Bias Field Correction in Magnetic Resonance Images of a Rat Brain

Xavier Peña Piña

Assistants : Dr. Meritxell Bach Cuadra, Dr. Nathalie Just
Director : Prof. Dr. Jean-Philippe Thiran, Prof. Dr. Rolf Gruetter

Signal Processing Institute (ITS),
Swiss Federal Institute of Technology (EPFL)

October 15, 2007

Abstract

Magnetic Resonance Imaging (MRI) is nowadays a widely used medical tool, as it is a non-invasive and non-harmful way to study inner soft tissues. One of the characteristics of this method is the bias field, also called Intensity Inhomogeneity (IIH), which is an artifact that affects quantitative image analysis consisting in a low frequency variation of the brightness through all the image acquired. This undesired effect makes difficult medical functions such as visual inspection or also intensity-based segmentation [1]. The bias field is a deterministic function tied to complex physical interactions between the magnetic and electric fields and the living tissues, and this is why in this paper this bias field is directly corrected from the corrupt image data, using all the *a priori* information at our disposal. We will specifically work on surface-coil images acquired at $9.4T$, where the inhomogeneity effect is even stronger [2].

Acknowledgments

This work was carried out between March 2007 and August 2007 at the Signal Processing Institute of the Swiss Federal Institute of Technology in Lausanne, Switzerland, in collaboration with the CIBM/LIFMET.

First of all, I would like to thank the director of this Master thesis, Professor Jean-Philippe Thiran, for giving me the possibility to work at LTS and for the valuable advices he gave me throughout this work. I would like to thank also my supervisor Doctor Meritxell Bach Cuadra for her availability, good mood and patience during these six months. I am also really grateful to my supervisor Doctor Nathalie Just for encouraging me so much with her opinions about this work and for helping me to understand the complex world of MRI, as well as to Professor Rolf Gruetter for sharing with me his extensive knowledge about this matter. I must also thank Professor Ferran Marquès from ETSETB, Barcelona, for encouraging me to go to Lausanne, as well as for how much I have learnt from him since I began my university studies.

My warmest thanks to my mother and my father, for their unconditional support which is so important to me. Also many thanks to the other members of my family, specially to my godson Pere for making me smile so much, and a warm welcome to this world to his little brother Pau. Thanks to my grandparents German, Aurelia, Joan and Maria, my godfathers Joan and Marta, aunts and uncles Jaume, Gaspar, Josep and Sari, as well as Anita, Francisca and Isaías, to my cousins Maria, Mercè, Toño, Pablo, Isaías, Kiko, Jesús, Lucía and Fátima, as well as Juanjo, Joan and Vanessa. A special loving memory for those who are no longer among us. In this paragraph I must also name my friend Pepe, without whose support I would not be here.

Also special thanks to my best friends from Mallorca: Deguer, Nacho, Camilo, Joan (adopted), Sion, Andreu, Artur, Josep, Rito, Jose, Emilio, Pere, Martí, Berto, Pomar, Jordi, Salva, Marc... and also the great people I have met in Barcelona, specially to Rubén for being my team-mate and such a good friend, but also to Hector, Tomeu, Marta, Topo, Christian and many more.

And finally I would like to say thanks to the great people I have known in Lausanne: the ones that paved the way for me and they have already left (specialy Andreu, and also Pablo, Agnès, André, Oscar (a pillar), Miguel, Eva, Ana, Marina, Jesús...), the ones who will remain with me in Lausanne (Oriol, Roser, the always smiling Anna, César *Aluchi*, Luis, Cathrin...), and also to the people in the laboratory (Cecilia, Alia, Azzurra, Luca, Davide, Ale...). Also thanks to Joahnn, Belén, Jose, Paco ("*Compae*"), Annegret, Javi, Luigi, Gilles,

Rui, Koeman, Belletti, Cris, Rusca, and many more.

My most sincere thanks to all of you, and also to all the other ones I missed when wrote this. The merit of this work is also yours.

Contents

Chapter 1

Introduction

1.1 Motivation of This Work

Enginery has always been a support and a source of improvement in medical fields. Since the discovery of the utility provided by magnetic fields to literally look inside a patient without harming him at all, a lot of effort has been put on research in order to improve that versatile and flexible source of diagnosis. The main tendency on this kind of investigation has been to increase the intensity of the magnetic field applied on the patient, which provides a better resolution in the image obtained. Up to now $1.5T$ systems are vastly used in hospitals around the world, and $3T$ scanners are also starting to be used. Meanwhile the scientific community is working with MR scanners of $7T$ adapted to human, and up to $14T$ adapted to small animals.

The aim of this research is to supply the medical community with a strong imaging tool to be combined with medical knowledge in order to improve medical diagnosis and treatment. Therefore, increasing the resolution when observing inner tissues in a patient is something completely desirable, but there are also some problems tied to those high intensity magnetic fields. These problems are still being studied and thus the tool is not yet prepared for medical purposes. One of those problems tied to the increase of the the magnetic field intensity is the increase of the so-called Intensity Inhomogeneity or bias field, which makes difficult intensity-based segmentation (an important feature for sclerosis diagnosis) or even simple visual inspection. In some other cases the bias field is not detectable with the eye only, but it is still affecting quantitative image analysis.

1.2 MRI Acquisition

Behind its high electronic complexity, a MR device consists of only three main hardware components: the main magnet, the magnetic field gradient system and the radio-frequency (RF) system [3]. Combined, they allow to build a device able to reproduce the acquisition principles described in 1.2.1. The whole process can be divided in two parts: the first one is the acquisition process, and the second one is the image reconstruction. The main inhomogeneities are due

to the magnetic fields imperfections and not to the reconstruction process, thus we are not going to get deeper into the algorithmic part of the MR Imaging process but interested reader can refer to [4].

1.2.1 MRI Acquisition Principles

Physically, MRI is based on the nuclear magnetic resonance (NMR) phenomenon [11]. Therefore MR signals used for image formation come directly from the object itself. In this sense, MRI is a form of emission tomography similar to PET and SPECT, but contrary to these devices MRI does not require injection of a radioactive isotope into the object. There are other tomographic techniques based on transmission or diffraction, such as X-ray, CT and acoustic tomography, but what makes the MRI system different is that it minimizes the effects on the object: no radioactive isotopes or rays go through the internal tissues of the patient. Besides, the information provided MR images is extremely rich.

The image pixel value is in general dependant on a host of intrinsic parameters including the nuclear spin density, the spin-lattice relaxation, the spin-spin relaxation time, molecular motion and chemical shift differences. The imaging effects of these parameters can be suppressed or enhanced in a specific experiment by another set of operator-selectable parameters, such as the repetition time, the echo time and also the flip angle. Therefore, an MR image obtained from the same anatomical sample can give as a result different images depending on the data acquisition protocol.

1.2.2 Signal Generation and Detection

All nuclei that contain odd numbers of protons or neutrons have an intrinsic magnetic moment and angular momentum. MR resonant frequencies for a particular substance are directly proportional to the strength of the applied magnetic field. MR studies magnetic nuclei by aligning them with an applied constant magnetic field and perturbing this alignment using an alternating electromagnetic field, the fields being orthogonal. The resulting response to the perturbing electromagnetic field is the phenomenon that is exploited in MR spectroscopy and magnetic resonance imaging, which use very powerful applied magnetic fields in order to achieve high resolution spectra.

An MR system is made by three main components: the main magnet, the gradient system and the radio-frequency system. They all are explained below.

1.2.3 The Main Magnet

The main magnet is the responsible for the main external magnetic field, usually denoted by $\vec{B}_0$. It produces the strong and uniform static field that will align the spins in our object. Its importance is that signal to noise ratio is directly dependent on the strength of this magnetic field: high fields induce better ratios. The problem is that when working at hight values (such as $7T$ or $14T$) the

perfect field uniformity is currently impossible to ensure.

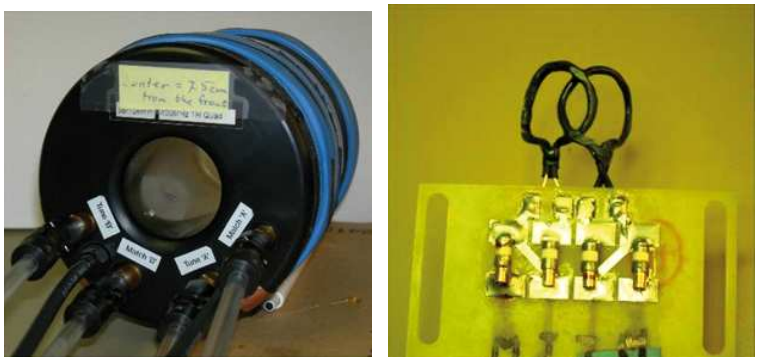
1.2.4 The Gradient System

The magnetic field gradient system consists of three orthogonal gradient coils. They are designed to produce time-varying magnetic fields of controlled spatial non-uniformity. The gradient system is essential for signal localization, thus there are some critical specifications to take into account such as maximum gradient strength and also the rate at which this maximum gradient strength can be obtained. The lower limit of the gradient strength required is determined by a basic criterion: the gradient field must be stronger than the main field inhomogeneity. The time interval for a gradient to ramp up to its full strength is called the *rise time*; the shorter the rise time, the better the gradient system. For conventional imaging methods rise times of approximately $1.0ms$ (rising from 0 to $10mT/m$) are considered to be good enough, but for some fast imaging methods a shorter rise time is needed. Gradient system does not usually introduce any remarkable noise to the image.

1.2.5 The Radio Frequency System

The Radio Frequency (RF) component is a key element in an MR system. It consists of a transmitter coil, which is capable of generating a rotating magnetic field, and a receiver coil which converts the magnetic field into an electric signal. Therefore it is the responsible for the signal acquisition.

There are several types of coils (Fig. 1.1) with different shapes and sizes, each one with different acquisition characteristics. They allow MR systems to capture either a complete image of our object (volume coils), either an image of a specific region (surface coils). All of these RF components try to provide a uniform rotating magnetic field and a high detection sensitivity, but those characteristics vary according to the coil type.



(a) (b)

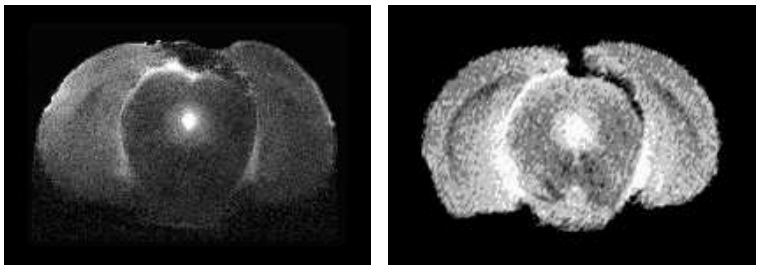
Figure 1.1: Small volume coil (a), and two-loop surface coil (b).

Volume Coils

The radio frequency volume coils completely surround the specific region studied, such as the head or the knee. Volume coils have a better RF homogeneity than surface coils, but they also present a worst detection sensitivity. The most commonly used design is the *birdcage coil*, consisting of wires running along the z-direction and arranged to provide a cosine current variation around the circumference of the coil. It is possible to use the same coil to transmit and to receive, but there is also the possibility to use a separate coil for each function. If two separate coils are used, the body coil works as the transmitter meanwhile a smaller coil is used to receive the signal.

Surface Coils

The simplest surface coil consists of a only loop of wire, and it is very useful for imaging a limited region. But there are many different types of surface coils with different shapes and sizes. Those coils do not have an adapted shape to capture a complete object as the volume coils do, but they require less electrical power than volume coils and so they heat less the the imaged region, as well as they provide a better detection sensitivity. The main problem when using this kind of coils is the strong intensity inhomogeneity they show (Fig. 1.2).



(a) (b)

Figure 1.2: Image acquired with a surface coil (a), and the same rat brain section acquired with a volume coil (b).

1.3 Goals of this Project

Our aim is to reduce intensity inhomogeneities in MRI at high field, in order to increase the performance of other techniques such as the atlas-based segmentation or statistical classification, which are not accurate on a biased medical image.

For that purpose we will follow the work of Rémi Charrier [3], and we will focus on how to best quantify and correct the intensity inhomogeneity. The problem will be studied through an histogram analysis from simulations, as well as testing the impact of the noise.

All the images treated in this report were acquired at $9.4T$ (VARIAN $31cm$ bore magnet) at the CIBM [10] (Fig. 1.3).

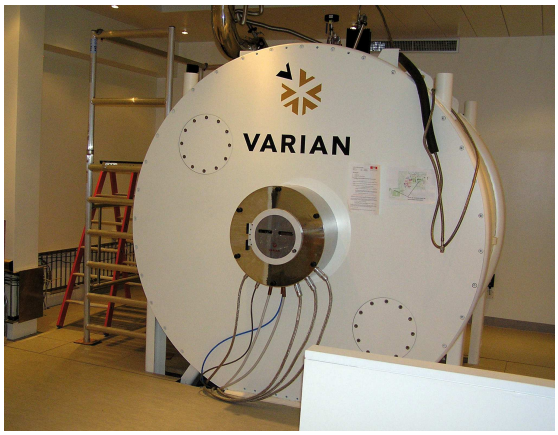


Figure 1.3: $9.4T$ magnet placed in the Centre d'Imagerie Biomedicale (CIBM), Lausanne.

Chapter 2

Analysis of the Bias Field in MRI

2.1 Mathematical Approach

The first step to understand the bias problem is to formulate it in a mathematical framework. This is a key point in understanding the processing used to remove this bias field from the corrupt image. But let us start from the beginning of the process.

The image processing of the MR acquisition begins when all the MR information is stored as $k-$space data. The $k-$space is the temporary space in which data from digitized MR signals are stored. When $k-$space is full (at the end of the scan), the data is mathematically processed with a simple FFT, producing a final image in what we call the $r-$space. Thus $k-$space holds raw data before reconstruction [12] [13], and then this raw data is transformed into the $r-$space or Euclidean space. It is in the latter one in where our processing will take place. In fact the two distributions carry exactly the same information. The transformation [14] from one to the other is

$$w(r) = \int W(k) e^{-2\pi j(kr)} dk. \tag{2.1}$$

The raw data contains the frequency information, and when we apply the Fourier Transform we obtain the spatial information which we will process. The mathematical model [2] of the signal acquired and translated to the $r-$space is the following one:

$$s(\underline{x}) = (o(\underline{x}) + n_{bio}(\underline{x})) * h(\underline{x}) + b(\underline{x}) + n_{MR}(\underline{x}). \tag{2.2}$$

where:

- $s(\underline{x})$ is the measured and digitalized signal
- $\underline{x} = (x, y, z)$, where z moves along the different slices, and (x, y) are the spatial coordinates of each slice (we consider each slice as a single image). The notation x *underlined* means that we are dealing with digital [1] data

[1] That is to say: discrete and quantized data.

- $o(\underline{x})$ is the ideal image
- $n_{bio}(\underline{x})$ is the biological noise due to interior structures of the tissue, physiological noise, cardiac rate, respiratory motion, et cetera
- $n_{MR}(\underline{x})$ is the noise due to the measuring device acquisition
- $h(\underline{x})$ is the model of the blurred border region effect due to discrete sampling of the tissue
- $b(\underline{x})$ is the bias field, considering $0 \leq b(\underline{x}) \leq 1$

We will not take advantage of the complexity of that model, since some of its components are not important for us because we are dealing only with the bias, which is a vast effect over the whole image. The border region effect $h(\underline{x})$ is a tiny and focalized effect over the image, and we can neglect it without changing our estimation of the bias field. That lead us to the following expression:

$$s(\underline{x}) = o(\underline{x}) + n_{bio}(\underline{x}) + b(\underline{x}) + n_{MR}(\underline{x}). \tag{2.3}$$

So now we can mix both biological noise and MR noise in a single term we will call $n(\underline{x})$ [2]:

$$n(\underline{x}) \cong n_{bio}(\underline{x}) + n_{MR}(\underline{x}). \tag{2.4}$$

The final expression is the following one:

$$s'(\underline{x}) \cong o(\underline{x}) + b(\underline{x}) + n(\underline{x}). \tag{2.5}$$

But as in MRI the bias is a multiplicative effect on the image, we must modify the expression to:

$$s''(\underline{x}) = o(\underline{x})b(\underline{x}) + n(\underline{x}) \tag{2.6}$$

Note that as we know that $n(\underline{x}) \ll o(\underline{x}) \cdot b(\underline{x})$ in the major part of the image, then the best way to remove the bias directly from the image is first estimating $b(\underline{x})$ and then dividing the corrupt image $s(\underline{x})$ by that term. In those parts where $n(\underline{x}) \sim o(\underline{x}) \cdot b(\underline{x})$ we will say that the signal is non recoverable. Calling $\hat{b}(\underline{x})$ to our bias estimation, the result when correcting is this one:

$$\hat{o}(\underline{x}) = s(\underline{x})/b(\underline{x}). \tag{2.7}$$

and supposing $\hat{b}(\underline{x}) \cong b(\underline{x})$, then

$$\hat{o}(\underline{x}) \cong o(\underline{x}) + n(\underline{x})/b(\underline{x}). \tag{2.8}$$

Due to that multiplicative effect, it is usual to apply a logarithmic transform in order to change the multiplicative effect of the bias into an additive one. The utility of that transformation will be seen later in Chapter 3.

$$log(\hat{o}(\underline{x})) = log(s(\underline{x})) - log(\hat{b}(\underline{x})) \cong log(o(\underline{x}) + n(\underline{x})/b(\underline{x})). \tag{2.9}$$

[2] Looking at the section of the images below (see figure 2.4) we notice there is no difference between high frequency variations (*id est*: the noise) in zones with no bias and the one in zones with strong bias.

Now we will focus on the noise term of the corrected data set, which we will call $n'(\underline{x}) = n(\underline{x})/b(\underline{x})$. The first factor involved in that expression is $n(\underline{x})$, an aleatory term with zero mean and constant variance all through the image (see explanation below). The second one is $b(\underline{x})$, a deterministic expression that varies smoothly throughout the image, remaining between the values 0 and 1. That means that the variance of the noise $n'(\underline{x})$ in this corrected image will strongly depend on the value of the bias at that point. In other words, noise variance will be higher in those points where this bias factor is closer to zero. This is tied to the fact that the acquired signal presents a non-uniform Signal to Noise Ratio all through the data set, because as the noise factor remains constant our signal intensity depends on the position in the image.

2.2 The Noise

We have seen that the noise is an important factor in our correction, and we are going to take a closer look on that point. We assume noise as an electrical variation modeled as $var(em_{noise}) = 4KT \cdot R \cdot BW$ [15]. We assume that the different components of the system all contribute to the noise independently, so in the $k-$space the total noise in the system is the sum of the noise from the individual sources:

$$\sigma^2_{thermal}(\underline{k}) = \sigma^2_{body}(\underline{k}) + \sigma^2_{coil}(\underline{k}) + \sigma^2_{electronics}(\underline{k}). \tag{2.10}$$

where σ^2 is the variance of the noise. So that when we translate it to the effective resistance of the noise contribution, the result is:

$$R_{eff} = R_{body} + R_{coil} + R_{electronics}. \tag{2.11}$$

The noise associated with the acquired MR signal is generally taken to be additive, uncorrelated and Gaussian with zero (or low value) mean and of comparable variance for each quadrature channel [3]. Our signal can be modeled as:

$$s_m(\underline{k}) = s(\underline{k}) + \epsilon(\underline{k}). \tag{2.12}$$

where $\epsilon(\underline{k})$ contains the sum over body and electronics contributions.

To understand the noise influence in the spatial representation after Fourier transform, and knowing that in the $k-$space the variance of the noise is zero:

$$\mu(p\Delta x) = \frac{1}{N}\sum_{p'} \overline{\epsilon(p'\Delta k)} e^{j2\pi p'\Delta k p\Delta x} = 0. \tag{2.13}$$

and its variance is

$$\begin{aligned}\sigma^2_0(p\Delta x) &= \tfrac{1}{N^2}\sum_{p'}\sum_{q'} \overline{\epsilon(p'\Delta k)\epsilon^*(q'\Delta k)} e^{j2\pi p\Delta x(p'\Delta k - q'\Delta k)} \\ &= \tfrac{\sigma^2_m}{N^2}\sum_{p'}\sum_{q'} \delta_{p'q'} e^{j2\pi p\Delta x(p'\Delta k - q'\Delta k)} = \frac{\sigma^2_m}{N}.\end{aligned} \tag{2.14}$$

In conclusion our model will assume that the noise is a single gaussian distribution added to the ideal image times the bias.

[3] Some other papers [8] it is modeled as a Rice distribution

2.3 Visual Analysis

After describing all the mathematics involved in the process, a visual inspection of the bias problem will be really helpful on figuring out the effect of all those things on an actual MR image.

2.3.1 Visual Analysis of The Bias

Let us take a look at the following rat brain image acquired with a surface coil at $9.4T$ (Fig. 2.1). This is a baby rat with a developing brain, which presents a mixed tissue represented as a single class in the image acquired. Unlike the human brain, for which gray matter and gray matter are readily distinct, the rat brain presents a mixed tissue.

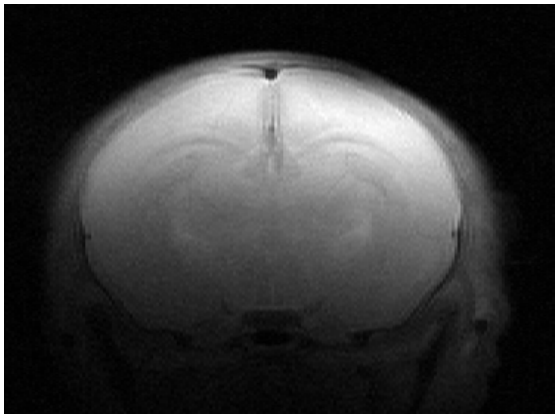


Figure 2.1: Baby rat brain acquired with surface coil at $9.4T$ (low contrast FSE image).

As this single class is surrounded by flesh and skull: we will first isolate the rat brain (procedure also called *skull stripping* or *masking*) in order to focus on the area [4] of interest (Fig. 2.2).

Looking at Fig. 2.2 it is easy to notice that the intensity is not uniform all through the rat brain. This brightness variation is the so called bias field, and it can be better seen in the Figure 2.3, in which the grey levels of the image are represented in a *3-D* view.

In order to see it better we can cut the masked rat brain image in both vertical and horizontal sections and look at the grey levels progression (Fig. 2.4). Notice that the vertical cut has a greater slope, because this section is parallel to the surface coil direction [5].

[4] Or in the case of *3-D* images, the *volume* of interest.

[5] When we talk about *surface coil direction* we mean the imaginary line in which the surface coil is sending and receiving signal.

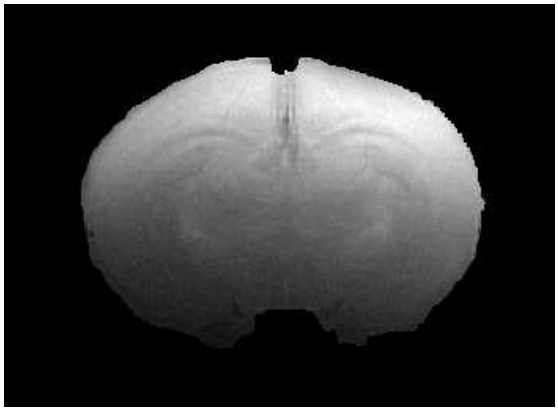


Figure 2.2: Masked rat brain.

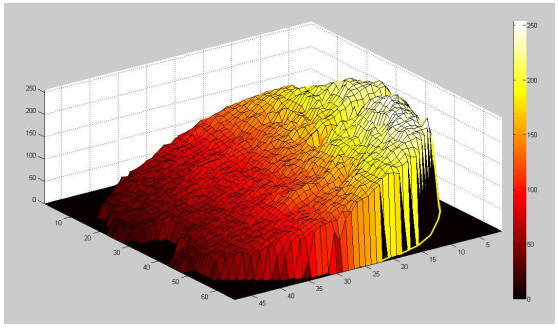
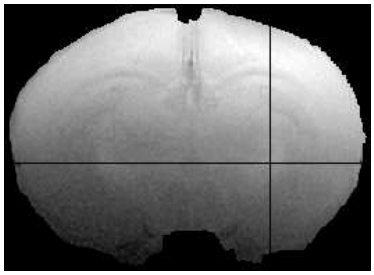
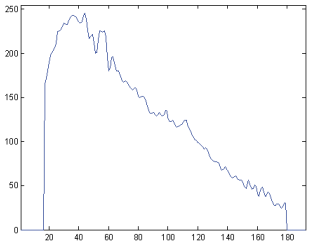


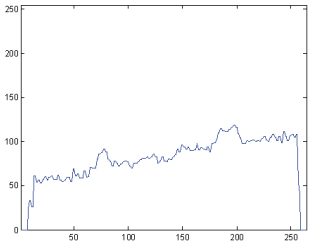
Figure 2.3: *3-D* profile of the masked rat brain.



(a)



(b)



(c)

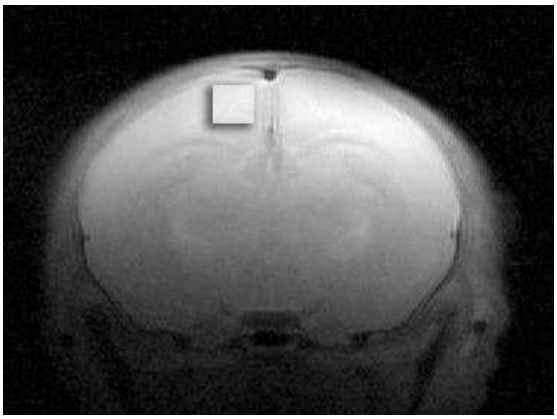
Figure 2.4: Vertical (b) and horizontal (c) profiles of the grey values of the masked rat brain image.

Looking at the vertical section we note that the maximum of the signal intensity is not exactly situated at the border of the brain, but a little bit further than that. That leads to a high frequency discontinuity in our image, which breaks our supposition that the bias is a smooth function, and it will suppose a challenge for us as seen in the chapters below.

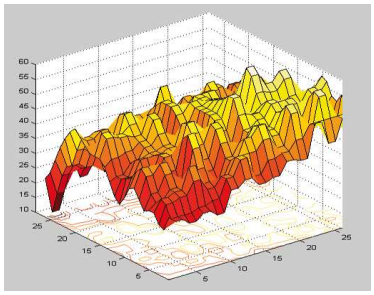
2.3.2 Visual Analysis of The Bias

Let us also take a closer look at the noise factor. For that purpose we will compare two different squares from the image. One of them has been extracted from the brain part, and thus it contains both biological noise and MR noise (Fig. 2.5(b)). The other one has been extracted from the background part of the image, and is supposed to have only MR noise (Fig. 2.5(c)).

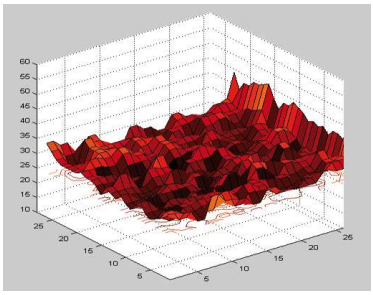
In this particular case, the bias field affects the longitudinal profile along the rat brain but almost not affects the transverse profile. Besides, the transversal profile presents the same value for each two neighbor pixels, while the longitudinal profile presents a variation pixel to pixel. To better show that effect, we have plot six consecutive columns from the image 2.2 (Fig. 2.6). It is showing that there is a strong correlation between the pixels in the longitudinal direction, but the correlation is much softer in the transversal direction.



(a)



(b)



(c)

Figure 2.5: Rat brain with marked tissue square (a), *3-D* view of the grey levels of the tissue square (b) and background square (c).

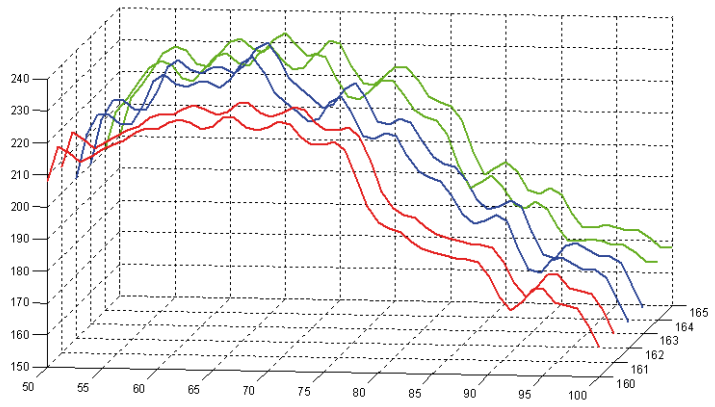


Figure 2.6: Six consecutive columns from the image 2.2.

2.4 Bias Field on Synthetic Images

In this section we will study the bias by simulating the same conditions observed in MRI. This will help us to better understand how the bias affects each kind of image. In this case we will mainly check the histogram behavior. For that purpose we use a simulated bias field (Fig. 2.7).

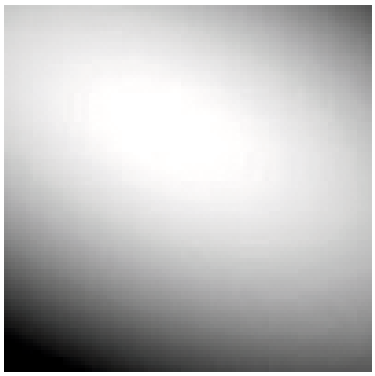


Figure 2.7: Synthetic bias field used in the tests, obtained from [17].

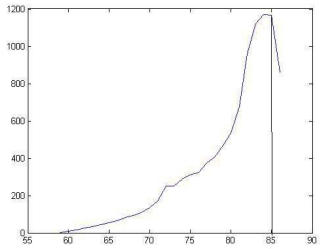
We want to understand how grey levels are affected by the bias field. For this purpose we will look at the histogram distribution of the grey levels present in the images. In our first test we want to know how the image histogram look like when applying the bias field. We will use three different *2-D* synthetic images, which are 100x100 pixel images filled with a single grey level (85, 170 and 255) [6] and then multiplied by the former synthetic bias field (shown in Fig. 2.8).

Looking carefully at those images we notice that:

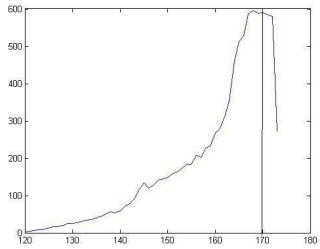
- The bias spreads the pixels along the dark side [7] near the true value, because we are multiplying this true value by a factor which varies between 0 and 1 and therefore the resulting grey value can only be smaller. In this concrete case our bias slightly overpasses the 1 value. Notice that the true grey value is arbitrary, and therefore we will keep on saying that $b(\vec{x})$ is confined between 0 and 1 for simplicity.
- If the value gets smaller, the resulting variance grows. Remember that the bias is multiplying by $min \leq b(\underline{x}) \leq max$ times a certain constant grey value. The variance is defined by the minimum bias times the constant grey value (this will be the most spread pixel). For the same bias then the higher is the grey value, the higher is the variance.
- As all histograms have the same number of pixels, and it means that the figures of the exponential seen in the three images is also the same. This leads us to the fact that a decrement of the peak produces an increase on the variance.

[6] In fact, those three synthetic phantoms will all look just the same as Fig. 2.7. By multiplying this bias field by a single value we are only scaling it.

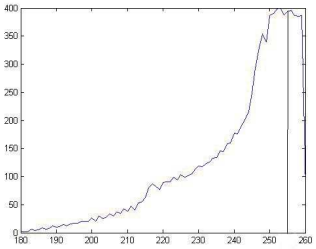
[7] That is to say: smaller grey values.



(a) $Value = 85$



(b) $Value = 170$

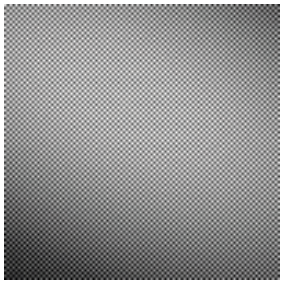


(c) $Value = 255$

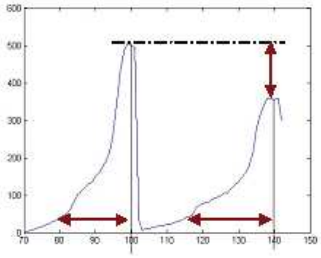
Figure 2.8: Single-class synthetic images.

- Note also that the histogram of the 85 value is smoother than the 170 one, and so on. That is due to a combination between the effects explained in the previous points and the sampling effect.

Now let us test a two-class synthetic image (grey values 100 and 140), applying the same bias on it (Fig. 2.9). We spread those two grey levels using maximum entropy all over the image, in order to better see the bias behavior.



(a) Synthetic image



(b) Histogram

Figure 2.9: Two-class synthetic image.

The variance produced by the bias is bigger in the case of 140 value, which leads to a fall of the peak situated in the true grey value.

To complete the study, and accordingly to the model of the signal seen in

section 2.1 [8], the next step is to add noise to the synthetic images. We add Additive White Gaussian Noise after applying the multiplicative bias. The sigma chosen is $\sigma = 20$. This noise is applied to the three synthetic images seen in the Fig. 2.8. The three image histograms have been mixed in the same axis (Fig. 2.10).

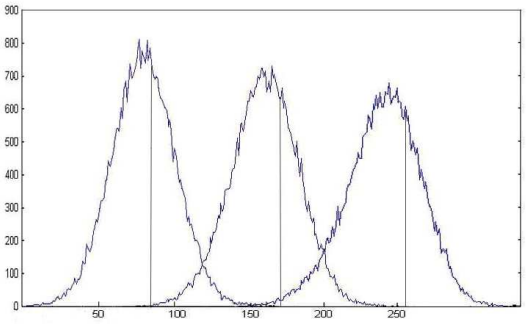


Figure 2.10: Synthetic images with bias field and noise added.

The higher is the true value, the more the bias spreads the pixels. It also produces a shifting of the peak respect to the known true value. Therefore the behavior is the expected one.

2.5 Effects of the Bias in a Phantom

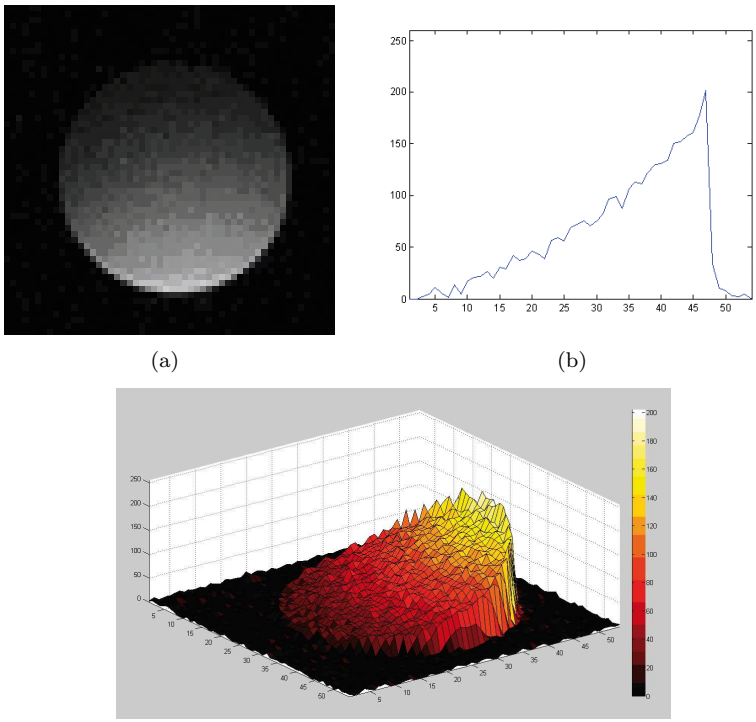
A phantom is usually defined as a device that simulates the human body or part of the human body and is used to calibrate or test the calibration of a detector that measures radiation emanating from within the body. Thus it is interesting for us to obtain the image of a single-class phantom to look at how the bias field modifies the image (Fig. 2.11).

All the particles reacts homogeneously during the MR acquisition, except for the n_{MR} noise and the bias field. This way we can clearly observe the bias behavior all over the object.

2.6 How to Correct the Bias Field

Now that we have studied the bias behavior, we must find a method to remove it from our image. Our goal is to achieve a common brightness level for the whole MR image. As said in the introduction, this bias field is a deterministic expression tied to complex physical interactions between the magnetic and electric fields with the living tissues, and this complexity is the reason why we

[8] Remember: $s''(\vec{x}) = o(\vec{x})b(\vec{x}) + n(\vec{x})$



(a) (b)

(c)

Figure 2.11: Slice from a single-class phantom acquired at 9.4T (a), axial slice profile of the center of the image (b), *3-D* profile of the grey levels (c).

will try to remove it directly from the corrupt image data.

There are lots of techniques developed to solve the bias problem [18] [19], and they can be classified into four main groups:

- Low-frequency model. One of the most important assumptions about the bias is that is is smooth, and therefore it is present in the low frequency part of the corrupt image. Those methods try to correct undesired frequencies directly from the image by averaging it and thus separating bias and useful information. The earliest methods rely on homomorphic unsharp filtering [5]. Those methods have been widely used due to their simplicity and efficiency in implementation. The most recent contribution has been the wavelet transform [6], that seems to be effective in correcting surface coil images.

- Hypersurface model. These methods consist in fitting a smooth functional model for the IIH map. They use to be iterative processes, which usually mean high computation times. They degree of complexity is higher than a low-frequency model. They are often combined with image segmentation. The parametric bias field correction is included in this type of correction, using Legendre polynomials to fit the bias field.

- Statistical model. This type of model assumes the IIH to be a random variable, and it uses statistical estimation to remove it from the corrupt image. Their characteristics are similar to the Hypersurface model: they are complex and iterative, and they are usually combined with image segmentation. The models used are the Gaussian distribution, or the Markov random field [7].

- Information acquired with other MR scans. It consists in acquiring extra information during the imaging process to aid in removing the bias field. An example about that is to use a phantom acquired in the same conditions as the image of interest, or a capture of a body coil image to help correct the surface coil image [8] and then use all this additional information to remove the bias.

Interested readers can refer to [9] in order to find a comparison between six of the already existing algorithms: Four-N3 (Sled et al., 1998), HUM (Brinkmann et al., 1998), EQ (Cohen et al., 2000), bfc (Shattuck et al., 2000), SPM (Ashburner and Friston, 1998), and CMA (developed by the Center for Morphometric Analysis at the Massachusetts General Hospital).

In this report we will focus only on two methods: the Low Pass Filtering and the Parametric Bias Field Correction [2] [21]. The Low Pass Filtering model has been chosen because of its demonstrated fitness in the correction of surface coil images, specially in a single-tissue image as a rat brain image. The Parametric Bias Field Correction has been chosen for being a well-suited tool in any kind of images, as well as for having access to its code (provided by the ITK C++ library).

Chapter 3

The Low Pass Filtering Correction

3.1 An Introduction to LPF Correction

In the last chapter we have already introduced the low-pass filtering method: it is mainly based on the supposition that the bias remains in the low frequencies of our image and therefore can be estimated by low-pass filtering. We are going to develop a simple method taking advantage of the frequency filtering concept.

First of all we must remember that the acquired signal is modeled in the $x-$space as $s(\underline{x}) = o(\underline{x})b(\underline{x}) + n(\underline{x})$. In surface coil images, $b(\underline{x})$ contains the biggest part of the energy of the low frequencies on that image. But what if we try to remove them directly from the image? We are going to observe the frequency behavior by calculating its Fourier transform:

$$\begin{aligned} \mathcal{F}(s(\underline{x})) = S(\underline{k}) = \mathcal{F}(o(\underline{x})b(\underline{x}) + n(\underline{x})) \\ = \mathcal{F}(o(\underline{x})b(\underline{x})) + \mathcal{F}(n(\underline{x})) = \mathcal{F}(o(\underline{x})) * \mathcal{F}(b(\underline{x})) + N(\underline{k}). \end{aligned} \tag{3.1}$$

So the low-frequency distortion is convoluted with the ideal signal, and thus we can not remove the set of digital frequencies that $b(\underline{x})$ contains by making the convolution with low-pass filter kernel. But applying the logarithmic transformation, we obtain:

$$\begin{aligned} log(s(\underline{x})) = log(o(\underline{x})b(\underline{x}) + n(\underline{x})) \\ = log((o(\underline{x}) + n(\underline{x})/b(\underline{x}))b(\underline{x})) = log(o(\underline{x}) + n(\underline{x})/b(\underline{x})) + log(b(\underline{x})). \end{aligned} \tag{3.2}$$

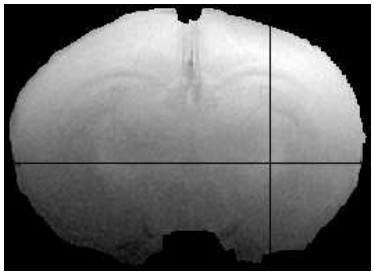
And therefore, in the frequency domain:

$$\mathcal{F}(log(s(\underline{x}))) = \mathcal{F}(log(o(\underline{x}) + n(\underline{x})/b(\underline{x}))) + \mathcal{F}(log(b(\underline{x}))). \tag{3.3}$$

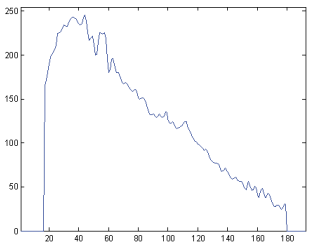
Now the bias term is additive, and it can be estimated by low-pass filtering the component $\mathcal{F}(log(o(\underline{x}) + n(\underline{x})/b(\underline{x})))$, assuming that the upper frequencies

are contained in there. Effectively, $n(\underline{x})$ contains the highest frequency part, and after the logarithmic transformation it will still remain in the upper part. If we consider a single-tissue ideal image, then $o(\underline{x})$ is completely flat and thus will be first kept by the low-pass filter (that is to say: it will be present in our estimation of the bias) and then removed from the original image when we use that estimation to correct the original image. That is not a problem, because the true continuous value is an arbitrary value that can be re-set according to our needs.

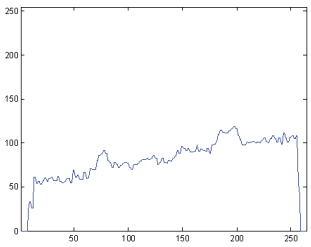
In order to see how the logarithmic transform is changing our image, we will compare both acquired image and the same image after the transformation. We already said ho the logarithmic transform changed the dynamics of the image, and thus re-scale the image to take profit of the whole range of grey values (from 0 to 255).



(a)



(b)



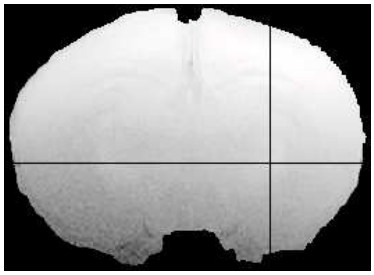
(c)

Figure 3.1: Vertical (b) and horizontal (c) profiles of the masked rat brain image.

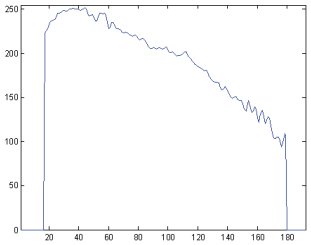
Observing the low frequency variation of both images (specially both vertical profiles) we can see how after the logarithmic transformation the variation is smoother and more curve. If we also observe the noise (high frequency variation), we can say that it has not changed significatively.

3.2 Proposed Method

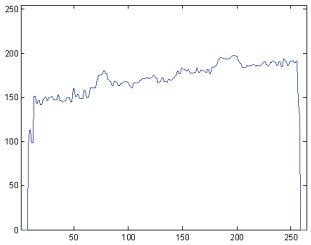
Once we have applied the logarithmic transform (Fig. 3.2) we must average it in order to estimate the bias field in the logarithmic domain. For this purpose,



(a)



(b)



(c)

Figure 3.2: Vertical (b) and horizontal (c) profiles after applying the logarithmic transform.

our technique is based in keeping only the single class of the rat brain, and thus directly obtaining the estimation of the bias after the average. That is to say, if we consider that the ideal rat brain image has a constant single class $o(\underline{x}) = C$, where C is a constant value, then:

$$log(s(\underline{x})) = log(C + n(\underline{x})/b(\underline{x})) + log(b(\underline{x})). \tag{3.4}$$

With the low-pass filter we are removing the $log(C + n(\underline{x})/b(\underline{x}))$ factor, except its constant value. Supposing that our low-pass filter is perfectly able to separate both factors, our estimation of the bias in the logarithmic domain will be:

$$log(\hat{b}(\underline{x})) = C' + log(b(\underline{x})). \tag{3.5}$$

Therefore it is important to keep only a single class in our image. We already did that when masking the original image (Fig. 2.1), but we must take into account other problems in order to make an accurate correction.

The first problem is that even inside a single-tissue zone we find some areas of too high-frequency which would lead to a bad estimation of the bias. As in the border effect case, those too high or too low values would be spread around by the filtering.

The second problem is that we must avoid the border effect when applying the low-pass filter. If we low-pass filter our image directly then, in border zones of the masked image (the limit between the tissue and the black background)

we are mixing both tissue values and zero values from the background. This effect can be avoided if we apply a Gradient Anisotropic Diffusion Image Filter, instead of the usual gaussian filter. But we will choose another option (explained below in 3.2.2) in order to combine the solution of this problem with the necessary [1] extrapolation of the bias to other zones of the image.

We propose here after a pre-processing chain to avoid such problems, which can be summarized in the following block diagram (Fig. 3.3):

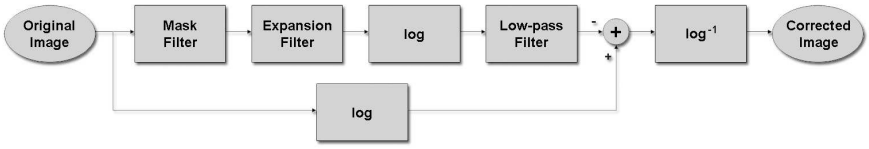


Figure 3.3: Block diagram of the LPF method proposed.

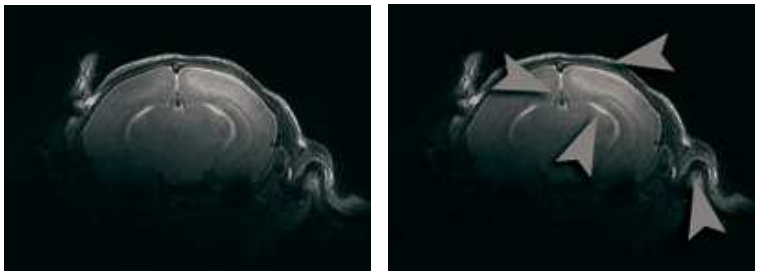
3.2.1 Computing the Mask

In order to illustrate this part of the process we have taken a single slice from the acquisition set of an adult rat brain (Fig. 3.4). We can note how even in the inner part of the brain there are several parts of the image (morphological peculiarities) which would impede the correct estimation of the bias field after the application of a low-pass filter. Those parts are areas with higher or lower grey values, and they are not totally removed after the filtering. Our aim in this section is to remove them before the estimation.

Note that those undesired parts are a high-frequency transition in our flat single class, and thus then their limits are characterized by a high gradient value. Therefore the gradient of the whole image provide interesting information about those undesired parts. Those desired parts of our image present after the gradient application a high-frequency but weak variation due to the noise, and a constant variation due to the bias. We first apply a low-pass filtering to the image in order to reduce the noise variation. After the filtering the *undesired areas* [2] still present much higher gradient values in their transitions. Therefore, setting a threshold between the low value of the tissue and the high value of the transitions, we will isolate the undesired zones. By now we are only marking the borders of those undesired parts, but we will also fill them with an iterative process (the Iterative Hole Filling Image Process provided by ITK library, see [25] for more information). The result is the image shown in Fig. 3.5(b). We will work without masking the image in order to show all the possibilities that presents this method.

As the image has not been masked yet, the background contains noise that must be also removed from our image. We will average the image in order to

[1] The necessity is also explained in 3.2.2.
[2] Note that we call them *undesired* in the context of the low-pass filtering.



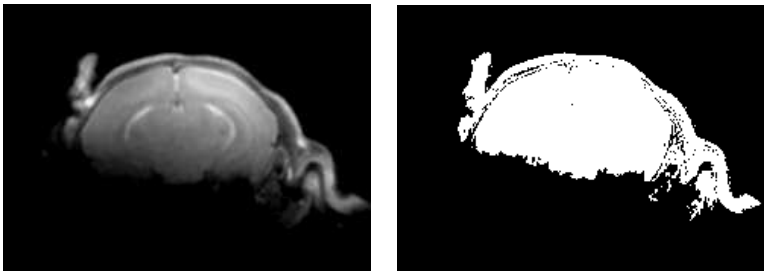
(a) (b)

Figure 3.4: Original image (a) and undesired parts marked (b).

(a) (b)

Figure 3.5: Gradient of the image (a) and Gradient image thresholded and filled (b).

make the background uniform. After this simple step all the background (with no variation) becomes the lowest grey level of the image, and now we only need to set a threshold to separate both information part and background (Fig. 3.6).



(a) (b)

Figure 3.6: Low pass filter applied to the original image (a) and thresholded image (b).

We already have all the information we need about the image: we have separated the undesired information (Fig. 3.5(b), which we will call A) and the whole set of information (Fig. 3.6(b), called B). Note how they both overlap in the image space. Our Region Of Interest (ROI) will be computed as follows:

$$ROI = B - A. \tag{3.6}$$

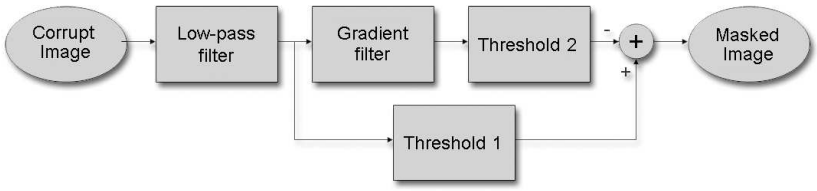


Figure 3.7: Block diagram of the masking process.

The result image is shown in Fig. 3.8.

This image is not yet useful for our purposes (extraction of the bias field by low-pass filtering). Next steps are to fill the holes of this image and extrapolate the tissue.

3.2.2 Filling the Holes: the Mirror

We are going to fill the holes, keeping at the same time the texture of the tissue. The solution we have developed consists in expanding the borders of the image to the background (black) zones. For this purpose we must keep the biggest

Figure 3.8: Image masked.

single connected object. Observing the Fig. 3.9 we can see those loose little particles which are going to be expanded as well. They do not have enough information to be taken into account, so we will remove them combining two morphological filters: the opening and the closing.

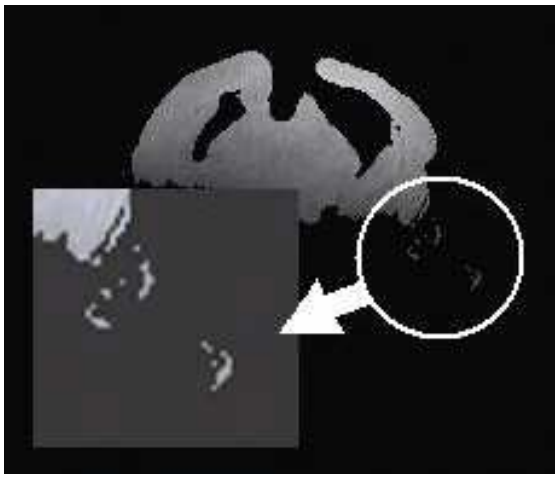


Figure 3.9: Undesired particles.

First of all we will threshold the image in order to work in a binary image. The opening will remove the cited small points, and then we will use this image to be computed with a closing in order to replace the original size of the information part. This process, as well as removed the little particles, is making the borders rounder. It is something we will appreciate in the next step when expanding the borders: the expansion follows a perpendicular direction from the border line, and if this line is too irregular then the expansion is not working properly. To keep the texture after the expansion we must avoid those perpendiculars to cross each other.

Finally the expansion is computed as follows: first of all we compute the

distance map of the final masked image. This will give us a resulting image that contains negative values in the inner part of the mask, and positive values in the outer part. Those values indicate the distance to the border, and the sign indicates in which side of the border is the pixel (positive values for texture pixels and negative values for background pixels). Our hypothesis is the following one: each positive pixel of a certain value corresponds to the negative pixel of the same value which is in the same perpendicular line to the border (Fig. 3.10).

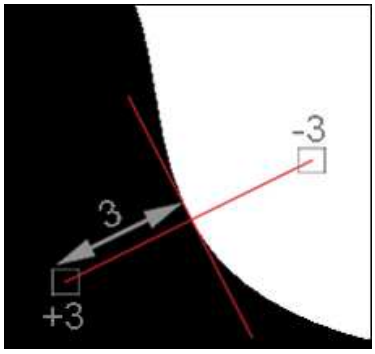


Figure 3.10: Scheme of the distance image.

Now we must find each perpendicular line to the border line. This will be obtained by first computing the gradient of the distance image, which will provide a *2-D* vector for each pixel indicating the steepest downslope from that point. This steepest downslope is in the distance image the perpendicular line to the border.

From each outer pixel (positive values) we are going to find its correspondent pixel inside. This is computed as follows:

$$\begin{aligned} i' &= i - 2 \cdot Distance \cdot \nabla_x. \\ j' &= j - 2 \cdot Distance \cdot \nabla_y. \end{aligned} \tag{3.7}$$

where $[i', j']$ are the coordinates of the pixel we will copy in the $[i, j]$ position, and (∇_x, ∇_y) is the normalized gradient vector of $[i, j]$. $Distance$ is the distance to the border and thus it must be multiplied by 2 in order to reach the position $[i', j']$.

We will compute this procedure iteratively: we first compute the mirror for a certain range of distances, and then compute it once again over the obtained image. This is done several time until (for example) the whole background is filled. The iterative mirroring is the best way to fill little holes inside the tissue (notice that a too large range of distances would lead to an undesired mirroring of the background from the other side of the tissue).

In the figure 3.11 we can see how the mirroring extrapolates the texture. We can note the iterative expansions, producing an effect of stripes or rings. This effect will be removed later by the low-pass filter.

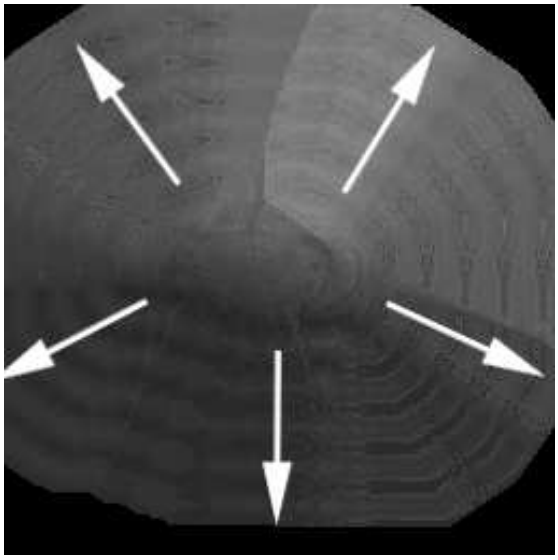


Figure 3.11: Result of the application of the iterative mirroring.

All these steps are summarized in the following block diagram (Fig. 3.12):

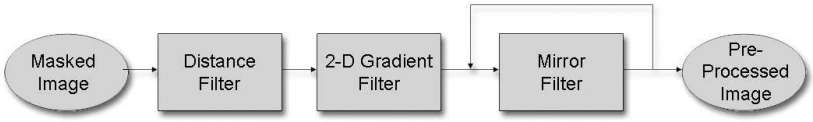


Figure 3.12: Block diagram of the expansion process.

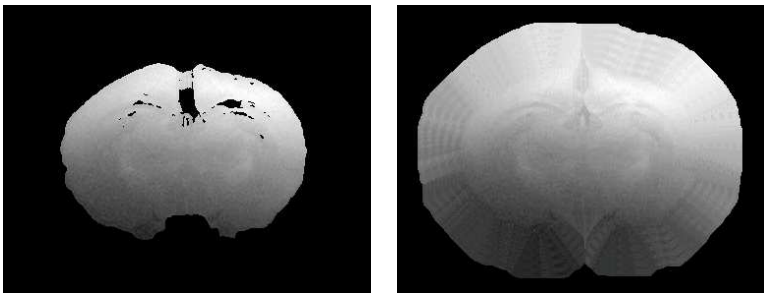
The interesting point about this mirroring is that it is doing an extrapolation of the brain tissue to those points that the masking removed. It does not matter if in those positions where before there was skin or skull. Another point in which we shall focus is that, in order to avoid the border effect, the minimum extrapolated distance needed is the half of the window size that we will use later in the low-pass filtering.

3.3 Applying the Low-Pass Filtering in *2-D*

We will apply the LPF method to the same *2-D* image we corrected before with PABIC (Fig. 2.1). They both will be compared in Chapter 5 of this report.

First of all we will mask the image as explained in 3.2.1. We will use a Gaussian filter with a window size of 100, and thus we must expand the edge of the image a number of pixels equal or bigger than that. In this case we will iterate the expansion five times with a distance limit of 10. It makes a total

expansion of 50 pixels (radial distance), which is exactly the half of the window size. We will test this lower limit because the mirroring process is expensive in terms of time-computation.



(a) (b)

Figure 3.13: Masked baby rat brain (a) and then expanded (b).

Note how the mirroring process is able to fill the little holes obtained after the masking process, as well as to keep a uniform texture all through the image.

Now we apply the logarithmic transform to the image 3.13(b) and then we apply a gaussian filter with a window of 100x100 and a sigma of 16. The result of this is the estimated bias, shown in Fig. 3.14.

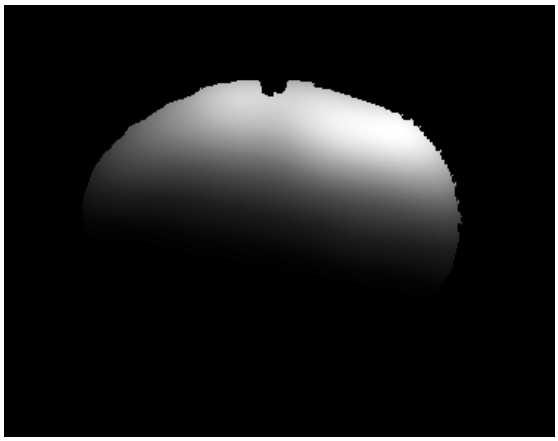
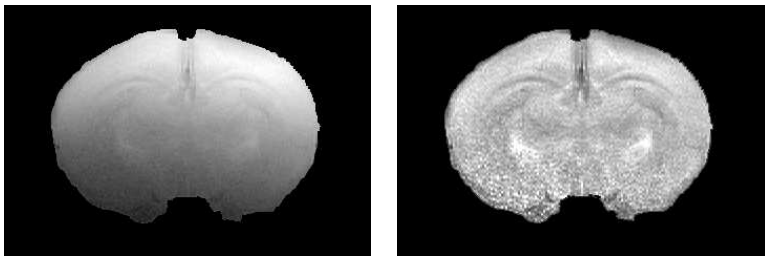


Figure 3.14: Bias estimated. Image obtained after a logarithmic anti-transformation and a final re-scaling to fit the range $[0, 255]$.

We have already estimated the bias. We must keep it in the logarithmic domain. The next step is to apply the logarithmic transformation to the image we want to recover (Fig. 3.15(a)), which is not the same as the image used to estimate the bias (Fig. 3.13(b)). Once we have both in the logarithmic domain we subtract the bias to the desired image ($log(I') = log(I) - log(\hat{b})$), obtaining the recovered image still in the logarithmic domain. Therefore, we must first

apply the anti-logarithmic transformation ($I' = 10^{log(I')}$) and then re-scale the resulting image I' to fill the $[0, 255]$ range of grey levels. The final result is shown in Fig. 3.15 [3].



(a) (b)

Figure 3.15: Biased image (a) and recovered image by LPF (b).

[3]For a more extense discussion about the corrected image go to Chapter 5.

Chapter 4

The Parametric Bias Field Correction

4.1 An Introduction to PABIC

The Parametric Bias Field Correction (PABIC) is a type of correction that works directly over the $r-$space image acquired from MRI [2] [21]. It is based on a simplified model of the imaging process, a model of tissue class statistics, and a polynomial model of the inhomogeneity field. This polynomial approach combines bias correction with histogram adjustment. Its ultimate goal is the determination of the correct class k for each pixel in the data set, in order to obtain an intensity-based segmentation of the scene. In the case of a human brain, those classes are usually taken as three: white matter (WM), grey matter (GM) and cerebral spinal fluid (CSF), and thus $k = \{1, 2, 3\}$. The other classes such as the skull or the skin are too small and morphologically complex to help in the correction, thus they are removed before the correction in a part of the process we call *masking* or *skull stripping*.

4.2 How Does PABIC Work

As an initial description, we can say that PABIC relies on two basic assumptions:

The first one is that $b(\underline{x})$ can be brought to a linear combination of m smooth basis functions. The basic functions selected in this case are Legendre polynomials [22]. Therefore the bias field is modeled by $b'(\underline{x}, \underline{p})$, where $\underline{x}$ are the spatial coordinates and $\underline{p}$ are the coefficients of the Legendre polynomials. The number of coefficients of $\underline{p}$ (m), depends on l, the maximum degree chosen for Legendre polynomials.

The second one is that image can be modeled in the $x-$space as a finite combination of regions, each one representing a single type of tissue. Those regions are characterized by their mean μ_k and their variance σ_k (both measured in grey levels), where k identifies the different tissues. In the concrete case of

rat brain images we will deal with one single class.

The goal of this method is to iteratively find the global minimum of the energy function, stepping out from non-optimal minimums. This minimum will determine the coefficients of the resulting $\underline{p}$ vector, which defines the bias estimation. We have seen that PABIC needs certain work previous to the correction: we need to choose and set certain parameters.

Class parameters:

- Number of classes K ($k = 1..K$): it is given by the number of dominant categories in the image scene.
- Mean (μ_k) and standard deviation (σ_k) of each Class: a bad choice could lead to a non-optimal solution. But minor changes do not affect the result.
- Mask: before starting to correct we need to remove those parts of the image which we decide they are not useful during the bias estimation.

Both original and mask images have the same size. The mask image will be superposed to the image we want to correct, and will determine those pixels from the original image that are going to be dismissed. A zero pixel in the mask will become a ruled out pixel in the original image. It can be a coarse mask, it is demonstrated that PABIC is not sensitive to masking errors smaller than 20% [21]. If we do not introduce our mask to the system, the same program will calculate a mask by itself, setting zero values on those pixels which have a grey value bellow the 10% of the histogram. The most important thing is to distinguish between both useful data and noisy background and then remove the latter one, so this automatically-computed threshold is useful under the hypothesis that the background is the darkest part of our image. There are also some other masking techniques such as the proposed by Rehm et al. [20] to remove skull, meninges, and blood vessels.

There are two types of mask needed as input:

- Input mask: indicates which pixels are going to be used in order to estimate the bias field.
- Output mask: indicates the area in which we want to apply the bias correction.

Bias Estimation Parameters:

- Type of bias field: in our case we are always dealing with multiplicative bias field.
- Order of Legendre polynomials: determines the accuracy and stability of the calculated bias field, but also the computation time. The usual parameters chosen are:
  - *2-D* images: up to the $2rd$ degree ($m = 2$).
  - *3-D* images: up to the $3rd$ degree ($m = 3$). Usual in Head-coils.

– Mammography: $m > 3$ is needed, which could mean that in this case the Legendre basis does not fit).

Optimization Parameters:
This set of parameters is optional and PABIC has already set the ones by default:

- Mutation Factors: c_{grow} and c_{shrink}. Typically: $1.01 < c_{grow} < 1.1$, and $c_{shrink} = c_{grow}^{1/4}$.
- Maximum Number of Iterations: we don't need a very high precision in the determination of our set of parameters. We can choose a reasonable [1] value.
- Initial Value of Coefficients $\underline{p}$: it is not critial, but a good initial choise will speed up the calculation.

For those readers interested in a more complete explanation of the procedure please refer to [2].

4.3 Tests: Two-Dimensional Images

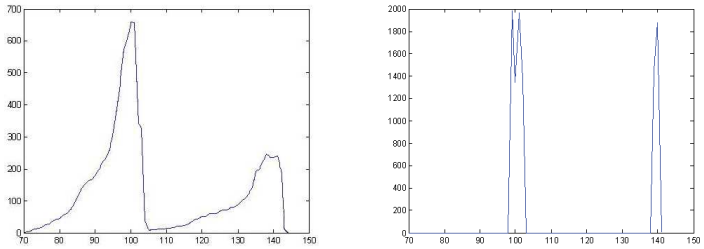
Despite in papers [2] and [21] we can find complete tests of the method we are dealing with, there are still some questions not answered yet that we will try to clarify in this section of the text. For this purpose we will treat the BIAS program as a 'black box', setting different inputs and observing their respective output. We will use the utility MRIBiasCorrection [23] from the ITK [24] C++ library.

4.3.1 Correcting Bias with no Noise

We will start testing a three-class synthetic image obtained with the same procedure shown in 2.4. In this case we have three different classes [2]: the fist two of them with grey levels 100 and 140 respectively, and we set another class very close to the 100 grey level, at 103 grey level. See how 100 and 103 peaks get overlaped in the histogram (Fig. 4.1(a)). No noise is added in this experiment. The parameters chosen for the PABIC correction are the following ones, as suggested in the previous section: grade $m = 2$ for Legendre polynomials (as we are treating a *2-D* image), and we also set the known true values $\mu_1 = 100$, $\mu_2 = 103$ and $\mu_3 = 140$. In this case we set each sigma to $\sigma_1 = \sigma_2 = \sigma_3 = 1$, because it is not possible to set it as zero value, which we know is the true value (remember that no noise is added). The result of the PABIC correction is shown in the Fig. 4.1:

We can see that PABIC is perfectly able to distinguish even both two close peaks (100 and 103 grey values) after a certain number of iterations. That is to say: it is able to correct the inhomogeneity effect and recover the original grey levels.

[1] The meaning of *reasonable* is discussed later in 4.4.
[2] We can think of them as three different tissues.



(a) (b)

Figure 4.1: Histogram of the synthetic phantom (a), histogram after correction (b).

4.3.2 Correcting Bias with Noise

We will test the same synthetic image once again, but now we are adding noise to the image (AWGN, $\sigma = 20$) (Fig. 4.2).

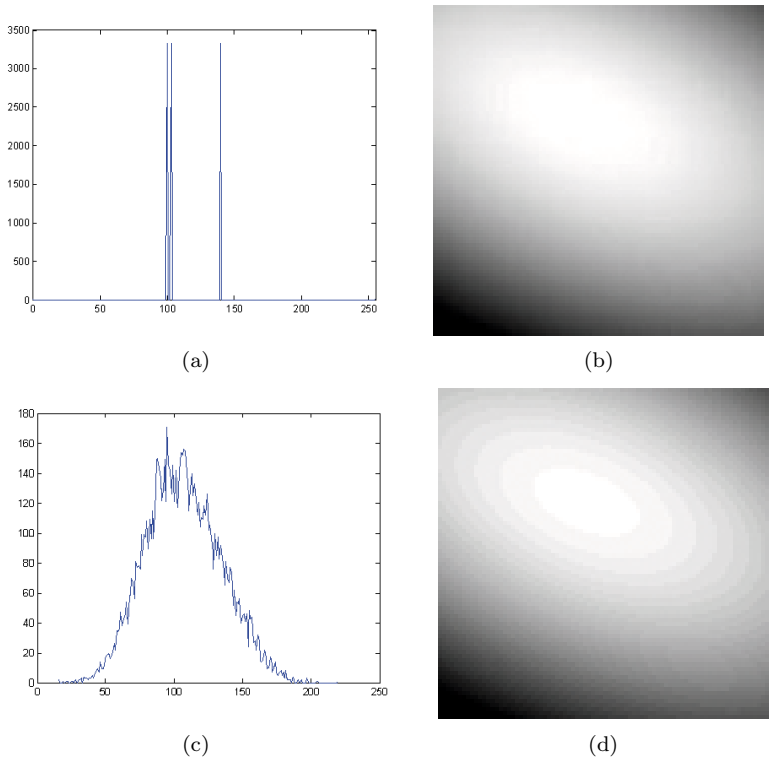
We can see in Fig. 4.2 that both bias (the true one and the corrected one) are the same one. Looking at the histogram in 4.2(c) we notice that, if had not knew them *a priori*, it could not be possible to determine the true values for each μ_k as well as of each σ_k. So our question is how sensible is the PABIC method to deviations in those parameters.

4.3.3 Correcting an Real MR Image

In this subsection we will use the image of the rat masked rat brain shown in Fig. 2.2. For that purpose we must first estimate the mean and the variance of the single class. For this estimation we are going to extract a sample from the image we are dealing with, taking care to take it from a place of little bias (Fig. 4.3).

This image sample contains both MR noise and biological noise. Observing the smoothed histogram we note that, despite the little inhomogeneity effect, it is shaped as a gaussian distribution. Therefore we will say that the 10% line of the histogram is defining the 4σ distance. As we observe $4\sigma \cong 250 - 200 = 50$, so $\sigma \cong 50/4 \cong 13$ (we do not need too much precision in this kind of estimation), as well as $\mu \cong 230$. The recovered image is shown in Fig. 4.4.

In order to test how critical is the mean chosen, we have repeated the previous test setting the estimated true mean as $\mu = 150$ and $\mu = 90$, obtaining the same result (comparing their bias extracted and estimated $\underline{p}$ coefficients). The explanation of this effect is that in PABIC both mean an sigma are mainly used to distinguish between different tissues in the segmentation parallel process. Therefore, the importance of those coefficients in a single-class image is limited.



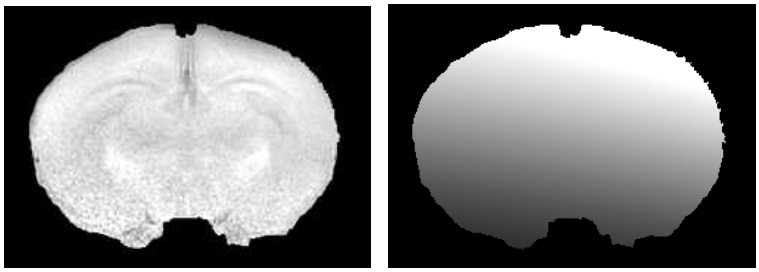
(a) (b)

(c) (d)

Figure 4.2: Histogram of the synthetic phantom before the application of the bias and the noise (a), image of the bias applied (b), histogram when bias and noise are already applied (c), histogram after correction (d).

(a) (b) (c)

Figure 4.3: Position of the square extracted (a), image of the square extracted (b), histogram of this square (c) smoothed and with a threshold line set at 10% of the smooth line.



(a) (b)

Figure 4.4: Image corrected with PABIC (a), and bias field extracted (b).

4.4 Tests: Three-Dimensional Images

An important feature about PABIC is its ability to correct the bias field in the *3-D* space. Remember that an MR acquisition translated into the $r-$space is corrupted by the bias field in all direction. That is to say: if we think about an MR acquisition as a *3-D* volume consisted of N slices, then the brightness variation in all x, y and z cartesian axis caused by the bias field.

We will use PABIC to correct a *3-D* MR acquisition at $9.4T$ of an adult rat brain, composed of 12 slices with 256x256 pixels per slice. This way we will test the method working full power. The set of slices to be corrected are these ones:

In this case we will not only evaluate the goodness of PABIC's correction, but we will observe other features such as the computation time or the evolution of the $\underline{p}$ vector per iteration.

First of all, we know that a *3-D* correction will increase the computation time compared to *2-D* correction. The reasons are the following ones: first of all the amount of data is multiplied times N, where N is the number of slices; the other reason is that we must set 3 as the degree of Legendre polynomials instead of degree 2 in *2-D*, which makes the $m-$space increase from 6 dimensions in *2-D* to 20 in *3-D* and thus PABIC must manage more information per iteration. We will take advantage of this increase of the computation time in order to obtain a global vision of the PABIC behavior, performing large time computations. The iterations shown below have taken about 2 days long. The main problem about the *2-D* tests is that the computation time was so low that we could not obtain a precise measure using minutes as the lowest time scale.

The other feature used is the $\underline{p}$ vector. We must keep in mind that each $\underline{p}$ vector can be represented in a $m-$dimensional Euclidean space. We will use this output to compare different simulations, taking advantage of Euclidean operators such as the distance, which will help us to estimate the goodness of each correction.

Having already described all features we will use to compare, now we must decide the way in which the simulation is done. First of all we already know

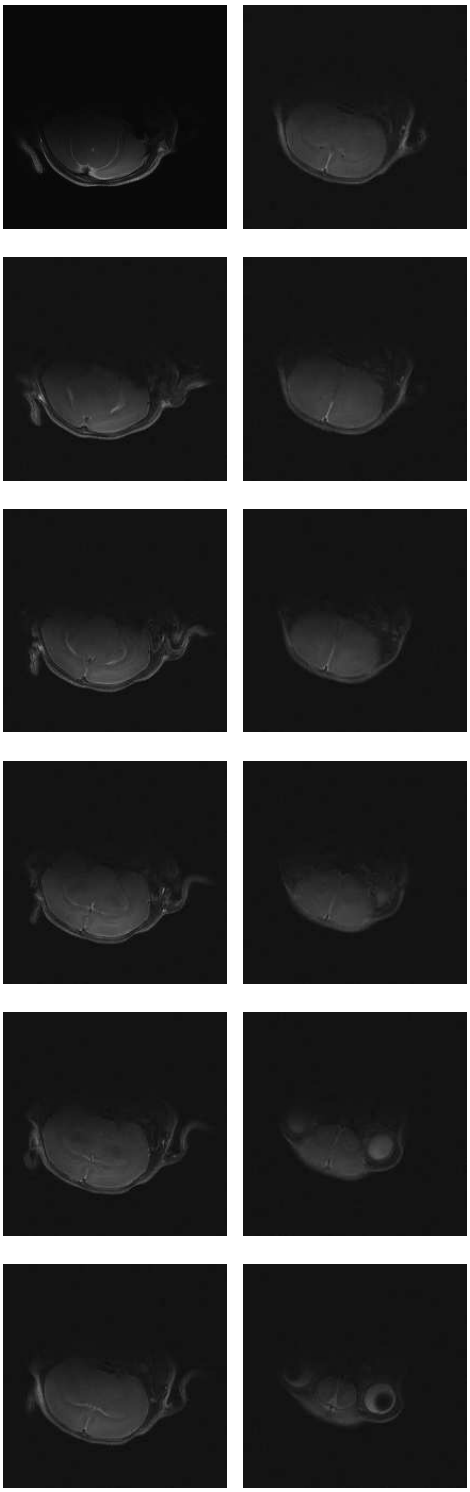


Figure 4.5: MR acquisition of an adult rat brain, 12 slices, 256x256 pixels per slice, sorted in columns (first column: slices 1 to 6; second column: slices 7 to 12).

that PABIC method is not sensible to masking errors smaller than 20%. Thus we will apply the method without any input mask at all. Remember that it means that the PABIC is calculating its own mask for us. In this case threshold is set at grey level 2 (pixels under the 10% value of the combined histogram of the whole set of images).

The number of classes is set at 1, and its mean and the sigma set as input are $\mu = 90$ and $\sigma = 7$ (estimated over a single slice of the whole set).

The test will consist in running consecutive simulations of the same data set (Fig. 4.5) with the same conditions (no mask, same mean, same sigma...) but changing the *maximum number of iterations* input, up to 200,000 iterations in steps of 10,000. First of all we must take a look at the time evolution per iteration, shown in Fig. 4.6.

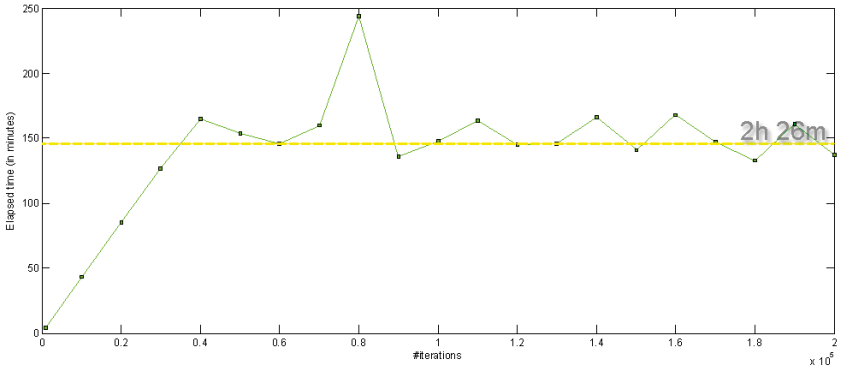


Figure 4.6: Evolution of the computing time while increasing the maximum number of iterations.

Note how the elapsed time is increasing until the test reach the 40,000 maximum iteration quote. After that time values get stable around a delay of 2h 26m. Now we can combine this latter information with the information given by $\underline{p}$ (Fig. 4.7).

Let us clarify the information shown in this graphic. First of all we must remark that $x-$axis correspond to the number of iterations, while the $y-$axis is defining the modulus for $\underline{p}$ vectors in the $m-$space (id est: $\|\underline{p}\|$), as well as the mean square error between two different vectors ($\sqrt{\|\underline{p}_2 - \underline{p}_1\|}$). Therefore the time evolution (in green in the graphic) has been arbitrarily set in the graph with no $y-$axis concrete values (all the information has already been given in Fig. 4.6). By mixing both $\underline{p}$ evolution graph and time graph we are able to globally understand PABIC's behavior.

The corrections have been computed with an Intel(R) Core(TM)2 CPU, 6600 @ 2.40GHz, 3.25 GB of RAM. The power of the computer is the main responsible for the time elapsed when running the correction, thus the time results obtained are tied to the computer used. Even so the results of this test can be translated into another computer by only multiplying all times by a fix scale

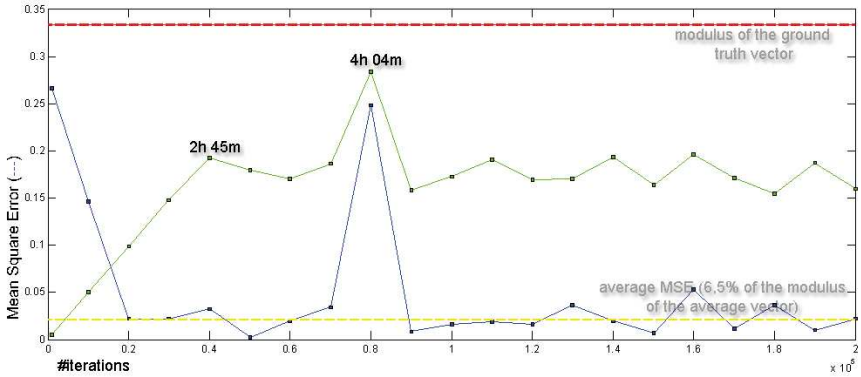


Figure 4.7: Evolution of the p vector and time elapsed while increasing the maximum number of iterations set in PABIC.

factor (that is to say: multiply the same graphic for a certain 'computation speed' factor).

To compute the evolution of the mean square error (in blue in the graphic) we had first to decide a ground truth for the $\underline{p}$ vector. For this purpose we averaged all $\underline{p}$ vectors obtained after the simulation get stable (as seen in the graphic it happens when we reach the 20,000 maximum iterations; we also excluded the aberrant measure obtained at 80,000 maximum iterations). The modulus of the resulting averaged vector (our *ground truth*) is represented by the red line. The yellow line represents the average between mean square errors, excluding the same measures as in the previous average. The modulus of the averaged mean square error represents the 6.5% of the modulus of our vector chosen as ground truth, so it is a value low enough to make us rely on the election of the ground truth $\underline{p}$ vector.

Now let us focus on the peak on both mean square error and computation time at 80,000 iterations, which appears after all other mean square errors values got stable around the 0.02 value, as well as the computation time got stable around 2h 26m (Fig. 4.6). In this case we must rely on a visual inspection of the correction (Fig. 4.8). We select any of the defined as *stable* corrections.

The first thing we notice is that the correction really seems to obtain a similar brightness in one slice, and also between different slices. After a visual inspection comparing the selected correction and the one obtained at 80,000 iterations we notice no difference. Another type of verification was applied to test the accuracy of the correction. Let us introduce first the following simulation, in which we forced the PABIC method to reach the maximum number of iterations in all cases (Fig. 4.9):

We notice a linear increase of this computing time. It means that there is an internal criterion in PABIC which stops the simulation when the result reached is good enough. It is not mandatory for the process to reach the maximum

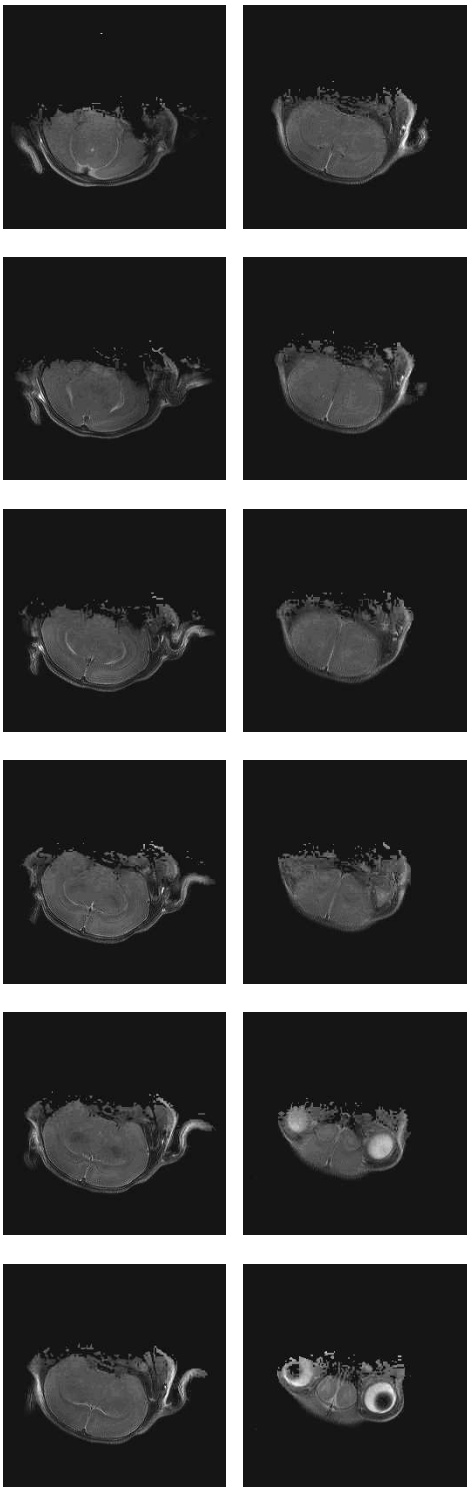


Figure 4.8: Correction of the rat brain seen in Fig. 4.5, sorted in columns (first column: slices 1 to 6; second column: slices 7 to 12).

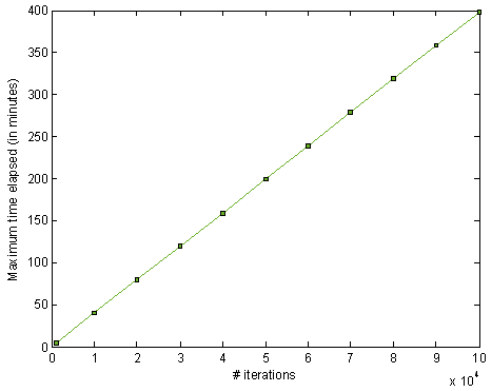


Figure 4.9: Evolution of the maximum computing time while increasing the maximum number of iterations.

number of iterations. As a consequence of that, we can establish a criterion relying on the PABIC criterion: those corrections which have been stopped before reaching the maximum numbers of iterations set, are considered as good enough. Let us check if it is like this in the case of the 80,000 iteration point we were talking about (graphic shown in Fig. 4.10).

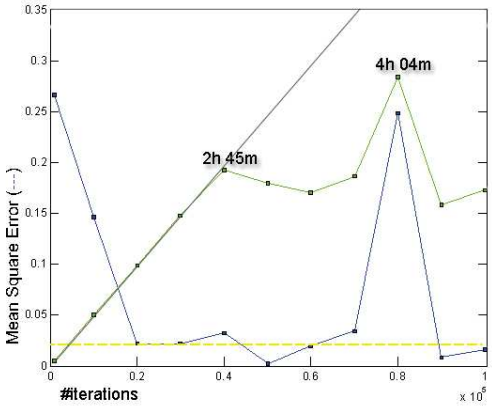


Figure 4.10: Zoom of the Fig. 4.7. Line in black showing the linear increasing of the elapsed time.

Now, keeping all the parameters as set before, we will study the behavior of the transitory part (the non-stable zone before 20,000 iterations). For this purpose we will limit the test up to 60,000 iterations in steps of 3,000 iterations. The result is the shown in Fig. 4.11).

We notice the appearance of another aberrant measure, this time at 45,000 iterations. Looking carefully at the temporal progression we can see that PABIC was not able to find the global minima before reaching the maximum number of iterations. This effect seems to appear randomly. If possible, it is highly

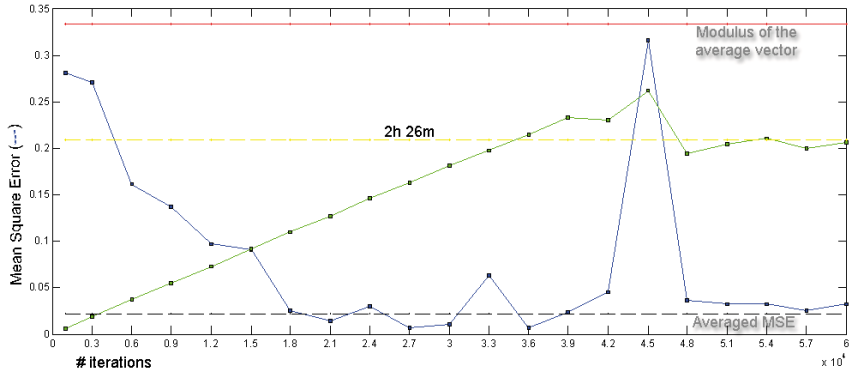


Figure 4.11: Test focusing on the transitory part.

recommended to keep a database of $\underline{p}$ vectors corrected for similar acquisitions. We can prevent those undesired measures, as well as reduce drastically the computation time, by averaging all these vectors and setting this average vector as an input for the PABIC, as the initial value of $\underline{p}$ coefficients. This can only be done when correcting similar acquisitions (same type of coil, same position of the coil in the image...).

4.5 Computation Time

At the end of the previous section we suggested that a clever use of the initial $\underline{p}$ coefficients could help us to improve the correction efficiency. Now we are going to check some other ways such as the masking or a previous average of the image, and see their effects on the correction. The purpose of this section is if to know whether if we can profit them or not in order to speed up the computation time.

4.5.1 Masking

First of all we will test how a non-automatic masking affects the correction. We are looking for a reduction in the computation time (allowing to find the global minimum earlier), and the hypothesis is it could be done by removing those parts in the image which do not belong to our defined classes $k = 1..K$. But instead of that, our tests failed in all types of masking used, being the automatic masking of the PABIC being the one which gave better results.

Using the same *3-D* image shown above (Fig. 4.5, the first mask tested was an accurate threshold in order to completely remove the background noise from the image. After 100,000 maximum iterations the test linearly increased up to 6h 40m (remember that the average of the stable zone of the previous tests was 2h 26m). The second masking method consisted in applying the method proposed in 3.2.1 to the whole set of images, slice per slice. The result of that

latter test was exactly the same: after 100,000 maximum iterations the test was increasing up to 6h 40m.

The reason of such bad results is probably that we are introducing a strong discontinuity in the z direction. Then we could minimize this effect by applying an average filter after the masking application, thus reducing all discontinuities in x, y and z directions (it will be discussed in the next section).

4.5.2 Averaging

In order to improve the results shown in 4.5.1, we will combine both accurate threshold and averaging (Fig. 4.12). The Masking shown in 3.2.1 has been discarded because of its hard discontinuities in z direction.

We will first compare the time progression achieved in this case with the one we had in the first test, both combined in Fig. 4.13. See how the both of them get stable around the 40,000 iteration value, but at the same time the linear increase of the new test is higher than the former one, which means that PABIC is spending more time per iteration in this new test. The result expected was the contrary: an averaged image should be less time-consuming. Now let us see what happens with the goodness of the results obtained with this new test.

In the Fig. 4.14 we can see the evolution of the mean square error between the $\underline{p}$ vector obtained in each sub-test, and the averaged $\underline{p}$ vector which we already took before as round truth. Taking as a reference the averaged mean square error obtained in the other test, we observe that this new correction maintains Mean Square Error (MSE) values far from that line. Comparing both images 4.12 and 4.5 we see how the eyes of the rat, present only in the three last slices in Fig. 4.5, are now clearly visible in the five last slices in 4.12, due to the *3-D* average. It causes an increasing on the computation time, because paradoxically the shape of the new set of slices is more complex than before, and also is leading to a wrong solution because we are changing the morphology of the original rat brain. Thus this masking plus averaging is not reaching the desired goals neither in time nor in goodness.

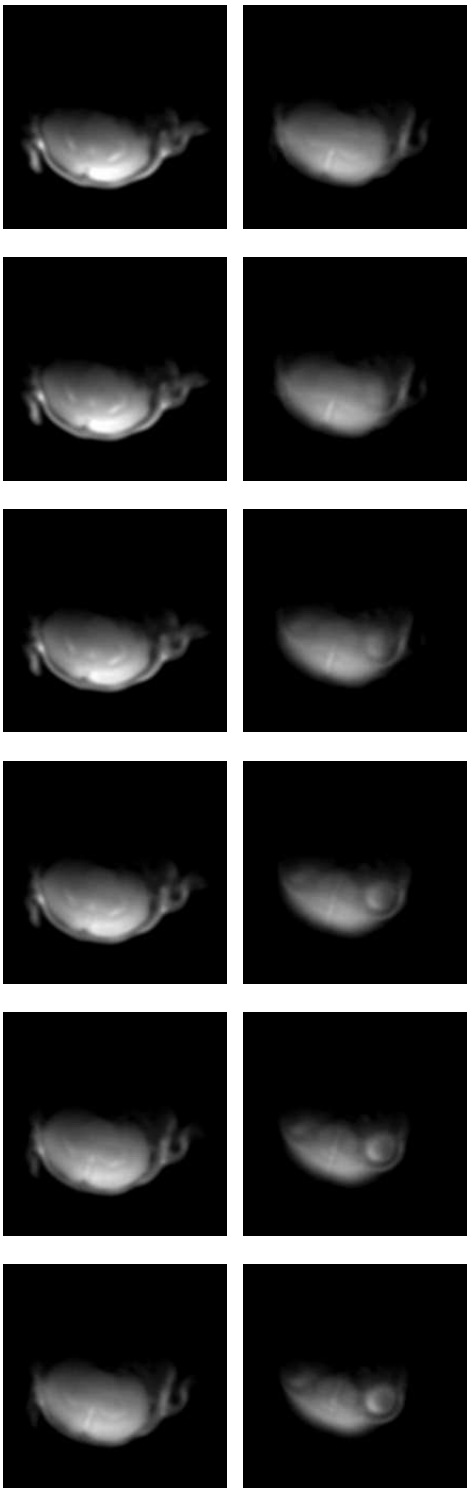


Figure 4.12: *3-D* image of the adult rat brain, first thresholded and then averaged, sorted in columns (first column: slices 1 to 6; second column: slices 7 to 12).

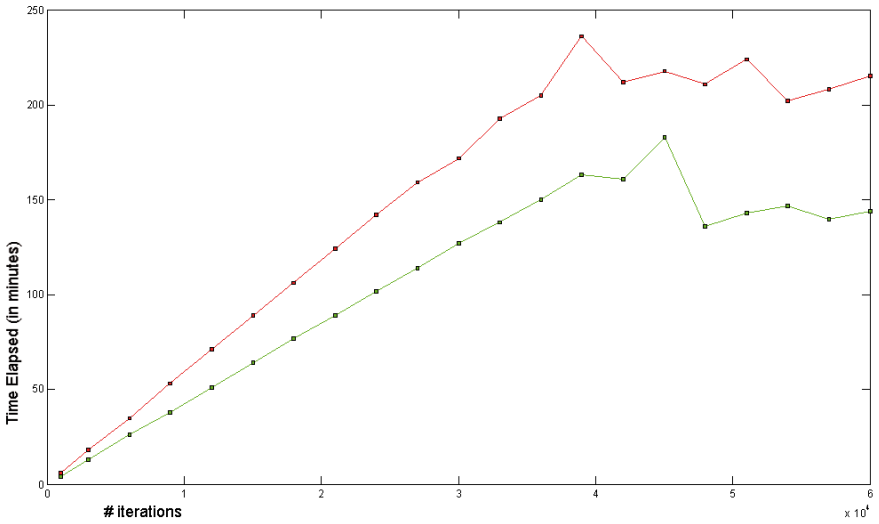


Figure 4.13: Graphic showing the time progression of the first test (in green) and the new test where we have averaged an thresholded de image (in blue).

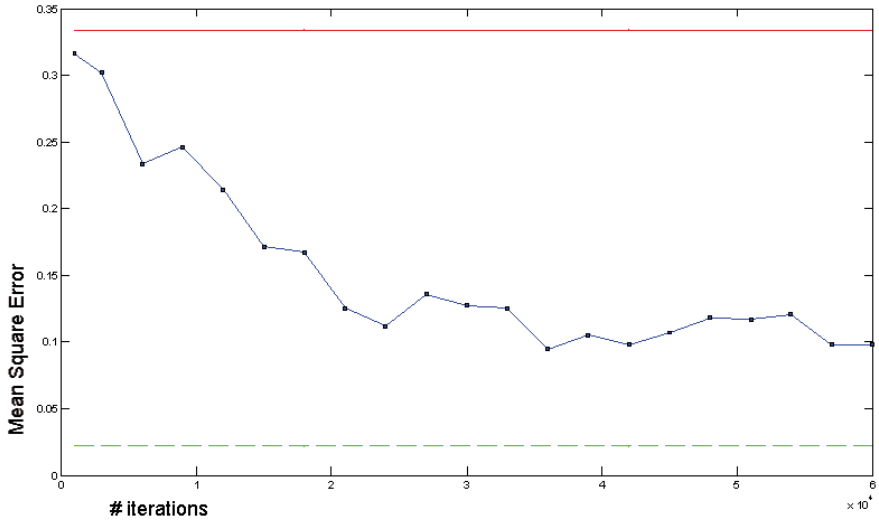


Figure 4.14: Progression of the mean square error per iteration (in blue), former averaged mean square error achieved (in green) and modulus of the ground truth p vector (in red).

Chapter 5

Results and Discussion

5.1 Validation

The validation of the results is a difficult part in bias correction. What makes so hard to determine the goodness of a correction is the lack of a ground truth image.

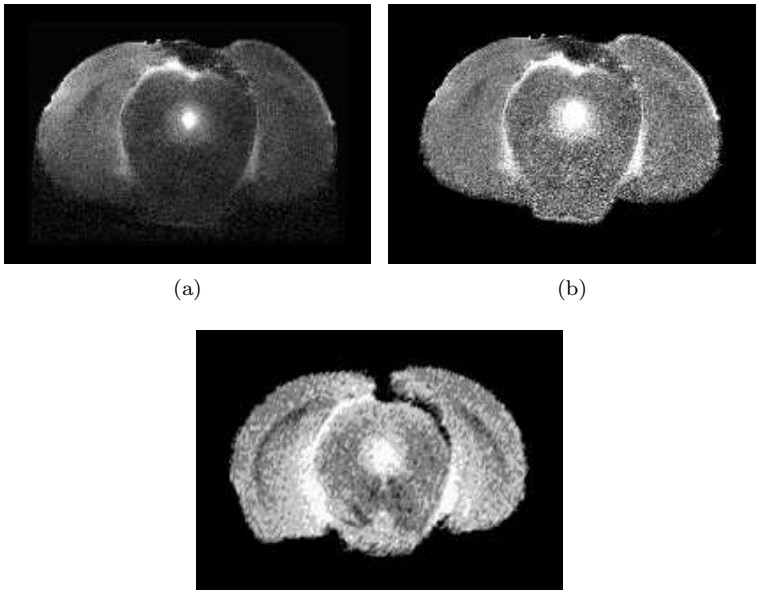
There are some ways to obtain a ground truth image, such as freezing a rat brain and then do a physical section taking a photograph from it. This method have two main problems: the first of them is that when we freeze the rat brain we are changing the tissue-characteristics; the second one is that the physical slice must correspond exactly to the one obtained with MR acquisition. As well as having those two problems, it is not easy to obtain the tools to carry out this job [1].

A simplest way to succeed in the validation is to compare both Surface Coil acquisition and Volume Coil acquisition. Remember that by now we are treating with Surface Coil images with a strong bias, while Volume Coil acquisitions present a much lower bias and thus can be taken as ground truth. For that purpose we obtained two acquisitions of the same rat brain with those two methods. The procedure was applied to an excised rat brain. Notice that both images (Fig. 5.1) can only be visually compared, because the brain position is changing significatively when changing the coil. Thus we can not compare them directly computing, for example, the mean square error. We have chosen the two most similar slices.

Notice also how the volume coil acquisition has a lower resolution than the surface coil one. But despite this fact we can also see that in the zones of the surface coil image where the bias is too strong (which we already know that means a poor SNR) there are details that can not be seen; looking at the downer side of the volume coil image we can distinguish parts of the morphology that can not be recognized in the recovered surface coil image (Fig. 5.2).

It demonstrates that in those parts in which the bias is very strong, the

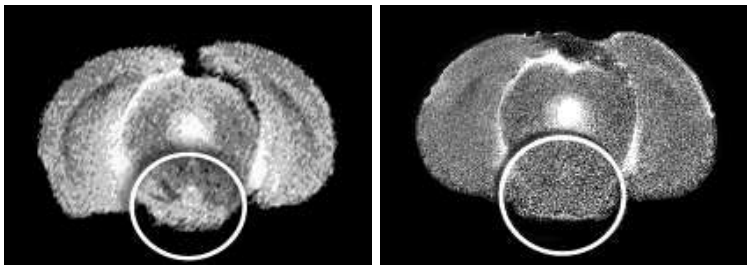
[1] For more information about this technique please refer to Visible Human Project [27].



(a) (b)

(c)

Figure 5.1: Surface coil acquisition (a), surface coil acquisition corrected by LPF (b), volume coil acquisition (c).



(a) (b)

Figure 5.2: Volume coil acquisition (a) and surface coil acquisition (b). Morphological peculiarity marked.

Signal to Noise Ratio (SNR) decreases drasticly.

A usual approach to estimate the goodness of a correction is looking at the histogram of both original image and corrected image. The validation consists in checking if the corrected histogram groups the pixels into the class which belongs those pixels. In our case we are treating single-class images and thus we expect all the pixels to be grouped in a single class.

We also must remember the main goal of all our efforts: to flatten the image intensity profile. Thus we can also rely on another type of validation: looking at the vertical and horizontal profiles of the image, and this way testing the behavior of the brightness along those cuts. This validation must be done visually, distinguishing with the eye between on one hand noise and morphological shapes and on the other hand the low frequency variation. The goal is to obtain a null low frequency variation, both in the vertical and in the horizontal profile. This method will also be used to compare the LPF correction and the PABIC correction.

5.2 Comparison

In order to compare both methods Low-Pass Filtering (LPF) and Parametrical Bias Field Correction (PABIC) shown in Chapters 3 and 4 respectively, we will use the two types of validation method explained: both comparing their histograms and checking visually the horizontal and vertical profiles of the image.

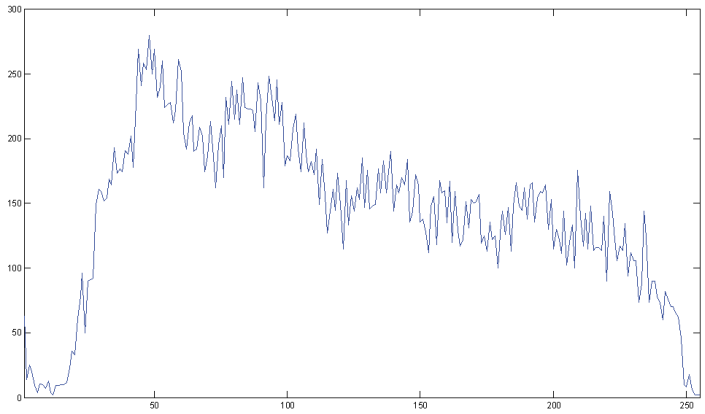
5.2.1 Histogram Comparison

For this first validation we will first use the histogram of both types of correction studied, comparing it to the original histogram (the non-corrected one). We can observe the three of them in the Figure 5.3. 5.7).

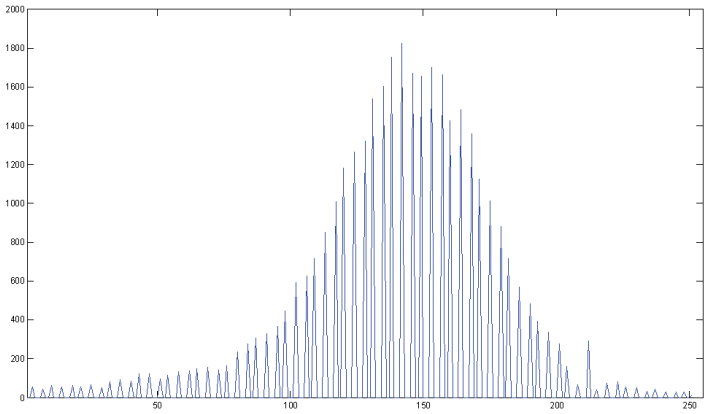
Remember that we expected an grouping of the pixels around a single mean value μ with a single variance σ^2. Observing both Figs. 5.3(a) and 5.3(b) we can see how this objective has been reached. We can also observe in both a Gaussian distribution of the pixel grey values. Note that while the parameter μ of this class is arbitrary, σ^2 is not.

The discontinuity observed in both corrections is due to the expansion of the grey values. In order to observe the behavior of the single class, too variant values have been cut. Thus we have reduced the grey value range from 256 to approximately 75 (in both cases), and then they have been re-scaled in order to fill the 256 range.

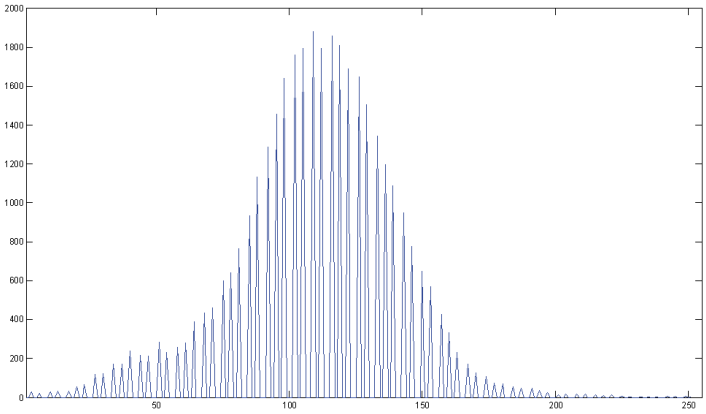
The information given by the histogram is limited, thus we must check other types of comparison.



(a) Histogram of the masked original image shown in 2.2.



(b) Histogram of the PABIC correction of image 2.2.

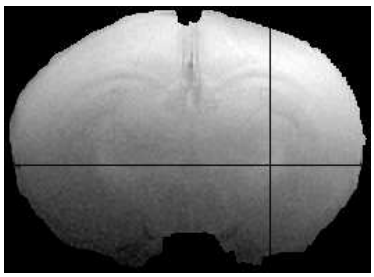


(c) Histogram of the LPF correction of image 2.2.

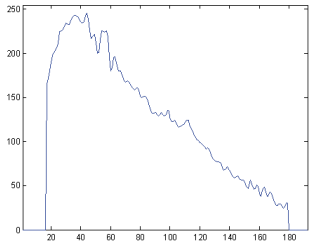
Figure 5.3: Histogram comparison. Zero pixel value is omitted in all cases.

5.2.2 Profile Comparison

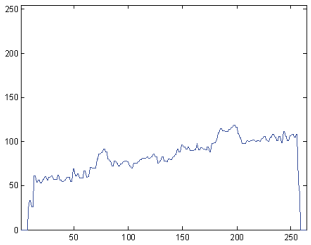
This type of validation consists in observe the vertical and horizontal profiles of each image in order to visually check the variation of their brightness. In the figures 5.4, 5.5 and 5.6 we can see details of the original image, the PABIC correction and the LPF correction, respectively.



(a)



(b)

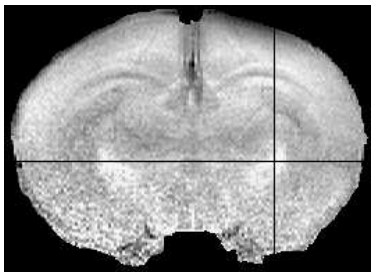


(c)

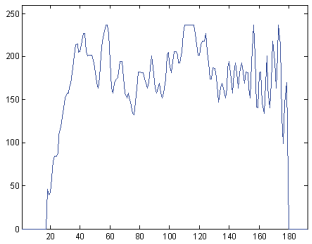
Figure 5.4: Original image (a), vertical profile (b), horizontal profile (c).

Looking at the graphics of the PABIC correction (Fig. 5.5) and LPF correction (Fig. 5.6) we can say that they are practically the same. The most noticeable difference is the hard slope seen between the pixel values 10 and 30 (horizontal axis) in the vertical profile of both corrections (Figs. 5.5(b) and 5.6(b)): LPF has been able to correct it better than the PABIC. The explanation for that is that PABIC uses very smooth functions (the Legendre polynomials) that are not able to model such a hard variation. The Gaussian low-pass filter did not succeed completely on that, but the result is better in this case. This is directly tied to the window size: the bigger the window size is, the smoother is the bias estimated (there are studies which talk about the optimum window size, interested readers can refer to [26]). One can also directly visually compare both images and see the difficulty of both methods when correcting this part of the bias (Fig. 5.7).

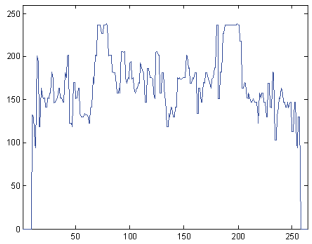
We have chosen the profiles in order to show us the behavior of the grey levels in a low-frequency morphological peculiarity. This zone is placed in the point where both horizontal and vertical profiles cross. Those zones can clearly be seen in the three horizontal profiles shown above, and they are a reference to understand the symmetry between the left and the right sides of the rat brain.



(a)

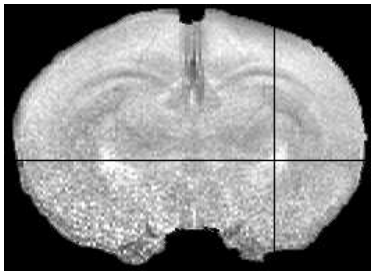


(b)

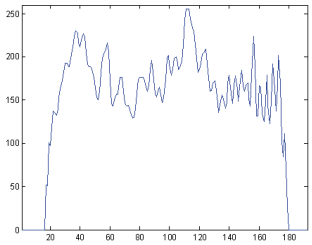


(c)

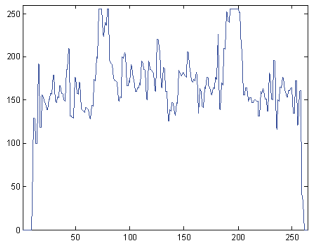
Figure 5.5: PABIC correction (a), vertical profile (b), horizontal profile (c).



(a)

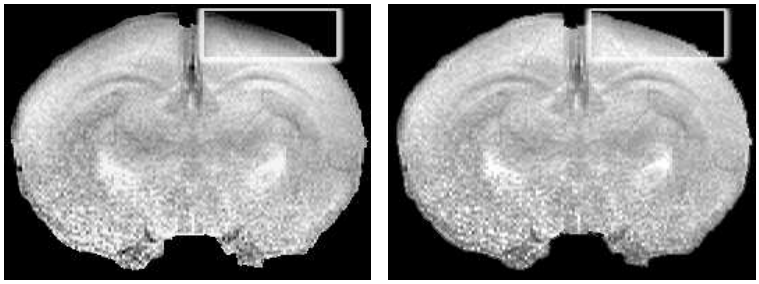


(b)



(c)

Figure 5.6: LPF correction (a), vertical profile (b), horizontal profile (c).



(a) (b)

Figure 5.7: PABIC correction (a), LPF correction (b). Marked zone of high-variation bias.

This symmetry is very helpful to determine the goodness of the correction: looking at Fig. 5.4(c) one can clearly see a slope, meanwhile after the corrections we see in 5.5(c) and 5.6(c) how both morphological peculiarities are clearly distinguished, as well as they present a perfect symmetry between both left and right part of the graphic. This reference is more difficult to obtain in the case of the vertical profile, but we can say that in both corrections the low frequency part of the variation seems to be quite flat along this axis.

5.3 Discussion

Now we have already deeply described both methods low-pass filtering and parametric bias field correction, as well as we have compared the results obtained. After all this study we have seen that both methods have advantages and pitfalls, which are listed below:

5.3.1 Advantages of the PABIC Method

- PABIC is a good method to deal with multi-class images, which is an important barrier when trying to correct the bias directly from the corrupt image, due to its segmentation parallel process.
- PABIC parametrizes the bias field, characterized by the $\underline{p}$ vector given as an output. This vector gives us the possibility to compare different bias fields corrected, or even to initialize the $\underline{p}$ coefficients of a similar correction in order to reduce the computation time.

5.3.2 Advantages of the LPF Method

- The LPF method proposed is much faster than the PABIC method. Our tests run on the same machine (Intel(R) Core(TM)2 CPU, 6600 @ 2.40GHz, 3.25 GB of RAM) and they determined that the delay for each one is (in average):
  - Less than 2 minutes in LPF computation, including the decision time of visually choosing both threholds.

- Around 20 minutes in PABIC computation (in this case we trust the goodness criterion applied by PABIC in order to finish the computation [2], as explained in 4.4).

- We have demonstrated that LPF can be more efficient in zones of high bias variation than PABIC method. It depends on the window size of the kernel used, which can be adjusted depending on the image we are working with (for example depending on the noise power of the image treated).

- The low-pass filter method proposed is much more predictable than the PABIC method, due to its simple implementation and its limited number of steps. The biggest problem of PABIC is the random possibility to fail in the correction. This is critical if we are talking about a three hours long correction. In PABIC, and due to its complexity, it is difficult to determine the optimum way to do the correction.

5.3.3 Other Points to Discuss

- They both rely on the interaction of the user. In the case of the PABIC the user shall provide the number of classes, as well as the mean and the sigma for each one of them (although we have demonstrated that it is not very sensitive to both in the case of single-class images). In the case of the LPF the user must provide two thresholds during the processing (this decision is very simple for the user, and it has a big error margin).

- Both of them are able to extrapolate the bias all over the image. This means that he region corrected can be bigger than the region measured to estimate the bias. This very useful in order to give a natural look to the corrected image (this way we can level parts of the image such as the skull and the flesh with the brain grey levels).

- None of them can completely ensure the success in bias extraction. They both can be useful depending on the kind of image we are dealing with. LPF fits well in *2-D* rat brain images acquired with surface coil (especially because its much lower computation time). Meanwhile, PABIC is a complete method and we have seen how in spite of its large computation time it can treat multi-class *3-D* brain images.

5.4 Future Work

After all this study about the intensity inhomogeneity in MRI, we have seen that the *perfect solution* does not exist. But some of the results obtained in this report makes us think about other ways to treat the bias problem.

One of them is to take advantage of the information that can be obtained from the histogram of the image, studied in 2. We have seen how the histogram reacts in different ways to the bias behavior. This additional information, after a deep study on this matter, could be used in bias correction.

[2] Maximum number of iterations set at 50,000, 100,000, 150,000 and 200,000. All of them took $20min \pm 1min$. Note also that both images compared are the ones shown in section 5.2.

Another point is to improve the programming feature in the LPF method proposed in Chapter 3. The points that must be improved are the following:

- Extend the method to three dimensions. A *3-D* mirroring can be easily performed by computing a *3-D* gadient and a *3-D* distance map.
- Study multi-class images behavior and their compatibility with the method proposed.

Such features are not computationally complicated, and thus it can be foreseen that the computation time after this *3-D* extension is going to be beneath PABIC's *3-D* computation time (which is of the order of hours, and it is one of the biggest pitfalls of this latter method).

Finally, and taking profit of having access at an MR system, we shall go deeper into physics and study carefully the forth type of correction seen in Chapter 2: *acquire extra information during the imaging process to aid in removing the bias field*. This correction, if well used, could be more consistent than the other ones (we have observed during all this report that trying to remove the bias directly from the corrupt image has serious limitations). In addition, both previous estimation and post-correction could be combined, making the correction method more robust. We shall also test how physical parameters of the acquisition (such as the flipping angle or the SNR) affects the bias.

In addition, we have seen how the parametric bias correction is a useful tool when comparing different bias. Therefore this feature will be a useful utility in future work.

List of Figures

Bibliography

[1] P. A. Mason, et al., *Database created from magnetic resonance images of a Sprague-Dawley rat, rhesus monkey, and pigmy goat*, 1995.

[2] Martin Styner, et al., *Parametric Estimate of Intensity Inhomogeneities Applied to MRI*, March, 2000.

[3] Rémi Charrier, et al., *Bias Field Correction in MR Images*, June, 2006.

[4] Zhi-Pei Liang, Paul C. Lauterbur, *Principles of Magnetic Resonance Imaging, a Signal Processing Perspective*, October 18, 1999.

[5] Benjamin H. Brinkmann, et al., *Optimized Homomorphic Unsharp Masking for MR Grayscale Inhomogeneity Correction.*

[6] F. Lin, et al., *A wavelet-based approximation of surface coil sensitivity profiles for correction of image intensity inhomogeneity and parallel imaging reconstruction*, Human Brain Mapping, vol. 19, no. 2, pp. 96-111, 2003.

[7] S. Geman, D. Geman, *Stochastic relaxation, Gibbs distributions, and Bayesian restoration of images*, 1984.

[8] Ayres Fan, *A Variational Approach to MR Bias Correction.*

[9] James B. Arnold, et al., *Qualitative and Quantitative Evaluation of Six Algorithms for Correcting Intensity Nonuniformity Effects*, 2000.

[10] Centre d'Imagerie Biomedicale (CIBM), Lausanne, Switzerland, *http://www.cibm.ch/*, Director: Prof. Rolf Gruetter, EPFL.

[11] Joseph P. Hornak, *The Basics of NMR*, 1997.

[12] McRobbie D., et al., *MRI, From picture to proton*, 2003.

[13] Hashemi Ray, et al., *MRI, The Basics 2ED*, 2004.

[14] Stanislav Skora,, *K-space formulation of MRI - http://www.ebyte.it/library/educards/mri/K-SpaceMRI.html*, March, 2005.

[15] Michael Thompson, *MRI Class Notes - http://www.phys.cwru.edu/courses/p431/notes-2003/node1.html*, 2003.

[16] Paul Summers, et al., *Impact of Image Processing Operations on MR Noise Distributions*, 2000.

[17] *Simulated Brainweb Database, McConnell Brain Imaging Centre of the Montral Neourological Institute.*

[18] Zujun Hou, *A Review on MR Image Intensity Inhomogeneity Correction*, October 11, 2005.

[19] Uros Vovk, et al., A Review of Methods for Correction of Intensity Inhomogeneity in MRI, March, 2007.

[20] Rehm, et al., *Putting our heads together: a consensus approach to brain/non-brain segmentation in T1-weighted MR volumes*, 2004.

[21] Martin Styner, Guido Gerig, *Evaluation of 2D/3D Bias Correction With 1+1ES-optimization*, 1997.

[22] Ing. Juan Sacerdoti, *POLINOMIOS Y FUNCIONES DE LEGENDRE, Notas para los alumnos del curso de Ecuaciones Diferenciales*, 2002.

[23] Developed by Martin Styner, et al., *Univ. of North Carolina at Chapel Hill.*

[24] *National Library of Medicine Insight Segmentation and Registration Toolkit (ITK) - http://itk.org/.*

[25] Luis Ibáñez, et al., *The ITK Software Guide*, 2005.

[26] Benjamin H. Brinkmann, et al., *Quantitative analysis of statistical methods of grayscale inhomogeneity corection in magnetic resonance images.*

[27] National Library of Medicine, *Visible Human Project, http://www.nlm.nih.gov/research/visible/visible_human.html.*