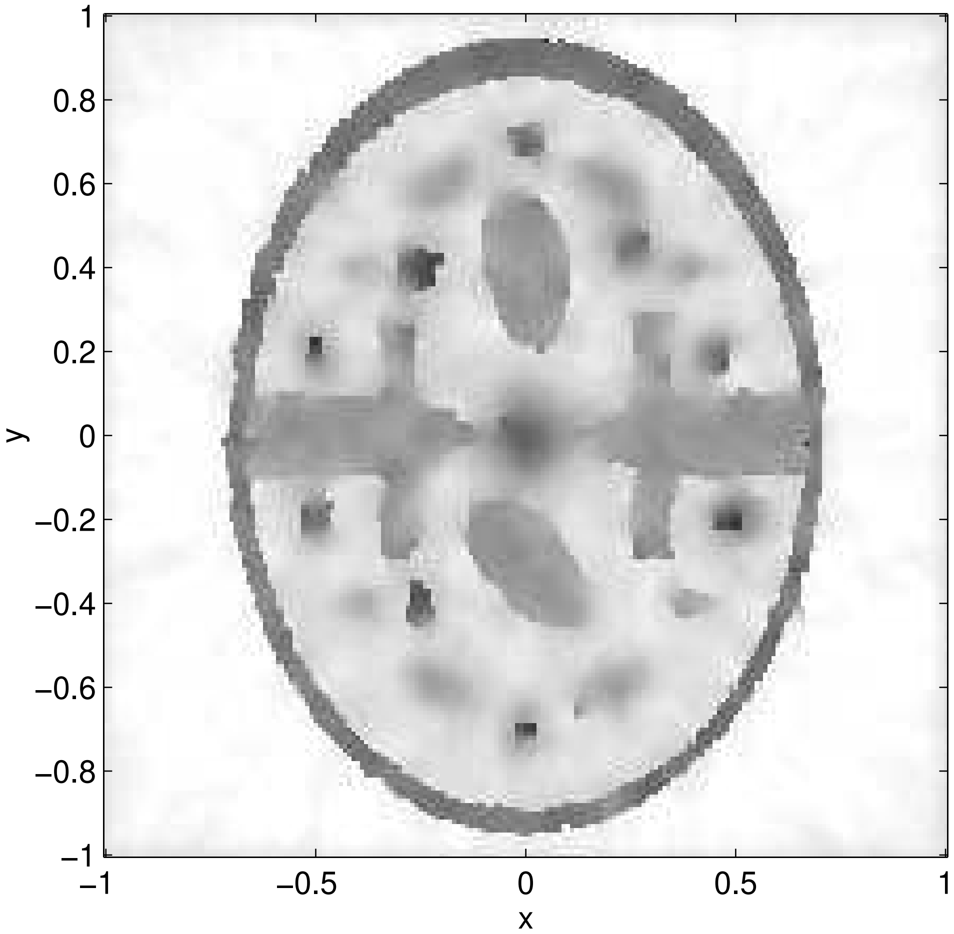

Figure 6: The restored head phantom using Snakes as edge priors to the Mean Field Annealing.

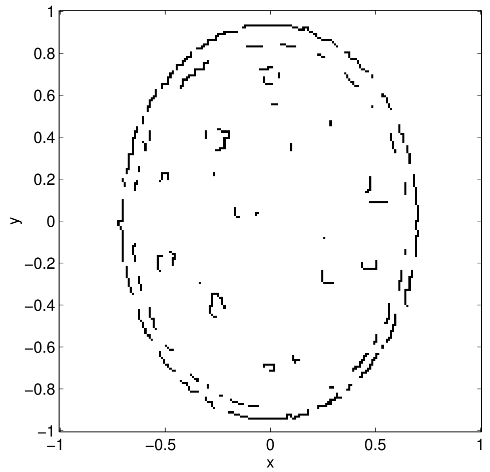

Figure 7: The estimated edges without using Snakes. Active edges are marked black.

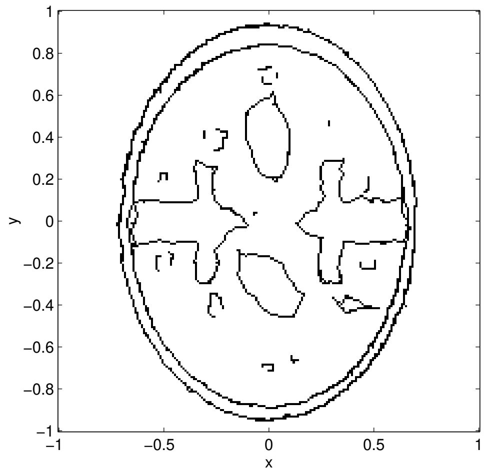

Figure 8: Estimated edges using Snakes as priors. Active edges are marked black.

In Figures 7 and 8 are shown the active edges (i.e., $h_{m,n} = +1$ or $v_{m,n} = +1$) without using Snakes respectively with Snakes. As seen from Figure 8 it is possible to incorporate strong priors in order to obtain closed edge contours. This can be used to segment the image.

In conclusion we have shown that strong structural priors may be introduced in the Weak Membrane model by invoking deformable models like Snakes. The present study was based on a head phantom; we are currently pursuing the viability of the approach in the context of Positron Emission Tomography.

Acknowledgment

This research is supported by the Danish Research Councils for the Natural and Technical Sciences through the Danish Computational Neural Network Center.

References

[Blake(1989)] A. Blake: *Comparison of the Efficiency of Deterministic and Stochastic Algorithms for Visual Reconstruction.* IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-11** 2-12 (1989).

[Cohen(1993)] L. Cohen and I. Cohen: *Finite Element Methods for Active Contour Models and Balloons for 2-D and 3-D Images.* IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-15** 1133-1147 (1993).

[Chen(1994)] X. Ouyang, W.H. Wong, V.E. Johnson, X. Hu, C.-T. Chen: *Incorporation of Correlated Structural Images in PET Image Reconstruction.* IEEE Transactions on Medical Imaging **13** 627-640 (1994).

[Geman(1984)] D. Geman and S. Geman: *Stochastic Relaxation, Gibbs distributions and the Bayesian Restoration of Images.* IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-6** 721-741 (1984).

[Hertz(1991)] J. Hertz, A. Krogh and R.G. Palmer: *Introduction to the Theory of Neural Computation.* Addison Wesley, New York (1991).

[Peterson(1987)] C. Peterson and J.R. Anderson: *A Mean Field Learning Algorithm for Neural Networks.* Complex Systems **1** 995-1019, (1987).

[Roth(1990)] M.W. Roth: *Survey of Neural Network Technology for Automatic Target Recognition.* IEEE Transactions on Neural Networks **1**, 28-43, (1990).

Appendix C

Contribution to IIW96

This appendix contains the paper "Brain Mapping by Positron Emission Tomography" [Toft et al., 1996], presented at the 1996 Interdisciplinary Inversion Workshop in Lyngby, Denmark.

1. Brain Mapping by Positron Emission Tomography

Peter Toft, Peter Alshede Philipsen, Lars Kai Hansen, Ulrik Kjems
Department of Mathematical Modelling, Technical University of Denmark,
Building 305, DK-2800 Lyngby, Denmark, Email pto,pap,lkh,uk@imm.dtu.dk.

Abstract

In this paper the basis of PET (Positron Emission Tomography) is reviewed, and it is shown that the measured signals can be modelled as the Radon transform of the desired spatial distribution of, e.g., the brain activity.

Next, two of the direct reconstruction methods are presented. Both are derived from inversion of the Radon transform. It is shown that the reconstruction can be based on filtering and integration techniques.

Another major class of reconstruction techniques is presented, namely the linear algebra based methods, which often are formed as iterative methods. A very fast way of implementing a set of iterative reconstruction techniques is shown along with a set of examples.

1.1 Introduction to PET

The PET scanner is based on radioactive tracers. A small dosage of a β^+ emitter, such as O-15 or C-11, is injected into the patient (or the object to be scanned). The β^+ emitter will be distributed in the tissue due to the blood circulation. If for instance the brain is to be scanned then the β^+ emitter will decay in correspondence to the activity of the brain. In the regions with high brain activity the blood flow will be high, and many of the radioactive nucleons will decay under emission of a positron in these regions.

Within a few mm the positron will interact with an electron and annihilate under emission of two 511 keV photons. The two photons will travel in opposite directions outward to the PET scanner where the photons are detected at sensors placed in a ring, as illustrated on Fig. 1.1.

The PET hardware will sort the arrival times of the photons, so only two photons that arrived at the same time at the sensor ring are taken into account (The photons travel with the speed of light, and here the small difference in arrival time is neglected). A two dimensional matrix of the possible line parameters is created, and here denoted by $E(\rho, \theta)$. If the two photons form a line with parameters $(\rho, \theta) = (\rho_0, \theta_0)$, the array is incremented with the value one at that position in the array. This is because the only obtainable

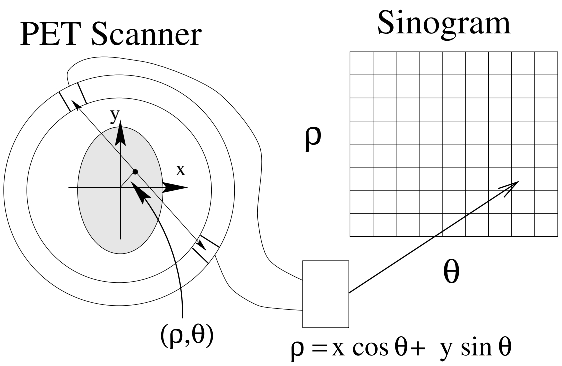

Figure 1.1: Emission of two photons from the place of decay.

information from the two photons is that the photon emission took place somewhere along that line. Depending on the radioactive dosage given to the patient many decays take place per second in each volume element. After the recording is terminated, an array of emissions has been recorded. For $(\rho, \theta) = (\rho_0, \theta_0)$ the array is developed by several β^+ emissions which took place along the line. The individual emission recordings are no longer stored. The obtained array of emissions (called the *sinogram*) are in each position approximately proportional to the emission intensity along that particular line (times the total recording time T). The obtained matrix $E(\rho, \theta)$ divided by T is in mathematical terms an integral of the emission intensity $g(x, y)$ over that particular line.

$$\frac{E(\rho,\theta)}{T} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(x,y)\ \delta(\rho - x\cos\theta - y\sin\theta)\ dx\ dy\ T = \breve{g}(\rho,\theta) \tag{1.1}$$

It can be recognized that the recorded array $E(\rho, \theta)$ divided by T is the Radon transform of the emission intensity $g(x, y)$, here denoted by $\breve{g}(\rho, \theta)$ [1],[2]. Note that this intensity will not be a constant in time as it varies with the activity of the brain. This will limit the allowable recording time.

Previously absorption of the photons in the tissue has been neglected, thus the inversion process must be modified to take this into account. In principle, this is easy when using double photon emissions. It can be shown, e.g. [2], that it only requires that the sinogram of emissions must be multiplied in each sample (pixel) with a sinogram corresponding to the transmission. This implies that reconstruction of PET images can be implemented as an inverse Radon transform of a sinogram.

1.2 Direct Reconstruction Methods

1.2.1 The Fourier Slice Theorem

Inversion of the Radon transform can be done in several ways. One standard algorithm is based on the Fourier Slice Theorem also known as the Central Slice Theorem [3],[4]. The Radon transform $\breve{g}(\rho, \theta)$ is to be inverted into $g(x, y)$, where the inversion is based on the Fourier transform.

First the 2D Fourier Transform of $g(x, y)$ is needed.

$$G(k_x, k_y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y)\ e^{-j2\pi(k_x x + k_y y)}\ dx\ dy \tag{1.2}$$

Introducing polar frequency parameters

$$\begin{pmatrix} k_x \\ k_y \end{pmatrix} = \nu \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} \tag{1.3}$$

and inserting in Eq. (1.3) into Eq. (1.2) gives

$$G(\nu\cos\theta, \nu\sin\theta) = \int_{-\infty}^{\infty} \breve{g}(\rho, \theta)\ e^{-j2\pi\rho\nu}\ d\rho \tag{1.4}$$

Thus, a 1D Fourier Transform of the Radon transform gives the 2D spectrum of $g(x, y)$, hence the Fourier Slice Theorem makes it possible to invert the Radon transform by use of 1D Fourier transforms and 2D Fourier transforms. Note, that the theorem can be used in reverse order to calculate the Radon transform, if the signal $g(x, y)$ is given.

1.2.2 Filtered Backprojection

Another very famous inversion scheme is the Filtered Backprojection [5],[6] or as it more correctly should have been named Backprojection of Filtered Projections method [1]. It is derived from the inverse of Eq. (1.2), by introducing polar coordinates

$$\begin{aligned} g(x, y) &= \int_0^{2\pi} \int_0^{\infty} \nu\ G(\nu\cos\theta, \nu\sin\theta)\ e^{j2\pi\nu(x\cos\theta + y\sin\theta)}\ d\nu\ d\theta \\ &= \int_0^{\pi} \int_{-\infty}^{\infty} |\nu|\ G(\nu\cos\theta, \nu\sin\theta)\ e^{j2\pi\nu(x\cos\theta + y\sin\theta)}\ d\nu\ d\theta \end{aligned} \tag{1.5}$$

and inserting Eq. (1.4) and splitting the formula into two parts, then Filtered Backprojection is written in two parts: A filtering part and an integration part.

$$\bar{\breve{g}}(\rho, \theta) = \int_{-\infty}^{\infty} |\nu| \left(\int_{-\infty}^{\infty} \breve{g}(\tilde{\rho}, \theta)\ e^{-j2\pi\tilde{\rho}\nu}\ d\tilde{\rho} \right)\ e^{j2\pi\rho\nu}\ d\nu \tag{1.6}$$

$$g(x, y) = \int_0^{\pi} \bar{\breve{g}}(x\cos\theta + y\sin\theta, \theta)\ d\theta = \int_0^{\pi} \int_{-\infty}^{\infty} \bar{\breve{g}}(\rho, \theta)\ \delta(\rho - x\cos\theta - y\sin\theta)\ d\rho\ d\theta \tag{1.7}$$

Note, the similarity of Eq. (1.7) to the Radon transform in Eq. (1.1). The operation shown in Eq. (1.7) is named backprojection and it can be shown (page 120 of [1]) that the backprojection operator is closely linked to the adjoint Radon transform.

The filtering part in Eq. (1.6) is a forward Fourier transform for each of the angles θ. In the Fourier domain the signal is high pass filtered with the filter $|\nu|$. Subsequently an inverse 1D-Fourier transform is used to get the filtered Radon domain $\bar{\breve{g}}(\rho, \theta)$.

The remaining backprojection part is merely an integration along a sine-curve in the filtered Radon domain. Eqs. (1.6) and (1.7) form the Filtered Backprojection inversion scheme. A movie showing the Filtered Backprojection method can be found at the URL `http://hendrix.imm.dtu.dk` under the movies entry.

1.3 Reconstruction Algorithms Based on Linear Algebra

Inversion of the Radon transform need not be based on the direct inverse formulas as shown in Section 1.2. A different class of reconstruction schemes is based on linear algebra [7],[1],[8].

The (discrete) Radon transform is a linear transform, with respect to the function $g(x, y)$, hence instead of considering the integral version of the Radon transform a matrix representation can used

$$\begin{array}{ccccc} \breve{g}(\rho,\theta) & = & \mathcal{R} & g(x,y) \\ \downarrow & & \downarrow & \downarrow \\ \boldsymbol{b} & = & \boldsymbol{A} & \boldsymbol{x} \end{array} \tag{1.8}$$

In the last 15 years the iterative reconstruction methods have gained much attention in the literature [7],[8]. Several methods have been very prominent, such as EM (Expectation Maximization) [5],[9],[10],[11], ART (Algebraic Reconstruction Technique) [12],[7],[13], and LSCG (Least Squares Conjugate Gradient) [14],[15].

These methods formulate the reconstruction problem as a linear set of equations

$$\boldsymbol{b} = \boldsymbol{A}\boldsymbol{x} \;\Leftrightarrow\; b_i = \sum_{j=1}^{J-1} a_{i,j} x_j, \quad i = 1, 2, \ldots, I \tag{1.9}$$

where $\boldsymbol{b}$ is an I-dimensional vector containing the known sinogram values wrapped into a vector, and $\boldsymbol{x}$ is a J-dimensional vector containing the unknown image to be reconstructed. Here $\boldsymbol{A}$ is the system matrix, which contains the weight factors between each of the image pixels and each of the values in the sinogram, corresponding to line orientations. The use of linear algebra has several advantages, such as easier incorporation of irregular geometries. The system matrix can model several real-world properties, such as finite, i.e., non-zero detector size and varying detector sensitivity. Furthermore, regularization can easily be incorporated [16],[14] in order to affect the often ill-conditioned reconstruction problem.

One problem is the huge size of the system matrix. A 2D sinogram from, e.g., a GE Advance PET scanner contains $I = 281*336$ values, and reconstructed into a $J = 301*301$ grid, i.e., the system matrix has approximately 8.5 billion elements, requiring over 34 GBytes of memory when using 4 bytes per matrix element. This is a large amount of memory, even looking some years into the future. Besides this aspect it would not be wise to store all that data, due to the fact that approximately 98% of the matrix entries will be zeros. This knowledge should be incorporated into the reconstruction schemes.

Assuming that memory is not available for storing the full system matrix, one possibility is to compute the individual matrix elements in each iteration when needed. This can be done by using the Radon transform, e.g., [2] or other modelling schemes for the scanner. This approach is rather easily implemented and is viable and storage requirements are reduced to a minimum, only requiring memory for the sinogram ($\boldsymbol{b}$) and the current solution ($\boldsymbol{x}$), and perhaps some additional temporary variables of the same size or smaller, but no system matrix is stored in memory. It will be demonstrated that this implementation has a major drawback in speed, since the system matrix will be computed many times during an iterative reconstruction. Each time at the same high computational cost.

1.4 Accelerated 2D Iterative Reconstruction

In this section a hybrid solution is investigated for accelerating the iterative reconstruction algorithms [17],[11],[18], but requiring as much memory as modern workstations are currently equipped with (or will be soon). The idea is to store the non-zero elements of the system matrix in the main memory using sparse matrix techniques. In this way the core of the reconstruction algorithms, highly based on matrix vector multiplications, can be accelerated significantly, and thereby solve one of the major drawbacks of the iterative methods.

It is proposed that the system matrix $\boldsymbol{A}$ is calculated one time only using all the modifications found for the actual scanner setup. If no specific scanner model is provided the system matrix can be modelled and generated using the Radon transform or other simpler schemes. From the system matrix the very small values in the matrix are truncated to zero. Let

$$\tilde{a}_{i,j} = \begin{cases} a_{i,j} & \text{if } a_{i,j} > \gamma \\ 0 & \text{Otherwise} \end{cases} \tag{1.10}$$

where the threshold γ can be chosen to a certain fraction of the maximum matrix value, e.g., $\gamma = 0.05 \max_{i,j}\{a_{i,j}\}$. If γ is chosen sufficiently low, a good compromise between resolution and the sparseness of the matrix can be reached, and normally this does not affect the behaviour of the algorithms. Current work concerns the quantification of the truncation error.

The sparse structure of $\boldsymbol{A}$ can be exploited by storing only non-zero values in the fast memory. For a certain row, number i, all of the matrix elements are calculated, stored, and truncated using Eq. (1.10). Hereby the number of non-zero elements in the row, denoted by Z_i, will be much smaller than the image size $J = M^2$. The values of Z_i are stored in a simple, one dimensional vector. Two vectors of length Z_i, indexed by an integer z, can then be allocated and stored containing the non-zero matrix value a_z and the corresponding column index j_z. The procedure is repeated for all rows.

Assuming a nearest neighbour approximation with one pixel for each point along the integration lines and using 4 bytes for storing each of the vector elements, the total storage requirement is then reduced to approximately $8\sum_{i=1}^{I} z_i \approx 8IM$ bytes. In the example shown above approximately 100 MBytes memory is required. Assuming this amount of memory is present most iterative algorithms can be implemented very fast from three basic operations: Matrix vector multiplication (forward projection) $\boldsymbol{A}\tilde{\boldsymbol{x}}$, scalar product between the i'th row of the system matrix and a vector $\boldsymbol{a}_i^T\tilde{\boldsymbol{x}}$, and finally multiplication with the transpose of the matrix (backprojection) $\boldsymbol{A}^T\tilde{\boldsymbol{b}}$.

A software package has been written in C including the proper structures for manipulating sparse matrices and vectors, along with an optimized code for computing matrix vector products, well suited for iterative reconstruction algorithms. In the package ART, EM, and LSCG are implemented both in a fast version using sparse matrix storage of the system matrix and in a slow version where the system matrix is not stored and needed matrix entries are computed as they are needed in each step of the iterative algorithms. The software package is available for free at `http://eivind.imm.dtu.dk/staff/ptoft`, but protected by the GNU General Public License.

Algebraic Reconstruction Technique (ART)

For a certain row i of the matrix (i depends on the iteration number k), the general iteration step incrementing the current solution $\boldsymbol{x}^{(k)}$ can be found in, e.g., [7]

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \frac{b_i - \boldsymbol{a}_i^T \boldsymbol{x}^{(k)}}{\boldsymbol{a}_i^T \boldsymbol{a}_i} \boldsymbol{a}_i \tag{1.11}$$

Expectation Maximization (EM)

The general iteration step of EM [10],[19] requires a forward projection, a backprojection, and two fast updates in each iteration.

$$x_j^{(k)} = \frac{x_j^{(k-1)}}{\sum_{i'=1}^{I} a_{i',j}} \sum_{i=1}^{I} \frac{a_{i,j} b_i}{\sum_{j'=1}^{J} a_{i,j'} x_j^{(k-1)}} \tag{1.12}$$

Least Squares Conjugate Gradient (LSCG)

The Least Squares Conjugate Gradient method requires some initialization [15]

$$\boldsymbol{s}^{(0)} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^{(0)} \tag{1.13}$$

$$\boldsymbol{r}^{(0)} = \boldsymbol{p}^{(0)} = \boldsymbol{A}^T \boldsymbol{s}^{(0)} \tag{1.14}$$

$$\boldsymbol{q}^{(0)} = \boldsymbol{A}\boldsymbol{p}^{(0)} \tag{1.15}$$

Then for each iteration the LSCG algorithm on the normal equations becomes

$$\alpha = \frac{\left(\boldsymbol{r}^{(k-1)}\right)^T \boldsymbol{r}^{(k-1)}}{\left(\boldsymbol{q}^{(k-1)}\right)^T \boldsymbol{q}^{(k-1)}} \tag{1.16}$$

$$\boldsymbol{x}^{(k)} = \boldsymbol{x}^{(k-1)} + \alpha \boldsymbol{p}^{(k-1)} \tag{1.17}$$

$$\boldsymbol{r}^{(k)} = \boldsymbol{A}^T \boldsymbol{s}^{(k-1)} \tag{1.18}$$

$$\boldsymbol{r}^{(k)} = \boldsymbol{r}^{(k-1)} - \alpha \boldsymbol{q}^{(k-1)} \tag{1.19}$$

$$\beta = \frac{\left(\boldsymbol{r}^{(k)}\right)^T \boldsymbol{r}^{(k)}}{\left(\boldsymbol{r}^{(k-1)}\right)^T \boldsymbol{r}^{(k-1)}} \tag{1.20}$$

$$\boldsymbol{p}^{(k)} = \boldsymbol{r}^{(k)} + \beta \boldsymbol{p}^{(k-1)} \tag{1.21}$$

$$\boldsymbol{q}^{(k)} = \boldsymbol{A}\boldsymbol{p}^{(k)} \tag{1.22}$$

For all three methods an initial value of the solution, i.e., $\boldsymbol{x}^{(0)}$ is needed. In the package an image found by, e.g., a fast direct method, can be supplied and used. If not provided, all of the initial values of the vector are initialized to a properly chosen constant.

1.5 Examples

The program has been used on two types of machines. A Linux machine with a 120 MHz Pentium processor and an Onyx from SGI equipped with four 200 MHz R4400 processors, where the program was running on one processor.

1.5.1 Reconstruction of a Small Sinogram

In the first example the (synthetic) sinogram has $125 * 101$ samples and the reconstructed image has $101 * 101$ samples. In Table 1.1 the reconstruction times on both machines are shown for the fast and the slow method as well as the ratio between the execution times (slow/fast).

Times are measured for ART, EM, and the LSCG-method, when EM and LSCG were running (arbitrarily) 20 iterations, and ART 20 full iterations, i.e., 20 times the number of rows (chosen randomly), which is 20∗125∗101 iterations in Eq. (1.11). Note that all times only correspond to the actual iterations. For the fast versions of the iterative reconstruction algorithms, the time to generate the system matrix once should be added if changing the system matrix, e.g., when changing the sampling parameters of the reconstructed image.

For this example the system matrix was modelled using discrete Radon transformation with linear interpolation, where the threshold γ was chosen to zero, hence the slow and the fast methods give exactly the same results. Note that the large difference in speedup between ART and EM/LSCG is due to the implementation of the forward projection is more efficient than the multiplication with the transpose of the system matrix. The slow methods can be accelerated some by implementing multiplication with the transpose of the system matrix (adjoint operator) as a backprojection integral, but note that this implies that the approximation of the system matrix will be different in the forward and the backprojection part. The sparse system matrix for this transformation geometry required approximately 13 MBytes, and each iteration requires approximately one second.

Machine	Type	ART	EM	LSCG
Pentium	Fast	26 sec	17 sec	17 sec
	Slow	2218 sec	5776 sec	5250 sec
	Ratio	85	340	309
SGI Onyx	Fast	16 sec	16 sec	16 sec
	Slow	1306 sec	2722 sec	2715 sec
	Ratio	82	170	170

Table 1.1: Time usage for 20 iterations of EM and LSCG. For ART the time is for 20 full iterations, i.e., 20∗125∗101 of the iterations used in Eq. (1.11). The time measurements are for a sinogram with $125 * 101$ samples reconstructed into a $101 * 101$ samples image.

1.5.2 Reconstruction of a Large Sinogram

In Fig. 1.2 a 2D sinogram with 281∗336 samples is shown, which was measured on a GE Advance PET scanner. The sinogram is reconstructed into a (large) image with 301∗301 samples. The system matrix has 94416∗90601 elements of which 0.23% are non-zero when modelling then system matrix using the Radon transform of a square. On the Onyx it required 1466 seconds to generate the 157 MBytes sparse matrix, and each iteration of EM required 15 seconds in the fast version and 11678 sec in the slow implementation. After 10 iterations the algorithm was stopped. The reconstructed image is shown in Fig. 1.3.

For sake of comparison, the same sinogram reconstructed image using Filtered Backprojection is shown in Fig. 1.4, and this reconstruction used 6 seconds on the Onyx. Next the LSCG method is used to reconstruct the sinogram shown in Fig. 1.2. The reconstructed image after 11 iterations is shown in Fig. 1.5 (Note that this particular image has only $101 * 101$ pixels). It can be seen that some low frequency oscillations are emerging from the edges. These can be controlled using regularization or constraints [2].

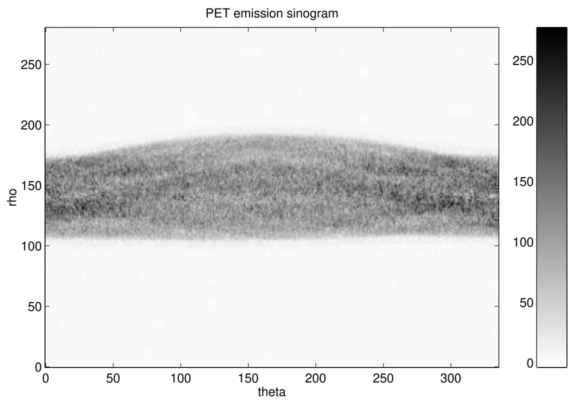

Figure 1.2: A PET sinogram of a human brain.

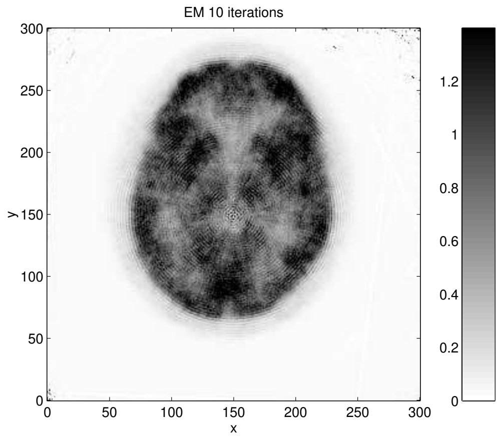

Figure 1.3: The reconstructed image after 10 iterations of EM.

1.6 Conclusion

We have described the basis of PET reconstruction techniques and demonstrated a very fast implementation of iterative reconstruction comparable in speed with direct reconstruction methods. The implementation is based on storing of the system matrix in fast memory using sparse techniques. The approach is mainly applicable to 2D reconstruction, due to the requirements of a sufficient amount of memory, but in principle the method can also be applied to 3D reconstruction.

The cost is that a large amount of memory is required, but for ART we have demonstrated a speedup factor of approximately 80 and for EM and LSCG 170-340 depending on the machine, for a fixed transformation geometry and interpolation level.

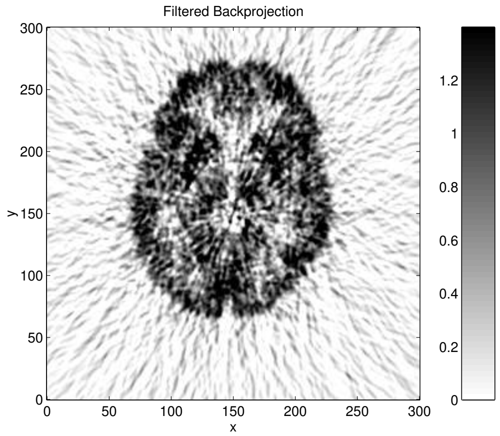

Figure 1.4: The reconstructed image using Filtered Backprojection with a ramp filter.

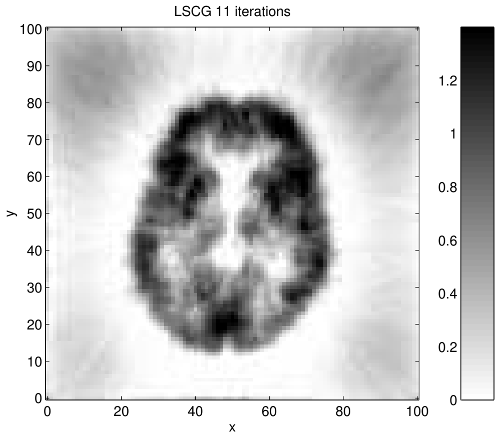

Figure 1.5: The reconstructed image using 11 iterations of the Least Squares Conjugate method. The artifacts around the brain can be removed by proper masking.

Acknowledgment

This research is supported by the Danish Research Councils for the Natural and Technical Sciences through the Danish Computational Neural Network Center.

Bibliography

[1] S. R. Deans. *The Radon Transform and Some of Its Applications.* Krieger Publishing Company, Malabar, Florida, 2 edition, 1993. Previously John Wiley & Sons Inc., 1983.

[2] Peter Toft. *The Radon Transform - Theory and Implementation.* PhD thesis, Department of Mathematical Modelling, Technical University of Denmark, 1996.

[3] R. M. Mersereau. Recovering Multidimensional Signals from their Projections. *Computer Graphics and Image Processing*, 1(4):179–195, Dec. 1973.

[4] Dan E. Dudgeon and Russel M. Mersereau. *Multidimensional Digital Signal Processing.* Prentice-Hall, Englewood Cliffs,N.J, 1984.

[5] L. A. Shepp and J. B. Krustal. Computerized tomography: The new medical x-ray technology. *Am. Math. Monthly*, 85:420–439, April 1978.

[6] A. Rosenfeld and A. C. Kak. *Digital Picture Processing*, volume 1. Academic Press, 2. ed. edition, 1982.

[7] Yair Censor. Finite Series-Expansion Reconstruction Methods. In *Proc. of the IEEE*, volume 71, pages 409–419, March 1983.

[8] Julia K. Older and Paul C. Johns. Matrix formulation of computed tomography reconstruction. *Physics in Medicine & Biology*, 38:1051–1064, 1993.

[9] L. A. Shepp Y. Vardi and L. Kaufman. A Statistical Model for Positron Emission Tomography. *Journal of the American Statistical Association*, 80(389):8–37, March 1985. The pages include comments and discussion by several authors.

[10] R. E. Carson and K. Lange. The EM parametric image reconstruction algorithm. *Journal of the American Statistical Association*, 389(80):20–25, March 1985.

[11] Linda Kaufman. Implementing and Accelerating the EM Algorithms for Positron Emission Tomography. *IEEE Trans. Med. Imag.*, MI-6(1):37–51, march 1987.

[12] Gabor T. Herman and Lorraine B. Meyer. Algebraric Reconstruction Techniques Can Be Made Computationally Efficient. *IEEE Trans. Med. Imag.*, 12(3):600–609, Sep. 1993.

[13] Huaiqun Guan and Richard Gordon. A projection access order for speedy convergence of ART (algebraric reconstruction technique): a multilevel scheme for cumputed tomography. *Phys. Med. Biol.*, 39:2005–2022, 1994.

[14] Linda Kaufman and Arnold Neumaier. Image Reconstruction Through Regularization by Envelope Guided Conjugate Gradients. Technical report, Bell Labs, 1993. Report 11274-94-14.

[15] John A. Scales. Iterative methods for large, sparse, inverse calculations. Samizdat Press. Colorado School of Mines, 76 Olcott Drive, White River Junction, Vermont 05001, April 1993. Lecture notes.

[16] Per Christian Hansen. *Rank-Deficient and Discrete Ill-Posed Problems*. Polyteknisk Forlag, Lyngby, Denmark, 1996. Doctoral Dissertation.

[17] Tom R. Miller and Jerold W. Wallis. Fast maximum-likelihood reconstruction. *Journal of Nuclear Medicine*, 33(9):1710–1711, September 1992.

[18] Peter Toft and Jesper James Jensen. A Very Fast Implementation of 2D Iterative Reconstruction Algorithms. *Accepted for presentation at the 1996 IEEE Medical Imaging Conference*, 1996.

[19] G. T. Herman and D. Odhner. Performance evaluation of an iterative image reconstruction algorithm for positron emission tomography. *IEEE Trans. Med. Imag.*, 10:336–246, 1991.

Appendix D

Contribution to IIW97

This appendix contains the paper "Restoring Functional PET Images using Anatomical MR Images" [Philipsen et al., 1997] , presented at the 1997 Interdisciplinary Inversion Workshop in Århus, Denmark.

Restoring functional PET Images using Anatomical MR Images

Peter Alshede Philipsen, Ulrik Kjems, Peter Toft and Lars Kai Hansen,
Department of Mathematical Modelling, Section for Digital Signal Processing,
Building 321, Technical University of Denmark, DK-2800 Lyngby, Denmark
email: pap,uk,pto,lkh@imm.dtu.dk

Abstract

In this paper we present a Bayesian method to enhance functional 3D PET images using apriori knowledge about the brain anatomy obtained from 3D MR images. We use a Markov Random Field as a prior model, which have the possibility of incorporating this information. We show both a small test example as well as a true PET-MR result, and further more show how to obtain the desired information from the MR images.

1 Introduction

Low signal to noise ratio and the low spatial resolution in functional Positron Emission Tomography (PET) images is a major limitation in brain imaging. Therefore it is desirable to use apriori knowledge about the brain to address this. We present a method using a Bayesian approach to include new structural apriori information from Magnetic Resonance (MR) images. This is possible because the tissue types in the brain, white matter, gray matter and cerebrospinal fluid (CSF), have different activations levels, in average gray matter is 4 times higher than white and CSF has virtually no activation. Using this information extracted from the MR images from the same person gives apriori information about activation edges in the PET image [Chen,84].

Magnetic Resonance Imaging is a technique using changes in the magnetic field to measure structural information, because the different tissue types have different magnetic responses. Using different input response functions it is possible to enhance specific tissue types. No functional information is obtained with MR.

Positron Emission Tomography (PET) imaging is based on radioactive tracers injected in the persons blood. The tracer can e.g. be O-15 and a part of water H_2O. The PET image gives a picture of the blood flow in the brain, which is correlated with the brain activity. In PET imaging there is an intrinsic blurring, because the positron diffuses in average up to 4 mm before annihilating with an electron.

Reconstruction of imagery of a non-ideal imaging system is a fundamental aim of computer vision. Geman and Geman [Geman and Geman,84] introduced Metropolis sampling from Gibbs distributions as a simulation tool for visual reconstruction and showed that a *Simulated Annealing* strategy could improve the efficiency of the sampling process. The sampling process implements a stochastic neural network with symmetric connections. Peterson and Anderson applied the *Mean Field* approximation, and observed substantial improvements in speed and performance [Peterson,87].

In the next section the Bayesian approach to reconstruction is outlined, as model example we study a *Markov Random Field* model.

Section three contains experiments and section four concluding remarks.

2 Bayesian Visual Reconstruction

The basic idea is to consider both the source signal and the processes of the imaging system as stochastic processes. The Bayes formula can then be used to obtain the distribution $P(\mathbf{V}|\mathbf{d})$ of the reconstructed signal $\mathbf{V}$, conditioned on the observed degraded signal $\mathbf{d}$:

$$P(\mathbf{V}|\mathbf{d}) = \frac{P(\mathbf{d}|\mathbf{V})P(\mathbf{V})}{P(\mathbf{d})} \tag{1}$$

This conditional distribution is the product of the distribution of the imaging system process, also called the observation model: $P(\mathbf{d}|\mathbf{V}) \equiv P(\mathbf{V} \rightarrow \mathbf{d})$, and the *prior* distribution of the reconstructed signal $P(\mathbf{V})$. $P(\mathbf{V}|\mathbf{d})$ of Equation (1) is referred to as the *posterior* distribution. Under a given observation, $\mathbf{d}$, $P(\mathbf{d})$ is constant and the posterior distribution is directly proportional to $P(\mathbf{d}|\mathbf{V})$ and $P(\mathbf{V})$.

A useful estimate of the reconstructed signal is given by the *Maximum A Posterior* estimate. Several strategies like simulated annealing and Monte Carlo simulating can be used [Blake,89]. We use a mean field approximation to the simulated annealing, to obtain a more efficient algorithm.

2.1 Markov Random Field Model

In this section a description of the prior distribution is given. The prior model is a Markov Random Field, which is a Gibbs-Markov distribution. The signal is indexed V_i where $i = 0, .., I-1$. Introducing edges p_i, q_i and r_i connected to edges in the x, y and z direction respectively. The edge element p_i have the label {Edge} if the signal elements V_i and V_{i+dx} are independent and similar for q_i in the y direction between V_i and V_{i+dy}, and r_i in z-direction. A 2D part of the model can have the following outline.

$V_{i-dx+dy}$	$p_{i-dx+dy}$	V_{i+dy}	p_{i+dy}	$V_{i+dx+dy}$
q_{i-dx}		q_i		q_{i+dx}
V_{i-dx}	p_{i-dx}	V_i	p_i	V_{i+dx}
$q_{i-dx-dy}$		q_{i-dy}		$q_{i+dx-dy}$
$V_{i-dx-dy}$	$p_{i-dx-dy}$	V_{i-dy}	p_{i-dy}	$V_{i+dx-dy}$

The binary labels for the edges is {Edge, Non-Edge}, implemented as $\{+1, -1\}$.

For simplicity the prior model is first derived without dependence between the edges. In the entire model a simple smoothness between the neighbouring intensity values is used, unless an edge-element is active and allows a discontinuity. In this paper text a simple second order smoothness between neighbour voxels is used.

The prior MRF distribution can be more or less complicated depending on the complexity of the observed model [Hertz,91, Chen,84].

The Gibbs distribution has the following form:

$$P(\mathbf{V}, \mathbf{p}, \mathbf{q}, \mathbf{r} \mid \theta) = Z_1^{-1} \exp\left(-E_{\text{prior}}(\mathbf{V}, \mathbf{p}, \mathbf{q}, \mathbf{r}|\theta)\right), \tag{2}$$

where $E_{\text{prior}}(\mathbf{V}, \mathbf{p}, \mathbf{q}, \mathbf{r}|\theta)$ is the energy or cost function for the MRF model, θ are the model parameters and Z_1^{-1} is a normalisation constant.

In 3D the prior energy function has the following form

$$\begin{aligned} E_{\text{prior}}(\mathbf{V}, \mathbf{p}, \mathbf{q}, \mathbf{r} \mid \boldsymbol{\theta}) &= \frac{\nu}{2}\sum_i (1-p_i)(V_i - V_{i+dx})^2 + \frac{\nu}{2}\sum_i (1-q_i)(V_i - V_{i+dy})^2 \\ &+ \frac{\nu}{2}\sum_i (1-r_i)(V_i - V_{i+dz})^2 + \mu\sum_i p_i + \mu\sum_i q_i + \mu\sum_i r_i \end{aligned} \tag{3}$$

where the model parameters, $\boldsymbol{\theta} = \{\sigma^2, \nu, \mu\}$, all are positive. The last 3 terms are called chemical potentials and are incorporated to control the number of active edges. In this case the value of edge elements p_i , q_i and r_i are set using information from the MR image.

2.2 Observation model

The observation model: $P(\mathbf{d}|\mathbf{V}) \equiv P(\mathbf{V} \to \mathbf{d})$ in (1) is the model describing how the observed data is created from $\mathbf{V}$. In general a non-linear transformation of $\mathbf{V}$ is examined, called $\vartheta(\cdot)$ and describing the mapping $\mathcal{R}^I \xrightarrow{\vartheta} \mathcal{R}^J$.

The distribution may be calculated as

$$p(\mathbf{d}|\mathbf{V}) = p_{noise}(\mathbf{d} - \vartheta(\mathbf{V})) \tag{4}$$

where $p_{noise}(\mathbf{e})$ is the noise distribution of the residual $\mathbf{e} = \mathbf{d} - \vartheta(\mathbf{V})$.

In this paper we only consider a linear transformation $\varphi(\cdot)$ and additive spatially independent Gaussian noise.

We look at the observations model for each point d_j of $\mathbf{d}$.

$$d_j = \varphi_j(\mathbf{V}) + e_j \tag{5}$$

where $\varphi_j(\mathbf{V}) = \sum_i \varphi_{ji} V_i$ and $e_j \in \mathcal{N}(0, \sigma_j^2)$ are independent for each d_j.

Inserting in (4) gives

$$\begin{aligned} p(d_j|\mathbf{V}) &= p_{noise}(\delta(d_j - \varphi_j(\mathbf{V}))) && (6) \\ &= \frac{1}{\sqrt{2\pi\sigma_j^2}} \exp\left(-\frac{(d_j - \varphi_j(\mathbf{V}))^2}{2\sigma_j^2}\right) && (7) \end{aligned}$$

The $p(d_j|\mathbf{V})$ are independent and therefore

$$p(\mathbf{d}|\mathbf{V}) = \prod_j p(d_j|\mathbf{V}) \tag{8}$$

$$= Z_2 \exp\left(-\sum_j \frac{(d_j - \varphi_j(\mathbf{V}))^2}{2\sigma_j^2}\right) \tag{9}$$

$$= Z_2 \exp\left(-E_{noise}(\mathbf{d} \mid \mathbf{V}, \boldsymbol{\theta})\right) \tag{10}$$

where $Z_2 = \prod_j \frac{1}{\sqrt{2\pi\sigma_j^2}}$ and $E_{noise}(\mathbf{d} \mid \mathbf{V}, \boldsymbol{\theta}) = \sum_j \frac{(d_j - \varphi_j(\mathbf{V}))^2}{2\sigma_j^2}$

2.3 3D system model

The system model in 3D then have the following form:

$$P(\mathbf{V} \mid \mathbf{d}, \mathbf{p}, \mathbf{q}, \mathbf{r}, \boldsymbol{\theta}) \propto \exp\left(-E_{\text{prior}}(\mathbf{V}, \mathbf{p}, \mathbf{q}, \mathbf{r} \mid \boldsymbol{\theta})\right) \cdot \exp\left(-E_{\text{noise}}(\mathbf{d} \mid \mathbf{V}, \boldsymbol{\theta})\right) \tag{11}$$

$$= \exp\left(-E_{\text{total}}(\mathbf{V}, \mathbf{d}, \mathbf{p}, \mathbf{q}, \mathbf{r} \mid \boldsymbol{\theta})\right) \tag{12}$$

where $E_{\text{total}}(\mathbf{V}, \mathbf{d}, \mathbf{p}, \mathbf{q}, \mathbf{r} \mid \boldsymbol{\theta}) = E_{\text{prior}}(\mathbf{V}, \mathbf{p}, \mathbf{q}, \mathbf{r} \mid \boldsymbol{\theta}) + E_{\text{noise}}(\mathbf{V} \mid \mathbf{d}, \boldsymbol{\theta})$ is the energy or cost function for the system model and $\boldsymbol{\theta} = \{\sigma^2, \nu, \mu\}$ are the model parameters, all positive.

In this case the value of edge elements p_i, q_i and r_i are set using information from the MR image.

Since the edge elements are fixed based on the MR scan, only an updating formula for $\mathbf{V}$ is needed. The MAP estimate is where E_{total} have its minimum. A simplified description of E_{total} gives an easier derivation.

$$E_{total} = \mathbf{V}^T \boldsymbol{\Sigma}_{\mathbf{MRF}} \mathbf{V} + E_{MRF-edge-control} + (\mathbf{d} - \mathbf{AV})^T \boldsymbol{\Sigma}_{\mathbf{gauss}} (\mathbf{d} - \mathbf{AV}) \tag{13}$$

where $E_{MRF-edge-control} = \sum_i \mu_i^p p_i + \sum_i \mu_i^q q_i + \sum_i \mu_i^r r_i$ and $\boldsymbol{\Sigma}_{\mathbf{MRF}}$ describes the MRF model, and the linear mapping $\varphi(\mathbf{V}) = \mathbf{AV}$. If each voxel is observed with the same noise level, then $\boldsymbol{\Sigma}_{\mathbf{gauss}} = \frac{1}{\sigma^2}\mathbf{I}$.

For the minimum of E_{total} the derivative $\frac{\partial E_{total}}{\partial V_i} = 0$ for all values of i. Using a mean field approach, and assuming that $V_{i' \neq i}$ is independent of V_i during update of V_i. Isolation of V_i gives.

$$V_i = k^{-1}\left[\mathbf{1}_i^T \frac{1}{2\sigma^2} \mathbf{A}^T(\mathbf{d} - \mathbf{AV}) + kV_i - \mathbf{1}_i^T \boldsymbol{\Sigma}_{\mathbf{MRF}} \mathbf{V}\right] \tag{14}$$

$$\text{where } k = \mathbf{1}_i^T \left(\boldsymbol{\Sigma}_{\mathbf{MRF}} + \frac{1}{2\sigma^2}\mathbf{A}^T\mathbf{A}\right)\mathbf{1}_i \tag{15}$$

and $\mathbf{1}_i$ is a zero vector with only element i set to 1. In (14) the right side is independent of V_i. Assuming that the linear transformation is a Gaussian blurring $\mathbf{A}^T = \mathbf{A}$ the updating scheme can be formulated as a double Gauss filtering.

3 Experiments

In this section a few examples using the restoration technique are described. And some of the considerations needed to extract desired edge information. First a small toy example.

3.1 A small example.

In this example a $6 \times 6 \times 6$ cube with intensity value of 100 on a $16 \times 16 \times 16$ background, is blurred with a Gaussian kernel and Gaussian noise is added. The blurring kernel is symmetric with standard deviation 1.5 and the standard deviation of the noise is 5. A slice of the observed signal is in figure (1), and the corresponding restored slice is in figure (2). The edges are fixed on the true edges. Profiles through the centre of the cube can be seen in figure (3).

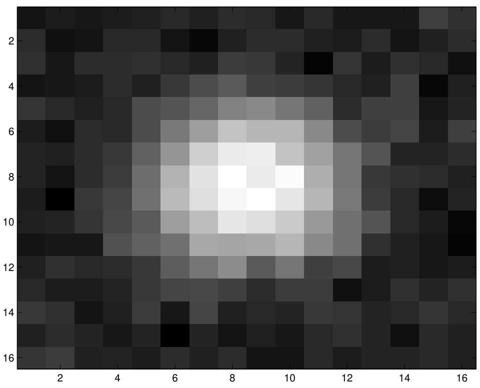

Figure 1: Slice of observed cube, blurred and noise are added.

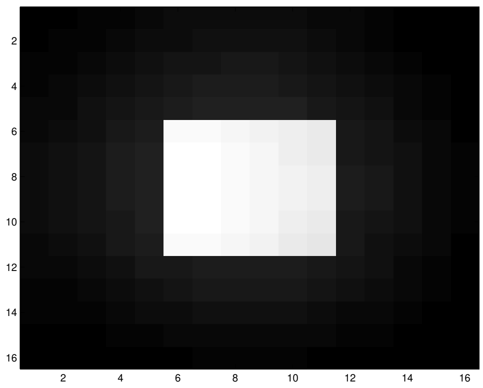

Figure 2: Slice of the restored image.

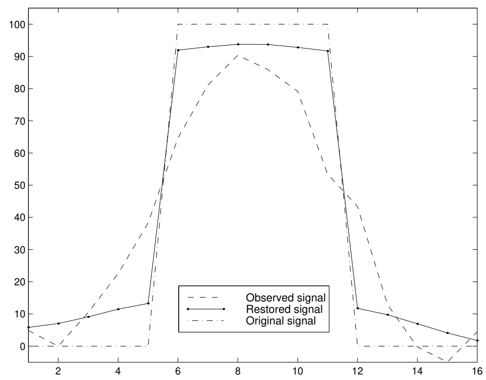

Figure 3: Profiles through the centre of the observed and restored image for the toy example.

3.2 Pre-processing MR Images

To obtain the desired edge information from the raw MR images shown in figure(4), several steps are needed. Focusing on edges between different tissue types based the on knowledge that tissue types in the brain have different activations levels. To obtain a tissue segmented brain several steps are necessary.

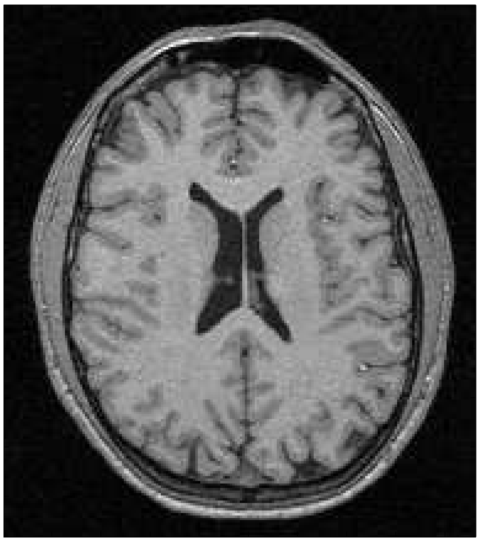

Figure 4: Slice of the Raw MR image, through centre of the brain.

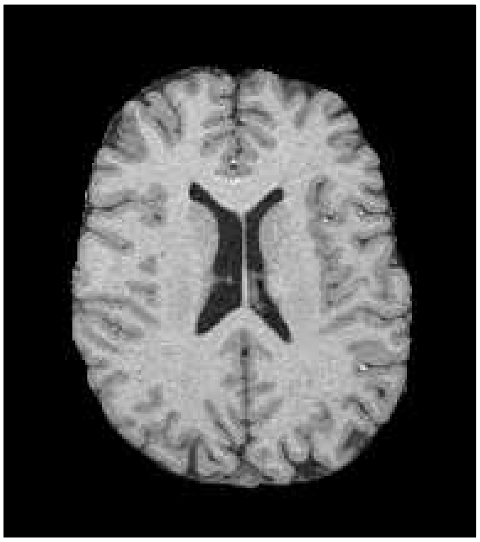

Figure 5: The same slice only containing brain voxels.

First a segmentation in brain/non-brain is done as shown in figure(5). This is needed to obtain a better alignment between PET and MR and to improve segmentation/classification of the brain. Alignment is done using AIR, [Woods,93], which is a 6 parameter transformation determining translation and rotation.

Next step is a segmentation of the brain to obtain volumes of the different tissue types. In this example we segment into CSF, white matter, cortex gray matter and subcortical gray matter, figure(6), using a Gaussian classification. To improve classification gray matter is separated into 2 types, but in the following these are merged again, because we have no information about difference in PET activation in these 2 types of gray matter.

From the segmented MR image a set of edge volumes $(\mathbf{p}, \mathbf{q}, \mathbf{r})$ are created based only on gray matter and white matter, figure(7).

3.3 Restoring PET image.

In figure (8) a slice of the resampled PET image is shown. The PET image is smoothed during the alignment and the resampling to MR resolution.

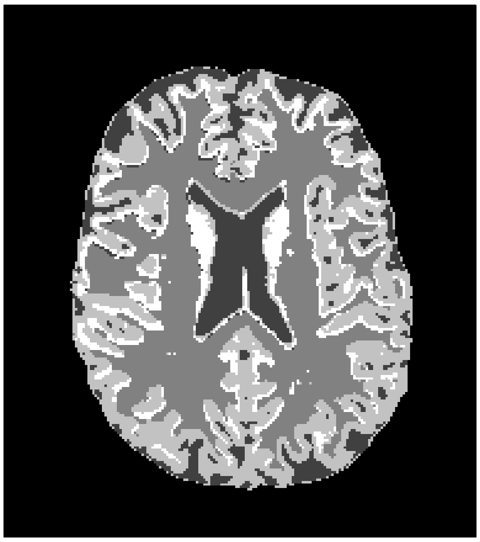

Figure 6: Slice of the segmented MR image, gray matter are the two brightest, then white matter and CSF.

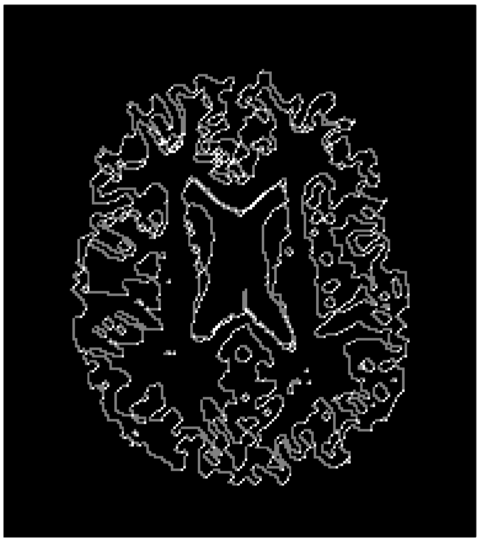

Figure 7: Slice of edges created from MR image, ignoring CSF and merging the two types of gray matter.

Figure 8: Slice of the original PET image, smooth due to resampling to MR resolution.

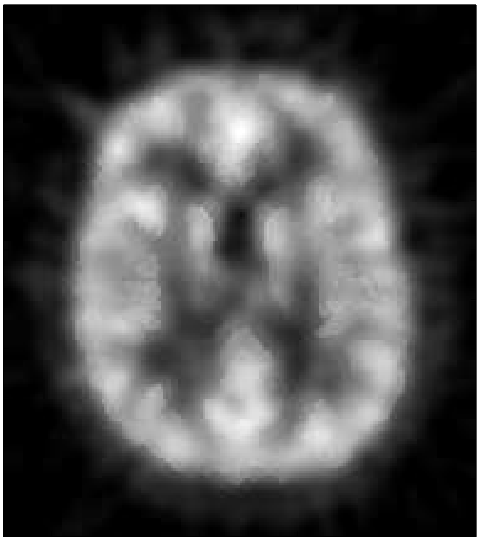

Figure 9: Slice of Restored PET image with a Gauss blurring on 3 mm and a noise level on 2.

A result of this strategy can be seen in figure (9). The Gaussian blurring was set to 3 mm with a noise level on 2. Running 100 iterations with an image size of $190 \times 215 \times 176$ voxels uses 220 Mb ram on a Silicon Graphics ONYX and less than 1 cpu hour. It was possible to preserve/enhance edges and avoid edge blurring on the outside of the brain.

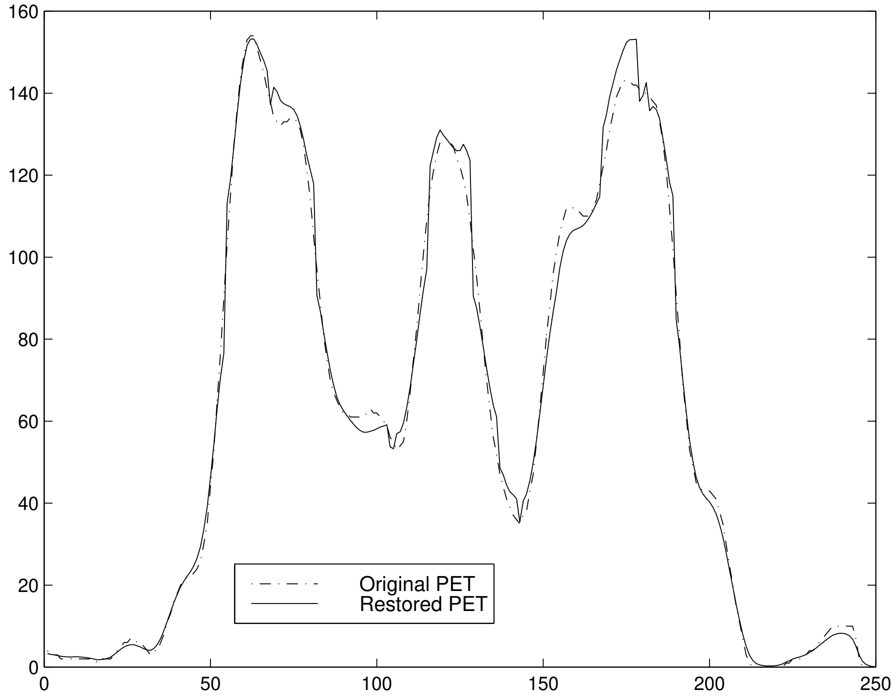

Figure 10: Profile through pet scans.

4 Concluding remarks

In this article we presented a method to enhance the resolution in functional positron emission tomography (PET) images, using anatomical information of the brain from MR images of a higher resolution by using a Markov Random Field (MRF) model which have the possibility of incorporating this information. This gives a new possibility of smoothing the very noisy PET images without blurring edges, and at the same time recovering the intrinsic blurring. In this scheme we do not preserve edges within tissue types, but an extension of the prior model can overcome this problem.

References

[Blake,89] A. Blake: *Comparison of the Efficiency of Deterministic and Stochastic Algorithms for Visual Reconstruction.* IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-11** 2-12 (1989).

[Chen,84] X. Ouyang, W.H. Wong, V.E. Johnson, X. Hu, C.-T. Chen: *Incorporation of Correlated Structural Images in PET Image Reconstruction.* IEEE Transactions on Medical Imaging **13** 627-640 (1994).

[Geman and Geman,84] D. Geman and S. Geman: *Stochastic Relaxation, Gibbs distributions and the Bayesian restoration of images.* IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-6** 721-741 (1984).

[Hertz,91] J. Hertz, A. Krogh and R.G. Palmer: *Introduction to the Theory of Neural Computation.* Addison Wesley, New York (1991).

[Peterson,87] C. Peterson and J.R. Anderson: *A Mean Field Learning Algorithm for Neural Networks.* Complex Systems **1** 995-1019, (1987).

[Roth,90] M.W. Roth: *Survey of Neural Network Technology for Automatic Target Recognition.* IEEE Transactions on Neural Networks **1**, 28-43, (1990).

[Woods,93] R. P. Woods and S. R. Cherry: *MRI-PET registration with automated algorithm.* Journal of Computer Assisted Tomography, 17(4):536-546, (1993).

Bibliography

[Carson and Lange, 1985] Carson, R. E. and Lange, K. (1985). The em parametric image reconstruction algorithm. *Journal of the American Statistical Association*, 80(389):20–5.

[Censor, 1983] Censor, Y. (1983). Finite series-expansion reconstruction methods. In *Proceedings of the IEEE*, volume 71, pages 409–19.

[Chan and et al., 1996] Chan, B. and et al., M. B. (1996). Scatter distribution in transmission measurements with positron emission tomograpy. *Journal of Computer Assisted Tomography*, 10(2):296–301.

[Chen et al., 1990] Chen, C.-T., Johnson, V. E., Wong, W. H., Hu, X., and Metz, C. E. (1990). Bayesian image reconstruction in poistron emission tomography. *IEEE transactions on Nuclear Science*, 37(2).

[Cherry et al., 1991] Cherry, S. R., Dahlbom, M., and Hoffman, E. J. (1991). 3d pet using a conventional multislice tomograph without septa. *Journal of Computer Assisted Tomography*, 15(4):655–68.

[Cho et al., 1993] Cho, Z.-H., Jones, J. P., and Singh, M. (1993). *Foundations of Medical Imaging*. John Wiley & Sons, Inc.

[Clack, 1992] Clack, R. (1992). Towards a complete description of three-dimensional filtered backprojection. *Physics in Medicine & Biology*, 37(3):645–60.

[Cohen and Cohen, 1993] Cohen, L. and Cohen, I. (1993). Finite element methods for active contour models and balloons for 2-d and 3-d images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI(15):1133–1147.

[Cootes et al., 1994] Cootes, T. F., Hill, A., Taylor, C. J., and Haslam, J. (1994). The Use of Actice Shape Models For Locating Structures in Medical Images. *Image and Vision Computing*, 12(6):355–366.

[Deans, 1983] Deans, S. R. (1983). *The Radon Transform and Some of Its Applications*. Krieger Publishing Company, Malabar, Florida. Previously John Wiley & Sons Inc., 1983.

[Defrise et al., 1989] Defrise, M., Townsend, D. W., and Clack, R. (1989). Three-dimensional image reconstruction from complete projections. *Physics in Medicine & Biology*, 34(5):573–87.

[DeGrado and et. al, 1994] DeGrado, T. R. and et. al (1994). Performance characteristics of a whole-body pet scanner. *The Journal of Nuclear Medicine*, 35:1398–406.

[Dudgeon and Merserau, 1984] Dudgeon, D. E. and Merserau, R. M. (1984). *Multidimensional Digital Signal Processing*. Prentice-Hall.

[Edholm and G. T. Herman, 1987] Edholm and G. T. Herman, P. R. (1987). Linograms in image reconstruction from projections. *IEEE transactions on Medical Imaging*, 6(4):301–7.

[Freifelder and et al., 1994] Freifelder, R. and et al., J. S. K. (1994). Design and preformance of the head penn-pet scanner. *IEEE conference records, Nuclear Science & Medical Imaging*.

[Geman and Geman, 1984] Geman, D. and Geman, S. (1984). Stochastic Relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6:721–741.

[Gindi et al., 1995] Gindi, G., Lee, M., Rangarajan, A., and Zubal, I. G. (1995). Bayesian reconstruction of functional images using anatomical information as priors. *IEEE, Transactions on Medical Imaging*, 12(4).

[Guan and Gordon, 1994] Guan, H. and Gordon, R. (1994). A projection access order for speedy convergence of art (algebraic reconstruction technique): a multilevel scheme for computed tomography. *Physics in Medicine & Biology*, 39:2005–22.

[Guedon et al., 1992] Guedon, J.-P., Unser, M., and Bizais, Y. (1992). Pixel intensity distribution models for filtered back-projection. *IEEE Transaction*.

[Gunn and Nixon, 1997] Gunn, S. R. and Nixon, M. S. (1997). A robust snake implementation; a dual active contour. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(1):63–68.

[Hansen et al., 1996] Hansen, L. K., Andersen, L. N., Kjems, U., and Larsen, J. (1996). Revisiting boltzmann learning: Parameter estimation in markov random fields. In *Proceedings of International Conference on Acoustics Speech and Signal Processin*, volume 6.

[Herman, 1980] Herman, G. T. (1980). *Image Reconstruction from Projections: The fundamentals of Computerized Tomography*. Academic Press.

[Herman and Meyer, 1993] Herman, G. T. and Meyer, L. B. (1993). Algebraic reconstruction techniques can be made computationally efficient. *IEEE trans. on Medical Imaging*, 12(3):600–9.

[Hertz et al., 1991] Hertz, J., Krogh, A., and Palmer, R. (1991). *Introduction to the Theory of Neural Computation*. Addison Wesley, New York.

[Hertz, 1989] Hertz, J. A. (1989). A gauge theory in computational vision: A model for outline extraction. *Physica Scripta*, 39:161–167.

[Hill et al., 1993] Hill, A., Thornham, A., and Taylor, C. J. (1993). Model–based interpretation of 3d medical images. In *4th British Machine Vision Conference, 1993, Guildford, England*, pages 339–348.

[Hopfield and Tank, 1985] Hopfield, J. and Tank, D. W. (1985). Neural computation of decisions optimization problems. *Biological Cypernetics*, 52.

[Jain, 1989] Jain, A. K. (1989). *Fundamentals of Digital Image Processing.* Prentice Hall.

[Jeng et al., 1993] Jeng, F.-C., Woods, J. W., and Rastogi, S. (1993). Compund Gauss-Markov Random Fields for Parallel Image Processing. In Chellappa, R. and Jain, A., editors, *Markov Random Fields,Theory and Application*, chapter 2. Academic Press inc.

[Jensen and Philipsen, 1995] Jensen, J. J. and Philipsen, P. A. (1995). Direkte metoder til rekonstruktion af pet billeder. Master's thesis, Electronic Institute, Technical University of Denmark.

[Jernigan et al.,] Jernigan, T., Archibal, S., Foster, D., Fennema-Notestine, C., Sellers, J., Symonds, L., Harris-Collazo, M., Philipsen, P., Stout, J., Ryan, L., Corey-Bloom, J., and Hesselink, J. Anatomically-segmented structural magnetic resonance image volumes: Reliability of methods and availability of images. *NeuroImage, In review.*

[Johnson et al., 1991] Johnson, V. E., Wong, W. H., Hu, X., and Chen, C.-T. (1991). Image restoration using gibbs priors: Boundary modeling, treatment of blurring, and selection of hyperparameters. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 13(5):413–425.

[Kaufmann, 1987] Kaufmann, L. (1987). Implementing and accelerating the em algorithm for positron emission tomograhy. *IEEE Transactions on Medical Imaging*, MI-6(1):37–51.

[Kinahan and Rogers, 1989] Kinahan, P. E. and Rogers, J. G. (1989). Analytic 3d image reconstruction using all detected events. *IEEE Transactions on Nuclear Science*, 36(1):964–8.

[Kjems, 1998] Kjems, U. (1998). *Bayesian Signal Porcessing and Interpretation of Brain Scans.* PhD thesis, Department of Mathematical Modelling, Technical University of Denmark.

[Lewitt, 1983] Lewitt, R. M. (1983). Reconstruction algorithms: Transform methods. *Proceedings of the IEEE*, 71(3):390–408.

[Li, 1995] Li, S. Z. (1995). *Markov Random Fields in Computer Vision.* Springer-Verlag, Tokoy.

[Lipinski et al., 1997] Lipinski, B., Herzog, H., Kops, E. R., Oberschelp, W., and Müller-Gärtner, H. W. (1997). Expectation maximization reconstruction of positron emission tomography images using anatomical magnetic resonance information. *IEEE transaction on Medical Imaging*, 16(2).

[M. Defrise and Geissbuhler, 1990] M. Defrise, D. T. and Geissbuhler, A. (1990). Implementation of three-dimensional image reconstruction for multi-ring positron tomographs. *Phys. Med. Biol*, 35(10):1361–1372.

[Magnusson, 1989] Magnusson, M. (1989). *The Linogram Method for Image Reconstruction from Projections.* PhD thesis, Linköbings tekniska högskola, Dep. of Electrical Engineering, S-581 83 Linköbing, Sweden. Revised version 1993.

[Mersereau, 1973] Mersereau, R. M. (1973). Recovering multidimensional signals from their projections. *Computer Graphics and Image Processing*, 1(4):179–95.

[Metropolis et al., 1953] Metropolis, N., Rosenbluth, A. W., Rosenblutha, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092.

[Mumcuoglu et al., 1996] Mumcuoglu, E. U., Leahy, R. M., and Cherry, S. R. (1996). Bayesian reconstruction of pet images: methodology and performance analysis. *Phys. Med. Biol.*, 41:1777–1807.

[Nadabar and Jain, 1996] Nadabar, S. G. and Jain, A. K. (1996). Parameter Estimation in Markov Random Field Contextual Models Using Geometric Models of Objects. *IEEE Transaction on PAMI*, 18(3).

[Ohlsson et al., 1997] Ohlsson, M., Toft, P., Hansen, L. K., and Nielsen, F. (1997). Active surface models for brain imaging. In *Proceedings of the Interdisciplinary Inversion Workshop 5*. Department of Earth Sciences, Aarhus University, ISBN 87-90400-12-7.

[Orlov, 1975a] Orlov, S. S. (1975a). Theory of three dimensional reconstruction. i. conditions for a complete set of projections. *Sov. Phus. Crystallogr.*, 20(3):312–314.

[Orlov, 1975b] Orlov, S. S. (1975b). Theory of three dimensional reconstruction. ii. the recovery operator. *Sov. Phus. Crystallogr.*, 20(4):429–433.

[O'Sullivan et al., 1993] O'Sullivan, F., Pawitan, Y., and Haynor, D. (1993). Reducing negativity artifacts in emision tomography: Post-processing filtered backprojection solutions. *IEEE Transactions on Medical Imaging*, 12(4).

[Ouyang et al., 1994] Ouyang, X., Wong, W. H., Johnson, V. E., Hu, X., and Chen, C.-T. (1994). Incorporation of correlated structual images in pet image reconstruction. *IEEE Transaction of Medical Imaging*, 13(4).

[Peterson and Söderberg, 1989] Peterson, C. and Söderberg, B. (1989). A new method for mapping optimization problems onto neural networks. *International Journal of Neural Systems 1*.

[Philipsen et al., 1996] Philipsen, P. A., Hansen, L. K., and Toft, P. (1996). Mean field reconstruction with snaky edge hints. In *Inverse Methods, Interdisciplinary Elements of Methodology, Computation, and Applications*, number 63 in Lecture Notes in Earth Sciences, pages 312–319. Springer.

[Philipsen et al., 1997] Philipsen, P. A., Kjems, U., Toft, P., and Hansen, L. K. (1997). Restoring functional pet images using anatomical mr images. In *Porceedings of the Interdisciplinary Inversion Workshop 5*. Department of Earth Sciences, Aarhus University, ISBN 87-90400-12-7.

[Pierro, 1992] Pierro, A. R. D. (1992). Multiplicative iterative methods in computed tomography. In *Mathematical Methods in Tomography, Proceedings, Oberwolfach 1990*, number 1497 in Lecture Notes in Mathematics, pages 167–186. Springer-Verlag.

[Radon, 1917] Radon, J. (1917). Über die bestimmung von funktionen durch ihre integralwerte längs gewisser mannigfaltigkeiten. *Ber. Ver. Sächs. Akad. Wiss. Leipzig, Math-Phys. Kl.*, 69:262–77. In German. English translation in S. R. Deans: The Radon Transform and Some of Its Applications.

[Rao, 1973] Rao, C. R. (1973). *Linear Statistical Inference and Its Applications.* John Wiley & Sons, second edition.

[Rogers et al., 1987] Rogers, J. G., Harrop, R., and Kinahan, P. E. (1987). The theory of three-dimensional image reconstruction for pet. *IEEE Transactions on Medical Imaging*, MI-6(3):239–43.

[Schalkoff, 1989] Schalkoff, R. J. (1989). *Digital Image Processing and Computer Vision.* John Wiley & Sons.

[Schorr et al., 1983] Schorr, B., Townsend, D., and Clack, R. (1983). A general method for three-dimensional filter computation. *Phys. Med. Biol.*, 28(9):1009–1019.

[Shepp and Krustal, 1978] Shepp, L. A. and Krustal, J. B. (1978). Computerized tomography: The new medical x-ray technology. *American Mathematics Monthly*, 85:420–39.

[Storvik, 1994] Storvik, G. (1994). A baysian approach to dynamic contours through stochastic sampling and simulated annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(10):976–986.

[Toft, 1996] Toft, P. (1996). *The Radon Transform, Theory and Implementation.* PhD thesis, Department of Mathematical Modelling, Section for Digital Signal Processing, Technical University of Denmark.

[Toft et al., 1996] Toft, P., Philipsen, P. A., Hansen, L. K., and Kjems, U. (1996). Brain mapping by positron emission tomography. In *Porceedings of the Interdisciplinary Inversion Workshop 4*. IMM, DTU, ISBN 87-88306-24-0.

[Townsend and T. Spinks, 1989] Townsend, D. W. and T. Spinks, T. J. e. a. (1989). Three dimensional reconstruction of pet data from a multi–ring camera. *IEEE Transactions on Nuclear Science*, 36(1):1056–65.

[Trussell et al., 1987] Trussell, H. J., Orun-Ozturk, H., and Civanlar, M. (1987). Errors in reprojection methods in computerized tomography. *IEEE Transactions on Medical Imaging*, MI-6(3):220–7.

[Vardi et al., 1985] Vardi, Y., Shepp, A., and Kaufmann, L. (1985). A statistical model for positron emission tomography. *Journal of the American Statistical Association*, 80(389):8–37.

[Woods and Cherry, 1993] Woods, R. P. and Cherry, S. R. (1993). Mri-pet registration with automated algorithm. *Journal of Computer Assisted Tomography*, 17(4):536–546.

[Yezzi et al., 1997] Yezzi, A., Kichenassamy, S., Kumar, A., Olver, P., and Tannenbaum, A. (1997). A geometric snake model for segmentation of medical imagery. *Transactions on Medical Imaging*, 16(2):199–209.

[Zerubia and Chellappa, 1993] Zerubia, J. and Chellappa, R. (1993). Mean field annealing using compound gauss-markov random fields for edge detection and image estimation. *IEEE Transactions on neural networks*, 4(4):703–709.

[Zhou et al., 1994] Zhou, Z., Leahy, R. M., and Mumcuoglu, E. U. (1994). A comparative study of the effect of using anatomical priors in pet reconstruction. *IEEE*.

[Zhu and Yuille, 1996] Zhu, S. C. and Yuille, A. (1996). Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):884–900.