Convex Formulation of Region-based Image Registration

Shima Sepehri

Assistants : Subrahmanyam Venkata Ravi Mohana Sai Gorthi
Dr. Meritxell Bach Cuadra
Director : Prof. Dr. Jean-Philippe Thiran

Signal Processing Laboratory (LTS5),
École Polytechnique Fédérale de Lausanne (EPFL)

January 2011

to my family

Abstract

Image registration is a key component in medical imaging, due to its widespread applications like quantitative analysis, medical diagnosis and treatment.

Registration can be performed based on various characteristics of the images, like pixel-based and region-based characteristics. In this thesis, we study the problem of region-based registration.

The original formulation of region-based registration is nonconvex, in nature. It means the final registration results are dependant on the initialization and other parameters.

In this work, we propose a convex formulation of the 'Region-based Image Registration'. The proposed formulation has been implemented and evaluated on 2D synthetic images.

Acknowledgment

I would like to appreciate my thesis director, Professor Jean-Philippe Thiran, for all his guidance, advice, help, support and good-self. Moreover, I would like to appreciate my assistants, Mr. Gorthi for useful discussions, time allocation and support, and Dr. Bach Cuadra for her assistance and making my work easy. Finally, many thanks are due to Xavier Bresson for his clarifying emails, and also to laboratory assistants, for their help.

Contents

Chapter 1

Introduction

Medical Imaging is a necessary component of many applications, such as clinical diagnosis, event tracking and decision making. Segmentation of anatomical organs in a medical image is a helpful processing of medical images for clinical purposes. Accurate manual segmentation requires considerable efforts and time, due to some challenges such as the invisibility of some organs in an image, for example. Here, image processing techniques are used to facilitate and improve the accuracy of such a segmentation.

Different methods have been introduced in the literature to fulfill image segmentation with applications to medical image segmentation. Introducing some constraints in the formulation, based on the image, is a common way to reach a better segmentation. An atlas, which incorporates the locations and shapes of anatomical structures and their spacial relationship, is a complete example of neighborhood relations constraints. In atlas-based segmentation, an accurate manual segmentation -in the atlas- is transformed to a segmentation of a new patient's image. In this sense, the segmentation problem is reduced to a registration problem which compensates for the usual differences between the atlas and the new image, such as anatomical differences.

However, finding the best match in the target image is always facing different obstacles. One of these problems can be finding a local minimum -a better match than some others- rather than a global minimum -the best match. Therefore, the available literature on finding a global solution to energy minimization problems is used to find a global solution for atlas-based image registration problem. Despite the advantages of such global solutions (convex methods), they require more complicated calculations which means great, and sometimes impractical, computational capacity.

Image registration is the process of overlaying different images. It consists of finding a transformation which maps the points in an image into corresponding points in other images. Medical imaging uses this concept to

overcome some of its problems, improve some of its features or satisfy some of its needs. For example, image registration can be applied to monitor same modalities in one patient (monitoring and quantifying the progression of a disease over time, evaluation of intra-operative brain deformation, etc). It is also applicable to different modalities in one patient (correction of a patient's position change between scans, etc) or to the same modalities on different patients (atlas construction, subject's variability studies, etc).

To clarify the essentiality of the usage of this concept in medical imaging, one can consider 'atlas-based image registration' which was also the initial motivation of this work: Computed Tomography (CT) Scans are becoming quite popular as a diagnostic tool, both for screening of a disease or for preventive medicine. As the Greek word tomo -meaning slice- suggests, in this method, a single slice of the body is represented on the radiographic film and a series of slices at different depths and with different thicknesses is usually output for diagnostic usages. For instance, cranial slices are more useful for some patients, however, special organs of interest are not visible in those slices. It will take several hours for a physician to segment those regions of interests, using other slices and anatomical knowledge. Therefore, automated segmentation of medical images is a necessity. Atlas-based segmentation is a widely used technique for such a purpose (See [3] for a survey). We can have a very accurate atlas in medical image applications. Therefore, atlas-based image registration might be very interesting for segmentation of medical images. In this segmentation technique, a reference image (called atlas) has been accurately segmented, usually by hand. To segment a new image (called target image), a dense deformation field which puts the atlas into a point-to-point spatial correspondence with the target image is computed. This transformation is then used to map the accurate manual segmentation of the atlas onto the new image. Since this dense deformation field is interpolated on the whole image from the registration of the visible features, it allows to easily segment non visible structures, as well. This segmentation can be very accurate [4]. The accuracy of the segmentation depends on the accuracy of the registration. For an accurate registration deformations should be compensated.Deformations such as patient's movements, which can be modeled, are easy to compensate. However, deformations like anatomical changes are more difficult to compensate.

Generally, image registration techniques optimize some global similarity measures (such as mutual information, sum of squared differences or cross-correlation) coming from the atlas intensity image. Therefore, these methods are prone to get stuck in local minima, during the optimization process. The energy term, which these methods usually try to minimize, become larger by going out of the local minimum in any direction and hence the algorithms

might result in the local minimum, as the final answer, instead of the correct answer -the global minimum. This problem can become even more important in medical images, due to the anatomical similarities, which increases the probability of finding a local minimum. Moreover, the importance of medical issues requires that all precautions are applied to find the result, as accurate as possible.

On the other hand, due to the symmetry of anatomical organs, and their particular shapes for different patient, special care should be taken into account. Otherwise, a global solution algorithm might find a wrong feature as the best match, because, for example, the right organ in the atlas is more similar to the left one, in the target image.

Hence, we tried to apply the available literature on 'convex relaxation methods' to 'image registration' problem. To do so, first, a full understanding of the advanced mathematics and techniques of available convex relaxation methods ([5, 6]), using different references ([7, 8, 9, 10]) and other materials ([11] to [20]), were required. Afterwards, these techniques were matched with the image registration problem. This was the most challenging part of this project, where many problems arose and several reviews, discussions and inspections were done, as well as ubiquitous programming debugging, in order to fulfill the goal.

The rest of this work is structured as follows. Next chapter will be a short review of the previous work and state of the art. In chapter 3, the convex relaxation method is described in the context of image registration. In chapter 4 the results of our convex image registration is shown. Finally, chapter 5 summarizes and concludes the work.

Chapter 2

Previous Work

Different problems in image processing are proposed as regularized minimization problems. Some examples are image registration [1, 21], segmentation [22, 23, 24], de-convolution [25, 26] and denoising [27, 28, 24]. When these problems are convex[1], standard minimization methods, for example using simple variational techniques (gradient descent [27], dual formulations [29, 30, 31] or split Bregman schemes [32, 33, 19]), yield reliable results. As explained in chapter 1, when the problem is nonconvex - like image registration - these techniques fail since they get stuck at the local minima.

One of the first methods to handle nonconvex problems is the LP relaxation technique, in which a 0-1 integer problem is replaced by using the weaker constraint that each variable belongs to the interval [0,1]. This results in a linear problem the solution of which can reveal information about the original problem.

[7] introduced a relaxation method, for global two phase image segmentation, partly based on the results of [9], on maximal flows. However, for complicated problems, the first step before convex relaxation is reformulation of the problem.

Such nonconvex problems can be reformulated as convex problems, using "functional lifting ([34], in MRF setting)" concept in optimization theory. These equivalent convex problems can, then, be easily minimized using standard techniques. The original function can be either scalar or vector valued. In segmentation problems, for instance, the unknown is a scalar showing which group the pixel is belonging to. Similarly, the stereo matching problems only involve image displacements in one direction. However, for a registration problem the unknown is a vector corresponding to the deformation fields in the horizontal and vertical directions (see chapter 3). At

[1]For a description of convex and nonconvex problems, refer to section 3.2, on page 11.

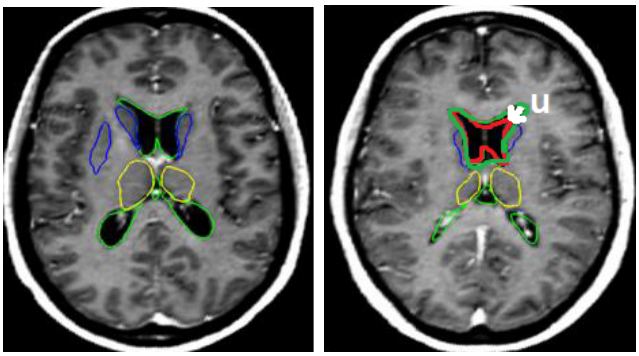
the beginning, the functional lifting concept was only applied for minimizing scalar-valued functions, within both MRF frameworks [34, 7] and analogously in the continuous domain [22, 14]. Afterwards, works such as [6, 5] generalizes this concept for a class of vector-valued problems.

Chapter 3

Convex Formulation of Image Registration

The implemented scheme is based on implementing the convex relaxation method, presented in [5] to the image registration problem.

Consider the two atlas and target images, as shown in Figure 3.1.



(a) atlas (b) target

Figure 3.1: Segmented atlas and target images. One of the segmented organs in the atlas image is shown on the target image in red to visualize the deformation field of a point ($\mathbf{u}$).

If $\mathbf{u}(x, y)$ is found correctly, the segmented target image can be obtained, by substituting, at position $(x + u_1(x, y), y + u_2(x, y))$, the intensity of the segmented atlas image at (x, y) -where (x, y) is the pixel position and u_1 *and* u_2 are the x and y components of $\mathbf{u}$.

To begin, we want to formulate the atlas-based registration model for 2-phase piecewise constant Mumford-Shah segmentation model. More specifically, we are expressing the energy functional associated with the above

model with "deformation field vector" as the varying parameter.

3.1 Two-Phase Piecewise Constant Atlas-based Registration

Our goal is implementing a convex relaxation technique [5] to piecewise constant atlas-based registration, the formula of which was informally available (see Primary Formulation). These formulas are in the continuous domain. However, delicate attention reveals that they should be changed to exactly match the correct position of pixels in the fixed and moving images (see Final Formulation). This correction was only resulted after unsuccessful implementations, careful search for the reasons and rewriting the formulation for a discrete toy example, as mentioned in Final Formulation.

3.1.1 Notation

Let $\mathbf{u}(\mathrm{x})$: $\mathbb{R}^n \to \mathbb{R}^n$ (n = 2 or 3) represent the deformation field vector. Let the atlas image be the fixed image, and the target image be the moving image. . Let $I_M(\mathrm{x})$: $\Omega^M \subset \mathbb{R}^n \to \mathbb{R}^n$ and $I_F(\mathrm{x})$: $\Omega^F \subset \mathbb{R}^n \to \mathbb{R}^n$ be the intensities of the moving and fixed images, respectively. Let c_0 and c_1 be respectively the mean intensity values inside and outside the contour in the fixed (atlas) image; these values are fixed for a given atlas.

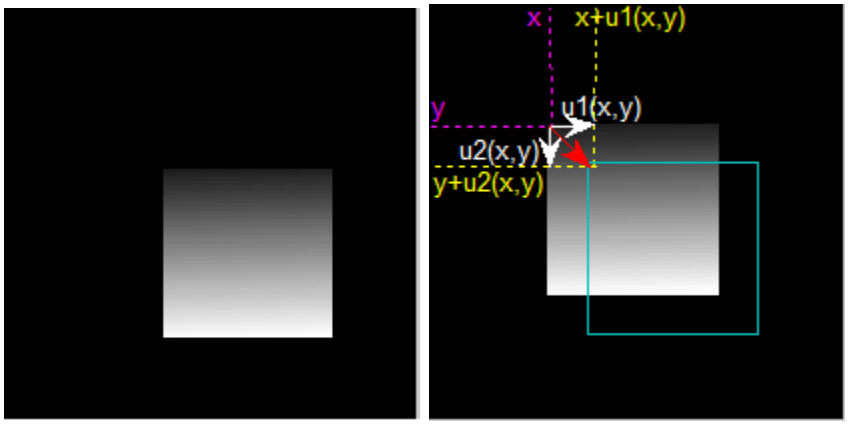
3.1.2 Energy Formulation

Primary Formulation

As described in [35], and by considering the mean intensities on each phase of the image (as in [8]) (here, inside and outside) to be constant, the energy to be minimized seemed to be[1]:

$$\begin{aligned} &\iint_{\Omega^F} \left(I_M\left(\vec{x}+\vec{u}\left(\vec{x}\right)\right)-c_0\right)^2 H\left(\Phi_F\left(\vec{x}\right)\right) \mathrm{d}\vec{x} \\ &+\iint_{\Omega^F} \left(I_M\left(\vec{x}+\vec{u}\left(\vec{x}\right)\right)-c_1\right)^2 \left(1-H\left(\Phi_F\left(\vec{x}\right)\right)\right) \mathrm{d}\vec{x} \\ &+\mu \iint_{\Omega^F} \left|\nabla \vec{u}\left(\vec{x}\right)\right| \mathrm{d}\vec{x} \end{aligned} \tag{3.1}$$

[1]The corrected formulas, with explanations can be found in Final Formulation on page 9.



(a) atlas (b) target

Figure 3.2: Toy atlas (fixed) and target (moving) images, where the domain of the moving image is covered, in the formulations. Therefore, the fixed image is fused on the moving image, on the right.

Let

$$\begin{aligned} g_i\left(\vec{x}+\vec{u}\left(\vec{x}\right)\right) &= \left(\left(I_M\left(\vec{x}+\vec{u}\left(\vec{x}\right)\right)-c_0\right)^2, \text{for i=0,1}\right., \\ \theta_M\left(\vec{x}+\vec{u}\left(\vec{x}\right)\right) &= H\left(\Phi_M\left(\vec{x}+\vec{u}\left(\vec{x}\right)\right)\right), \\ \theta_F\left(\vec{x}\right) &= H\left(\Phi_F\left(\vec{x}\right)\right). \end{aligned} \tag{3.2}$$

Then the energy minimization problem can be written as follows:

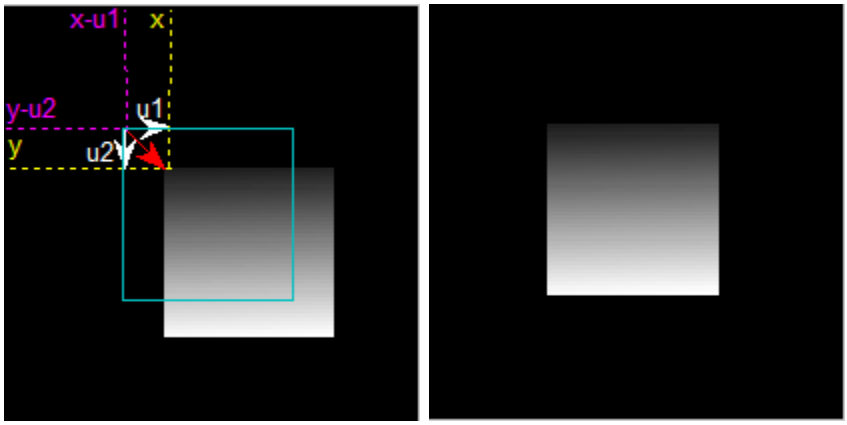
$$\begin{aligned} \min_{\vec{u}(\vec{x})} \Bigg\{ E\left(u\right) := & \iint_{\Omega^F} g_0\left(\vec{x}+\vec{u}\left(\vec{x}\right)\right)\theta_F\left(\vec{x}\right) \\ & + g_1\left(\vec{x}+\vec{u}\left(\vec{x}\right)\right)\left(1-\theta_F\left(\vec{x}\right)\right)\mathrm{d}\vec{x} + \mu \iint_{\Omega^F} \left|\nabla\vec{u}\left(\vec{x}\right)\right|\mathrm{d}\vec{x} \Bigg\} \end{aligned} \tag{3.3}$$

Final Formulation

In this part, the correction suggested to equation 3.1 is expalined. Consider the atlas and the target images, as shown in Figure 3.2.

Note that the deformation field, $\vec{u}$ is defined as the deformation from the moving image to the atlas image, as shown in Figure 3.2. Hence, to the best of the author's knowledge, the following changes are required to the equation 3.1:

1. If the integrals are on the domain of the fixed image (Ω_F), as shown in Figure 3.3, for a point (x, y) inside the object of the fixed image, the



(a) atlas (b) target

Figure 3.3: Toy atlas (fixed) and target (moving) images, where the domain of the fixed image is covered, in the formulations. Therefore, the moving image is fused on the fixed image, on the left.

intensity of the moving image should be evaluated at position $(\vec{x} - \vec{u})$. The problem is that, this $\vec{u}$ is in fact $\vec{u}(\vec{x} - \vec{u})$, and the same for this $\vec{u}$, the argument of the previous $\vec{u}$. So, we should have something like:

$$\begin{aligned}
&\iint_{\Omega^F} \left(I_M\left(\vec{x} - \vec{u}\left(\vec{x} - \vec{u}\left(\vec{x} - \cdots\right)\right)\right) - c_0\right)^2 H\left(\Phi_F\left(\vec{x}\right)\right) \mathrm{d}\vec{x} \\
&+ \iint_{\Omega^F} \left(I_M\left(\vec{x} - \vec{u}\left(\vec{x} - \vec{u}\left(\vec{x} - \cdots\right)\right)\right) - c_1\right)^2 \left(1 - H\left(\Phi_F\left(\vec{x}\right)\right)\right) \mathrm{d}\vec{x} \\
&+ \mu \iint_{\Omega^F} \left|\nabla \vec{u}\left(\vec{x} - \vec{u}\left(\vec{x} - \vec{u}\left(\vec{x} - \cdots\right)\right)\right)\right| \mathrm{d}\vec{x}
\end{aligned}$$

So, the author suggests to have the integrals on the domain of the moving image (Ω_M).

2. However, since the inside and outside regions are unknown in the target (moving) image, the mean intensities should be evaluated according to the fixed image. Therefore, we still need the heaviside functions of the domain of the fixed image. Nevertheless, the arguments should be changed, because the integrals are over the domain of the moving image. So, in order to go to the domain of the fixed image, for a point (x, y) inside the object on the moving image, as in Figure 3.2, $H\left(\phi_F\right)$ should be calculated at $\phi_F\left(\vec{x} + \vec{u}\left(x\right)\right)$.

Considering the above-mentioned changes, the following energy formulation -for minimization- is suggested:

$$\begin{aligned}&\iint_{\Omega^M} \left(I_M\left(\vec{x}\right) - c_0\right)^2 H\left(\Phi_F\left(\vec{x} + \vec{u}\left(\vec{x}\right)\right)\right) \mathrm{d}\vec{x} \\ &+ \iint_{\Omega^M} \left(I_M\left(\vec{x}\right) - c_1\right)^2 \left(1 - H\left(\Phi_F\left(\vec{x} + \vec{u}\left(\vec{x}\right)\right)\right)\right) \mathrm{d}\vec{x} \\ &+ \mu \iint_{\Omega^M} \left|\nabla \vec{u}\left(\vec{x}\right)\right| \mathrm{d}\vec{x}\end{aligned} \tag{3.4}$$

Considering the convention described in 3.2, the energy minimization problem can be written as follows:

$$\begin{aligned}\min_{\vec{u}(\vec{x})} \Bigg\{ E\left(u\right) := &\iint_{\Omega^M} g_0\left(\vec{x}\right) \theta_F\left(\vec{x} + \vec{u}\left(\vec{x}\right)\right) \\ &+ g_1\left(\vec{x}\right) \left(1 - \theta_F\left(\vec{x} + \vec{u}\left(\vec{x}\right)\right)\right) \mathrm{d}\vec{x} + \mu \iint_{\Omega^M} \left|\nabla \vec{u}\left(\vec{x}\right)\right| \mathrm{d}\vec{x} \Bigg\}\end{aligned} \tag{3.5}$$

3.2 Convex Relaxation Method

The energy function of the equation 3.5 is a member of a class of vector-valued minimization problems described in [5] and can be represented as:

$$\min_{\vec{u}} E\left(\vec{u}\right) : \sum_{i=1}^{m} \int_{\Omega} \left|\nabla u_i\right| + \int_{\Omega} \rho\left(x, \vec{u}\left(x\right)\right) \mathrm{d}x \tag{3.6}$$

with

$$\begin{aligned} m &= 2\,, \\ x \text{ denotes } \vec{x} &= (x, y)\,: \text{ pixel position vector}\,, \\ \rho\left(x, \vec{u}\left(x\right)\right) &= g_0\left(x\right) \theta_F\left(x + \vec{u}\left(x\right)\right) + g_1\left(x\right) \left(1 - \theta_F\left(x + \vec{u}\left(x\right)\right)\right). \end{aligned}$$

It is assumed that the function $\rho : \Omega \times \mathbb{R}^m \to \mathbb{R}$ is bounded from below, so that without loss of generality it may be assumed that ρ is non-negative by adding a constant to E if necessary. However, no other assumptions are made on ρ; in particular, ρ may be nonconvex, In fact, it is nonconvex in a wide variety of problems arising in image processing and computer vision. Hence, we have a minimization problem in which both the minimization function and the set to which the minimization variable ($\mathbf{u}$) belongs are, in general, nonconvex.

To clarify the essence and the goals of the presented convex relation method, let us consider convex functions and convex sets, in short. Pictorially, a function is called 'convex', on an interval, if the function lies below or on the straight line segment connecting two points, for *any* two points in that interval, as shown in 3.4.

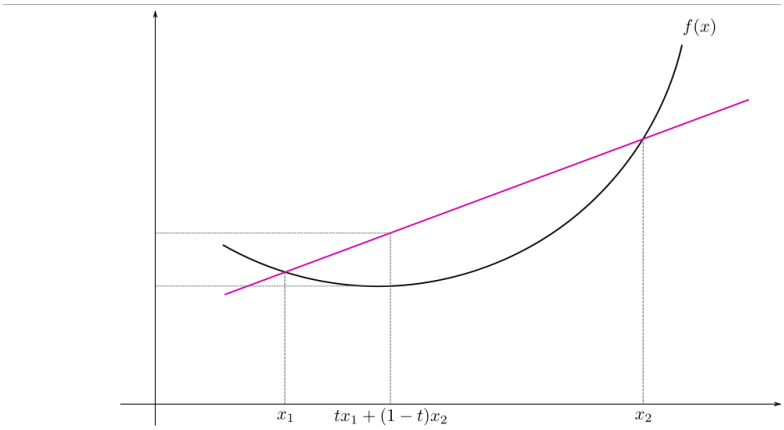
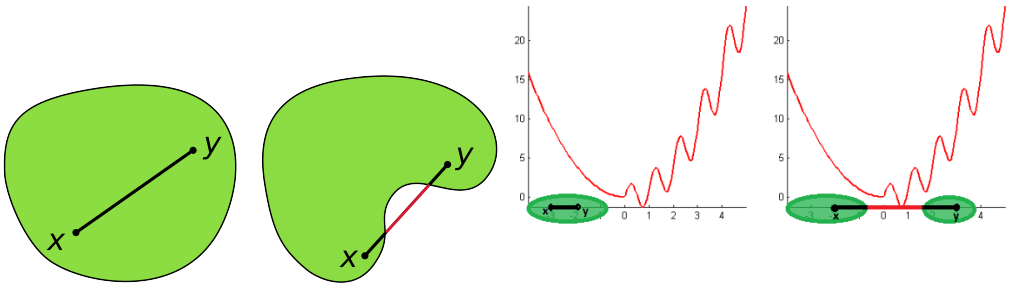


Figure 3.4: Convex function on an interval.

It should be clear that in a convex function every local minimum is a global minimum[1]. Moreover, an object (ex. a set or an interval) is 'convex' if for every pair of points within the object, every point on the straight line segment that joins them is also within the object. Examples of convex and nonconvex sets and intervals are depicted in 3.5.



(a) a convex set (b) a nonconvex set (c) a convex interval (d) a nonconvex interval

Figure 3.5: Convex Objects.

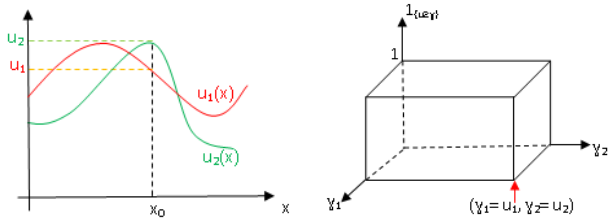
[1]Because, if the function has multiple ups and downs, there exists two points, for which, it does not lie under the straight line segment connecting those two points

It is observed that when a convex function is minimized over a convex set, every locally optimal solution is globally optimal. Therefore, it is desired to find an equivalent for the general nonconvex energy minimization of 3.6, where the function and the minimization set are both convex.

To describe the reformulation, some functions and operators are necessary. Firstly, the following function, which is a generalization of the *super-level set function* is defined in [6, 5]:

$$1_{\{\vec{u}(x)\succeq\vec{\gamma}\}} := 1_{\{u_1\geq\gamma_1,\cdots,u_m\geq\gamma_m\}} \begin{cases} 1 & \text{if } u_1 \geq \gamma_1, \cdots, u_m \geq \gamma_m \\ 0 & \text{otherwise} \end{cases} \tag{3.7}$$

This is called a box function. The reason is well explained, in Figure 3.6. As seen in 4.5b, if $\vec{u}$ has two components (like in our 2D registration case), for a particular $\vec{u} = (u_1, u_2)$, the function defined in Equation 3.7 will be a box with values 1, over the area where $\gamma_1 \leq u_1, \gamma_2 \leq u_2$, as shown in 3.6b.



(a) u(x) (b) box function

Figure 3.6: The box function created from the function u, with the Equation 3.7.

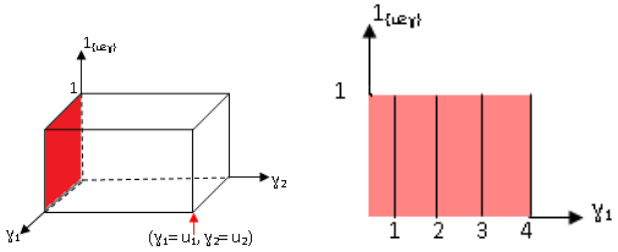
The goal of the introduction of Equation 3.7 is convex relaxation. In fact, as mentioned previously, the energy formulation of the Equation 3.6, might be nonconvex, due to the function $\rho\left(x, \vec{u}\left(x\right)\right)$. However, if we can convert it to $\rho\left(x, \vec{\gamma}\right) \times f\left(\vec{\gamma}, \vec{u}\right)$, the energy minimization might become convex. As done in the relaxation method of [5], the reformulation makes the energy functions of the form of Equation 3.6 convex.

There is a one-to-one correspondence between $\vec{u}$ and its associated box function, as:

$$u_i\left(x\right) = \sum_{\ell=1}^{N_i} 1_{\{\vec{u}(x)\succeq\vec{\gamma}\}}\left(x, l\vec{e_i}\right) \tag{3.8}$$

Where N_i is the maximum possible value of u_i and $\vec{e_i}$ is the ith standard basis.

This formula can be best understood from 4.5b. Consider u_1 in figure 4.5b to be, for example, 4 and the maximum possible value of u_1 to be 6. Moreover, as suggested by $1_{\{\vec{u}(x)\succeq\vec{\gamma}\}}\left(x, l\vec{e_1}\right)$, in Equation 3.8, consider a slice of the box function of figure 3.6b, for $\gamma_2 = 0$, as shown in 3.7a. It is then clear that $u_1 = 4 = \sum_{\ell=1}^{6} 1_{\{\vec{u}(x)\succeq\vec{\gamma}\}}\left(x, l\vec{e_1}\right) = 1+1+1+1+0+0$.



(a) a slice of the box function

(b) $1_{\{\vec{u}(x)\succeq\vec{\gamma}\}}\left(x, l\vec{e_1}\right)$

Figure 3.7: The reconstruction of u_1 from a slice of the box function, using Equation 3.8.

Besides the box function, an mth order difference operator is required for the convex reformulation. To deal with the boundary conditions, let us define the forward difference operators as:

$$\left(D_i\phi\right)\left(x,\vec{\gamma}\right) = \begin{cases} 0 & \text{if } \gamma_i = N_i + 1 \\ \phi\left(x,\vec{\gamma}+\vec{e_i}\right) - \phi\left(x,\vec{\gamma}\right) & \text{otherwise} \end{cases} \tag{3.9}$$

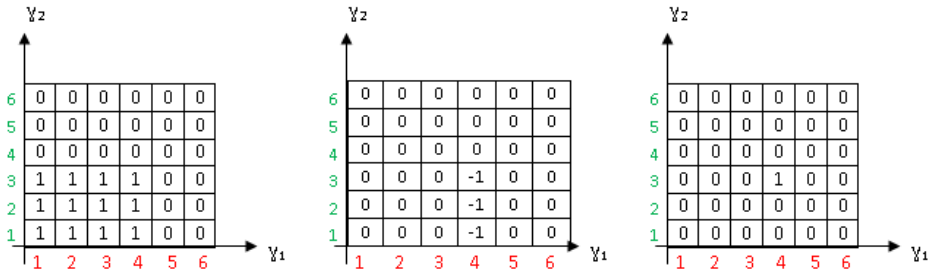
Then, the mth order mixed difference is defined as subsequent difference operators:

$$D^m_{1,\cdots,m} := D_m\left(D_{m-1}\left(\cdots\left(D_1\right)\right)\right) \tag{3.10}$$

This mixed difference operator is useful because of its following property:

$$\left(D^m_{1,\cdots,m} 1_{\{\vec{u}\succeq\vec{\gamma}\}}\right)\left(x,\vec{\gamma}\right) = \begin{cases} (-1)^m & \text{if } \vec{\gamma} = \vec{u} \\ 0 & \text{otherwise} \end{cases} \tag{3.11}$$

This is best illustrated in Figure 3.8, where the box function of figure 3.6b is depicted in 3.8a, for $u_1 = 4, u_2 = 3$ and $N_1 = N_2 = 6$. According to the definition of 3.10, to compute $\left(D^2_{1,2} 1_{\{\vec{u}\succeq\vec{\gamma}\}}\right)\left(x,\vec{\gamma}\right)$, we should first compute D_1 - the forward difference with respect to γ_1 - of the box function, as in 3.8b. Then D_2 - the forward difference with respect to γ_2 - is calculated from



(a) the box function (b) D1, from the box function (c) D2, from D1

Figure 3.8: $D^2_{1,2}$, calculated from the box function (a), in 2 steps, according to the definition of Equation 3.10.

the previously calculated D_1, as shown in 3.8c. As seen here, $D^2_{1,2}$ satisfies Equation 3.11 for $m = 2$.

Hence, using the property of the mth order difference of Equation 3.11, $\rho(x, \vec{u}(x))$ can be convert to $\rho(x, \vec{\gamma}) \times f(\vec{\gamma}, \vec{u})$, as we aimed:

$$\rho(x, \vec{u}(x)) = (-1)^m \sum_{\vec{\gamma} \in \Gamma} \rho(x, \vec{\gamma}) D^m_{1,\cdots,m} 1_{\{\vec{u} \succeq \vec{\gamma}\}} \tag{3.12}$$

where $\Gamma = \{0, 1, \cdots, N_1\} \times \cdots \times \{0, 1, \cdots, N_m\}$.
Furthermore, note that

$$1_{\{u_i \geq \gamma_i\}} = 1_{\{\vec{u} \succeq \vec{\gamma}\}}(x, \gamma_i \vec{e_i}) \tag{3.13}$$

Therefore, using the linearity of the gradient operator, we have:

$$\int_\Omega |\nabla u_i| \mathrm{d}x = \sum_{\ell=1}^{N_i} \int_\Omega |\nabla 1_{\{\vec{u} \succeq \vec{\gamma}\}}| \mathrm{d}x \tag{3.14}$$

Using Equation 3.12 and Equation 3.14, and by the convention of $\phi = 1_{\{\vec{u} \succeq \vec{\gamma}\}}$, the energy minimization of Equation 3.6 can be written as:

$$\min_\phi F(\phi) := \sum_{i=1}^{m} \sum_{\ell=1}^{N_i} \int_\Omega |\nabla \phi(x, l\vec{e_i})| \mathrm{d}x + (-1)^m \sum_{\vec{\gamma} \in \Gamma} \int_\Omega \rho(x, \vec{\gamma}) D^m_{1,\cdots,m} 1_{\{\vec{u} \succeq \vec{\gamma}\}} \tag{3.15}$$

While the energy function of Equation 3.6 might be nonconvex in $\vec{u}$ (due to ρ), this energy function is convex in ϕ. However, the minimization is still

conducted on the nonconvex set of box functions. Therefore, [5] introduces a condition on the set of box functions and consider the convex set

$$X = \left\{ \phi \in C : (-1)^m D^m_{1,\cdots,m} \geq 0 \right\} \tag{3.16}$$

Where the set C is used to deal with the boundary conditions:

$$\begin{aligned} C = \{ \phi : \Omega \times \tilde{\Gamma} \to [0,1] : \phi(x, \vec{0}) = 1 \qquad & \text{and} \\ \phi(x, \vec{\gamma}) = 0 \quad & \text{whenever } \gamma_i = N_i + 1 \text{ for some i} \} \end{aligned}$$

With $\tilde{\Gamma} := \{0, 1, \cdots, N_1 + 1\} \times \cdots \times \{0, 1, \cdots, N_i + 1\}$.

Therefore, the goal of convex minimization problem (in which both the energy function and the minimization set are convex) is achieved and the convex problem is

$$\min_{\phi \in X} F(\phi) \tag{3.17}$$

[5] also provides the certificate when the solution of the convex minimization problem of Equation 3.17 is equivalent to the solution of the nonconvex problem of Equation 3.6.

3.2.1 Algorithm

[5] derived an algorithm to solve the discrete version of Equation 3.17. In this algorithm, the following iterative formulas should be solved:

$$\begin{cases} (\vec{p})^{n+1} & = \Pi_{D^h} \left((\vec{p})^n + \tau_p A \left(\bar{\phi}^h \right)^n \right) \\ \left(\phi^h \right)^{n+1} & = \Pi_{C^h} \left(\left(\phi^h \right)^n + \tau_\phi A^* (\vec{p})^{n+1} \right) \\ \left(\bar{\phi}^h \right)^{n+1} & = 2 \left(\phi^h \right)^{n+1} - \left(\phi^h \right)^n \end{cases} \tag{3.18}$$

With any initial values $\left(\left(\phi^h \right)^0, (\vec{p})^0 \right) \in C^h \times D^h$ and put $\left(\bar{\phi}^h \right)^0 = \left(\phi^h \right)^0$.

Where

- $\vec{p} = (\vec{p_1}, \cdots, \vec{p_m}, p_\gamma)$ are the dual variables appears during the change of Equation 3.17 to a saddle point problem [5, Section 3].
- n is the number of iteration.
- $\Pi_{D^h} = \left(\frac{p_1}{\max(|(p_1, \cdots, p_m)|, 1)}, \cdots, \frac{p_m}{\max(|(p_1, \cdots, p_m)|, 1)}, \hat{p_\gamma} \right)$,
  with $\hat{p_\gamma} = \begin{cases} \max(p_\gamma, -\rho) & \text{if } m \text{ is odd,} \\ \min(p_\gamma, \rho) & \text{if } m \text{ is even.} \end{cases}$
  and ρ is the upper bound of p_γ in the minimization set.

- $\Pi_{C^h}\left(\phi^h\right)$ is a simple truncation of ϕ^h to the interval $[0,1]$, and setting the boundary conditions $\phi^h(x,\vec{0})=1$ and $\phi^h\left(x,\vec{\gamma}\right)=1$, if $\gamma_i=N_i+1$.

- τ_p and τ_ϕ are time steps, which should be chosen such that $\tau_\phi\tau_p\|A\|^2<1$ with $\|A\|^2=\frac{4}{h_x^2}+\frac{4}{h_y^2}+\frac{4^m}{\Pi_{i=1}^m h_{\gamma_i}^2}$ (h_x and h_y are spatial step sizes and $h_{\gamma_1},\cdots,h_{\gamma_m}$ are the step sizes of $\tilde{\Gamma}$, all can be considered 1, as in our case).

- A is the linear operator that maps

$$\phi^h\mapsto\left(\nabla^h\phi^h,\cdots,\nabla^h\phi^h,D^m_{1,\cdots,m}\right)\in\mathbb{R}^{2m+1} \tag{3.19}$$

  and A^* is the adjoint of A.

- ϕ^h is a discrete and relaxed version of our box function ϕ. Recall that $\phi:\Omega\times\Gamma\to\{0,1\}$. Therefore, $\phi^h:\Omega^h\times\tilde{\Gamma}\to[0,1]$. Note the differences: $\Omega^h=\{0,\cdots,N_x\}\times\{0,\cdots,N_y\}$ which is, in our case, the discrete image domain, $\tilde{\Gamma}$ which is the same as $\Gamma=\{0,1,\cdots,N_1\},\cdots,\{0,1,\cdots,N_m\}$, except that one is added to each N_i to deal with the boundary conditions. Finally, the binary $\{0,1\}$ codomain of ϕ is changed to the interval $[0,1]$ in ϕ^h.

- The convex sets C^h and D^h are defined as

$$\begin{aligned}C^h=\{&\phi^h:\Omega^h\times\tilde{\Gamma}\to[0,1]:\phi^h\left(x^h,\vec{0}\right)=1\text{ and}\\&\phi^h\left(x^h,\vec{\gamma}\right)=0\text{ whenever }\gamma_i=N_i+1\text{ for some i}\}\end{aligned} \tag{3.20}$$

  Where x^h is the discrete x.

$$\begin{aligned}D^h=\{\ \ &\vec{p}=(\vec{p_1},\cdots,\vec{p_m},p_\gamma):\Omega^h\times\tilde{\Gamma}\to\mathbb{R}^2\times\cdots\times\mathbb{R}^2\times\mathbb{R}=\mathbb{R}^{2m+1}:\\&|\vec{p_i}|\le1,(-1)^m p_\gamma\le\rho\text{ and }\vec{p_i}\left(x^h,\vec{\gamma}=0\right)\text{ if }\gamma_i\neq0\}\end{aligned} \tag{3.21}$$

As $n\to\infty$, the above mentioned scheme is guaranteed to converge to a solution of the min-max equivalent of 3.17 (see [5, p. 16] ,cf. [36]).

In practice, after a reasonable number of iterations, the function ϕ calculated as a result of the iterative scheme 3.18 is used to reconstruct the resultant deformation fields, using Equation 3.8.

Chapter 4

Results

In this chapter, first a convention used for "Convex Image Registration", described in chapter 3, is explained. The goal is to provide a scientific and detailed documentation to ease the following up of such a project. Finally, the results of the evaluation of our algorithm is presented and explained.

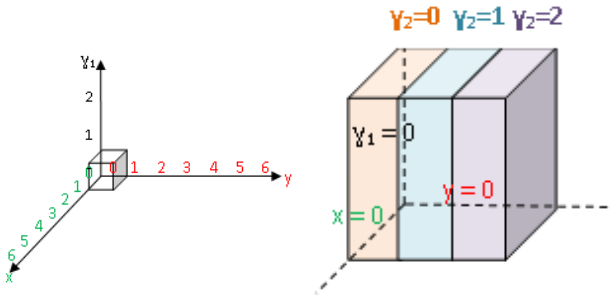
We were aware that the optimal solution, mentioned in section 3.2, was only achieved at the price of complexity. In fact, in this method, the difficulty of 'nonconvexity' has been transformed to the difficulty of 'dimensionality'. Therefore, without fast implementation techniques, the method might be impractical. Hence, we first invested and implemented the following technique, in order to reduce time allocations.

The variables of Equation 3.18 are 4D arrays (functions of the 2D space parameters and the 2D minimization parameters). Performing a reasonable number of iterations using 4D arrays will be time consuming. Hence, these arrays were changed to 1D arrays, using the following convention:

Consider the $x - y$ plane in Figure 4.1 to be the discrete image domain. Then, for showing the other two variables, γ_1 and γ_2 (of the functions mentioned in section 3.2) we use the following trick: γ_2 will be normally displayed as the third dimension, as in 4.1a. To add the forth variable, γ_2, let us consider a small cube of 4.1a, as shown in 4.1b. For this small cube, $(x = 0, y = 0, \gamma_1 = 0, 0 \leq \gamma_2 \leq 2)$, where the sub cubes show different values of γ_2.

Now, consider the following functions constructed from our four parameters:

$$\begin{aligned} x_{equivalent} &= ((x \times N_y) + y) \times N_\gamma \\ xy_{equivalent} &= (x_{equivalent} + \gamma_1) \times N_\gamma \\ xy\gamma_{equivalent} &= xy_{equivalent} + \gamma_2 \end{aligned} \tag{4.1}$$



(a) 3variables (b) 4th variable

Figure 4.1: Illustration of our four parameters $(x, y, \gamma_1, \gamma_2)$.

$\gamma_1 \backslash \gamma_2$	**0**	**1**	**2**
0	0	1	2
1	3	4	5
2	6	7	8

Table 4.1: $xy\gamma_{equivalent}$ for $x = 0$ and $y = 0$.

Where N_x, N_y and N_γ are the maximum values of x, y and γ respectively.

These values can be easily calculated using the attached MATLAB file, and is depicted in Figure 4.2, for the example of Figure 4.1.

x_equi =

0	3	6	9	12	15	18
21	24	27	30	33	36	39
42	45	48	51	54	57	60
63	66	69	72	75	78	81
84	87	90	93	96	99	102
105	108	111	114	117	120	123
126	129	132	135	138	141	144

(a) $x_{equivalent}$

xy_equi =

0	9	18	27	36	45	54
63	72	81	90	99	108	117
126	135	144	153	162	171	180
189	198	207	216	225	234	243
252	261	270	279	288	297	306
315	324	333	342	351	360	369
378	387	396	405	414	423	432

(b) $xy_{equivalent}$

xyg_equi =

0	9	18	27	36	45	54
63	72	81	90	99	108	117
126	135	144	153	162	171	180
189	198	207	216	225	234	243
252	261	270	279	288	297	306
315	324	333	342	351	360	369
378	387	396	405	414	423	432

(c) $xy\gamma_{equivalent}$

Figure 4.2: $x_{equivalent}$, $xy_{equivalent}$ and $xy\gamma_{equivalent}$ of Equation 4.1, calculated for $\gamma_1 = 0$ and $\gamma_2 = 0$.

Now, consider the values of $xy\gamma_{equivalent}$, for the cube shown in Figure 4.1, which means for $x = 0$ and $y = 0$. These values are listed in Table 4.2, for different possible values of γ_1 and γ_2.

In fact, the indices of a 1D array is assigned, in a special order, to all the possible combinations of our original 4D arrays. Table 4.2 and the values of $xy\gamma_{equivalent}$ shown in figure 4.2b reveals this order. More precisely, the indices of the 1D array starts from 0 and increses as the arrows of Figure 4.3 show. This means that, for every pixel, (x, y), the indices are assigned for

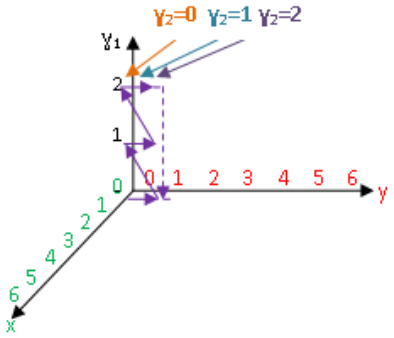


Figure 4.3: The index assignment of the equivalent 1D array

4D expression	**1D equivalent**
$\phi(x+1, y, \gamma_1, \gamma_2)$	$\phi\left(xy\gamma_{equivalent} + N_\gamma^2 N_y\right)$
$\phi(x-1, y, \gamma_1, \gamma_2)$	$\phi\left(xy\gamma_{equivalent} - N_\gamma^2 N_y\right)$
$\phi(x, y+1, \gamma_1, \gamma_2)$	$\phi\left(xy\gamma_{equivalent} + N_\gamma^2\right)$
$\phi(x, y-1, \gamma_1, \gamma_2)$	$\phi\left(xy\gamma_{equivalent} - N_\gamma^2\right)$
$\phi(x, y, \gamma_1+1, \gamma_2)$	$\phi\left(xy\gamma_{equivalent} + N_\gamma\right)$
$\phi(x, y, \gamma_1-1, \gamma_2)$	$\phi\left(xy\gamma_{equivalent} - N_\gamma\right)$
$\phi(x, y, \gamma_1, \gamma_2+1)$	$\phi\left(xy\gamma_{equivalent} + 1\right)$
$\phi(x, y, \gamma_1, \gamma_2-1)$	$\phi\left(xy\gamma_{equivalent} - 1\right)$
$\phi(x, y, \gamma_1+1, \gamma_2+1)$	$\phi\left(xy\gamma_{equivalent} + N_\gamma + 1\right)$
$\phi(x, y, \gamma_1-1, \gamma_2-1)$	$\phi\left(xy\gamma_{equivalent} - N_\gamma - 1\right)$

Table 4.2: Equivalents of some required expressions, using the 1D array convention.

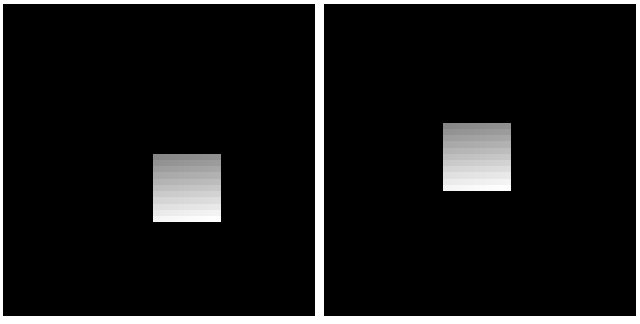
$\gamma_1 = 0$ and increasing γ_2, then, $\gamma_1 = 1$ and the same. The same pattern is then repeated for the neighboring pixel, $(x, y+1)$. Other rows (by increasing x) will be covered using the same convention.

Using the above mentioned convention, the required combinations for gradients will also have equivalent representations, as in Table 4.2.

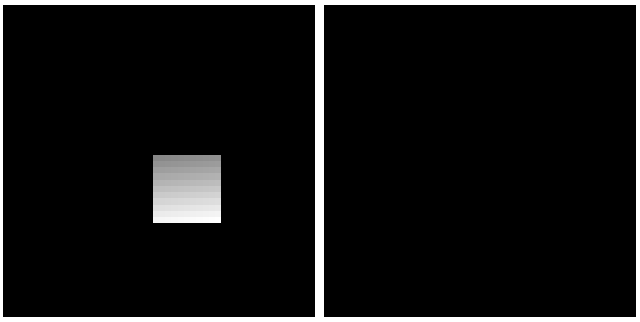
We also implemented the "Convex Image Registration" without the above mentioned technique, using 4D arrays, to first make sure about the results.

The results of our registration, are shown in 4.4c. In this example, the fixed image is a translated version of the moving image, where the translation is 5 pixels, in both x and y directions. The difference between the fixed image (4.4a) and the result of registration (4.4c) is zero, for all pixels, as shown in 4.4d. The initial and final deformations are shown in Figure 4.5.

The initialization shown in 4.5a is a random patch initialization, with all



(a) fixed image (b) moving image



(c) result (d) difference between the fixed and resultant images

Figure 4.4: Results of registration.

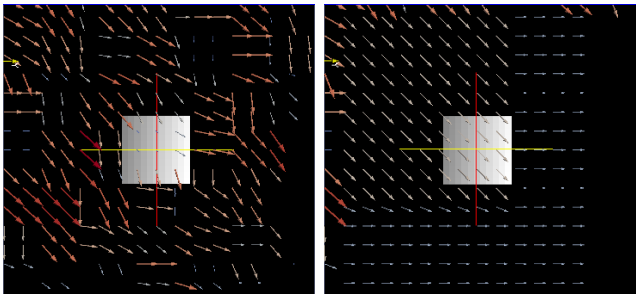
possible values of the deformation fields. The image was divided to small patches, and each patch was assigned two random values, as the x and y components of its deformation field. For pixels near the boundaries, it is possible that the deformed pixel (with the initial random values) falls out of the image frame. To avoid such a problem, the initial deformations were assumed to be zero, in these cases.

The deformation fields of Figure 4.5 are shown using ParaView [37], and some codes for the adaptations. It should also be mentioned that since the deformation field is a vector field image - which has a vector value at each pixel - it cannot be stored as a 'PNG' file, but as an 'MHD' file. Since ParaView treats these two file types differently, regarding the x and y axes, they should be appropriately converted.

The convergence criterion used is:

$$\|\phi^{n+1} - \phi^n\|_{\ell_2} < \epsilon \tag{4.2}$$

Where n is the iteration number and $\phi = 1_{\{\vec{u} \succeq \vec{\gamma}\}}$. For $\epsilon = 0.01$ and with $\lambda = 100$, a maximum deformation field of 10 pixels and an initialization



(a) (b)

Figure 4.5: (a)The initial deformation field and (b)The resultant deformation field for the example of Figure 4.4. The vectors show both the magnitude and the direction of the deformation fields.

of u values, as in Figure 4.5, the convergence was achieved in around 2000 iterations. On our machine the time is less than a minute for small images. This CPU time was obtained by running the program on a Desktop computer with Intel (R) Core CPU 2.26 GHz.

Chapter 5

Conclusion

In this work, "Region-based Atlas-based Image Registration" problem was studied. The existing formulations of this problem is nonconvex and thus their convergence is dependant on the initialization, as well as the evolution parameters, such as the step size and the algorithm.

In this work, we formulated the "Region-based Registration" as a 'convex optimization problem', so that the solution will be independent of the initialization. For this purpose, recent convex relaxation methods for vector-valued optimization problems were adapted to the region-based registration problems and its algorithm was developed. The evaluation of the proposed algorithm was performed on synthetic 2D images. The results show the convergence of the algorithm, the expected deformation field vector and a zero difference between the resultant and the fixed images, as expected.

The advantage of this algorithm is finding a *global* solution, for region-based image registration.

However, the vector valued convex relaxation could only be done at a cost of increased dimensionality and complexity. Hence, fast implementation techniques are required, in order to make this method practical. The formulation has been implemented using the primal-dual algorithm, which speeds up the convergence. As a future work, we are interested to implement this algorithm on 'GPU', in order to gain more speed up. Furthermore, the 'Piece-Wise Constant' model was used here for our region-based registration formulations. As an extension, we would like to apply the more sophisticated assumption of gaussian distribution for the intensity of each region. Finally, we would like to evaluate the algorithm on 3D image registration.

Lausanne, the 21^{st} of January 2011

Shima Sepehri

Appendices

Appendix A

Work Progress

Since this work was consisted of different tasks, a brief narration of its progress is added here.

Firstly, the advanced mathematics of [5] was covered. Then, the required algorithms for the implementation of an existing "convex segmentation" code with the watershed method - as suggested by one of the authors of [5] - were derived. Such a task might yield better results than available works. However, since the goal of this project, "Registration", was, in itself, a demanding task, it was decided to focus on it.

Afterwards, as a practice for a beginner student for a better understanding, I was guided to try to derive, myself, the "Convex Image Registration" formulations, which was already done by the assistant and scientist of the lab. However, I found some differences between my derivations and the informal available convex formulas. These differences were only cleared, by contacting their author, later on, where I found out about the problems in the informal formulation and its update, as in [5].

Then, the correlation of [5] and its implementation was studied and documented, as a commented code. Meanwhile, the fast implementation techniques, for a global solution, were also thoroughly investigated.

Subsequently, the "Two-Phase Piecewise Constant Atlas-based Registration" was implemented, based on the Primary Formulation, mentioned in subsection 3.1.2, on page 8. In this code, the fast implementation techniques - described in chapter 4 - were used. This primary implementation had problems.

In fact, I discovered the main problem of the implementation to be in the iterative formulas mentioned in subsection 3.2.1. These formulas were already implemented on 'image segmentation'. However, in the 'registration' problem, due to the change of the minimization parameter, its initialization is independent of the target image. This caused the above mentioned for-

mulas to fail and stay constant after one iteration. Questioning from one of the authors of [5] revealed that other constraints are required to solve this registration problem because of the aperture problem. After discussions with the thesis supervisor and the assistant, it was decided to first re-implement the 'pixel-based' image registration, as done in [6]. However, the details of this implementation was not in hand. Therefore, a curious reinvestigation of [6] was fulfilled. During this reread, a similar failure was noticed, when not considering a difference between the model of [6] and its previous similar work [14]. To clarify some vague points, it was tried to have some contacts, though without response. Hence, this task could not be fulfilled due to time constraints, in order to focus on our 'region-based' registration.

At this point, a line-by-line review of the code and a correlation search seemed to be necessary to make sure about the implementation. Firstly, boundary conditions were reinvestigated. Moreover, the increased dimensions were cross-checked with the algorithm, presented in [5]. Finally, the implementation of defined operators were studied thoroughly. To achieve this, basic definitions were applied to the full multi-dimensional case of the code. However, the available operator, for the iterative formulas of subsection 3.2.1, did not seem to satisfy the basic definitions. This could be due to special implementation required for this available operator (for the'segmentation' problem) in order to fit it to the 'registration' task. More precisely, to make the exact implementation of the operator A, mentioned in Equation 3.19, crystal clear, the computation added in Appendix B was performed.

Moreover, we also thought about that the implementation of the algorithm presented in subsection 3.2.1 might stay unchanged, by changing the initializations for different problems. Therefore, the implementation of [5] for 'segmentation' was also tested, for our problem, by changing its initializations.

More importantly, the problems in the initial formulas used for the implementation (Primary Formulation on page 8) were found and corrected, as in Final Formulation, on page 9.

Appendix B

Calculation of the Operator A and its Adjoint

The calculations presented here was performed after unsuccessful implementations and discussions with many experts about possible problems. It is added here in order to help to conceptually clarify all the components of the 'Convex Relaxation Method' used, for our 'Region-based Image Registration'. Furthermore, issues can sometimes be results for other works.

Firstly, the rechecking of the adjoint, implemented for the application of [5] for the 'segmentation' problem, is presented. The goal is to make sure that this calculation is suitable for our registration problem or result in the required adaptation or changes.

In this implementation the operator A is defined as follows:

$$\begin{aligned} A(\vec{\phi}) = (&\phi\left[x+1, y, \gamma_1, \gamma_2\right] - \phi\left[x, y, \gamma_1, \gamma_2\right], \\ &\phi\left[x, y+1, \gamma_1, \gamma_2\right] - \phi\left[x, y, \gamma_1, \gamma_2\right], \\ &\phi\left[x, y, \gamma_1+1, \gamma_2+1\right] - \phi\left[x, y, \gamma_1+1, \gamma_2\right] - \phi\left[x, y, \gamma_1, \gamma_2+1\right] + \phi\left[x, y, \gamma_1, \gamma_2\right]) \end{aligned} \tag{B.1}$$

and θ is defined, as

$$\begin{aligned} \vec{\theta} = (&\theta_x\left[x, y, \gamma_1, \gamma_2\right], \\ &\theta_y\left[x, y, \gamma_1, \gamma_2\right], \\ &\theta_g\left[x, y, \gamma_1, \gamma_2\right]) \end{aligned} \tag{B.2}$$

Then the left hand side of Equation B.7 should be calculated as follows for a 3×3 neighborhood:

$$\langle A(\phi), \theta \rangle =$$

$$
\begin{aligned}
&\phi[x, y-1, \gamma_1, \gamma_2]\, \theta_x[x-1, y-1, \gamma_1, \gamma_2] - \phi[x-1, y-1, \gamma_1, \gamma_2]\, \theta_x[x-1, y-1, \gamma_1, \gamma_2] \quad + \\
&\phi[x-1, y, \gamma_1, \gamma_2]\, \theta_y[x-1, y-1, \gamma_1, \gamma_2] - \phi[x-1, y-1, \gamma_1, \gamma_2]\, \theta_y[x-1, y-1, \gamma_1, \gamma_2] \quad + \\
&\phi[x-1, y-1, \gamma_1+1, \gamma_2+1]\, \theta_g[x-1, y-1, \gamma_1, \gamma_2] - \\
&\phi[x-1, y-1, \gamma_1+1, \gamma_2]\, \theta_g[x-1, y-1, \gamma_1, \gamma_2] - \\
&\phi[x-1, y-1, \gamma_1, \gamma_2+1]\, \theta_g[x-1, y-1, \gamma_1, \gamma_2] + \phi[x-1, y-1, \gamma_1, \gamma_2]\, \theta_g[x-1, y-1, \gamma_1, \gamma_2] \\
&+ \\
&\phi[x+1, y-1, \gamma_1, \gamma_2]\, \theta_x[x, y-1, \gamma_1, \gamma_2] - \phi[x, y-1, \gamma_1, \gamma_2]\, \theta_x[x, y-1, \gamma_1, \gamma_2] \quad + \\
&\phi[x, y, \gamma_1, \gamma_2]\, \theta_y[x, y-1, \gamma_1, \gamma_2] - \phi[x, y-1, \gamma_1, \gamma_2]\, \theta_y[x, y-1, \gamma_1, \gamma_2] \quad + \\
&\phi[x, y-1, \gamma_1+1, \gamma_2+1]\, \theta_g[x, y-1, \gamma_1, \gamma_2] - \phi[x, y-1, \gamma_1+1, \gamma_2]\, \theta_g[x, y-1, \gamma_1, \gamma_2] - \\
&\phi[x, y-1, \gamma_1, \gamma_2+1]\, \theta_g[x, y-1, \gamma_1, \gamma_2] + \phi[x, y-1, \gamma_1, \gamma_2]\, \theta_g[x, y-1, \gamma_1, \gamma_2] \\
&+ \\
&\phi[x+2, y-1, \gamma_1, \gamma_2]\, \theta_x[x+1, y-1, \gamma_1, \gamma_2] - \phi[x+1, y-1, \gamma_1, \gamma_2]\, \theta_x[x+1, y-1, \gamma_1, \gamma_2] \quad + \\
&\phi[x+1, y, \gamma_1, \gamma_2]\, \theta_y[x+1, y-1, \gamma_1, \gamma_2] - \phi[x+1, y-1, \gamma_1, \gamma_2]\, \theta_y[x+1, y-1, \gamma_1, \gamma_2] \quad + \\
&\phi[x+1, y-1, \gamma_1+1, \gamma_2+1]\, \theta_g[x+1, y-1, \gamma_1, \gamma_2] - \\
&\phi[x+1, y-1, \gamma_1+1, \gamma_2]\, \theta_g[x+1, y-1, \gamma_1, \gamma_2] - \\
&\phi[x+1, y-1, \gamma_1, \gamma_2+1]\, \theta_g[x+1, y-1, \gamma_1, \gamma_2] + \phi[x+1, y-1, \gamma_1, \gamma_2]\, \theta_g[x+1, y-1, \gamma_1, \gamma_2] \\
&+ \\
&\phi[x, y, \gamma_1, \gamma_2]\, \theta_x[x-1, y, \gamma_1, \gamma_2] - \phi[x-1, y, \gamma_1, \gamma_2]\, \theta_x[x-1, y, \gamma_1, \gamma_2] \quad + \\
&\phi[x-1, y+1, \gamma_1, \gamma_2]\, \theta_y[x-1, y, \gamma_1, \gamma_2] - \phi[x-1, y, \gamma_1, \gamma_2]\, \theta_y[x-1, y, \gamma_1, \gamma_2] \quad + \\
&\phi[x-1, y, \gamma_1+1, \gamma_2+1]\, \theta_g[x-1, y, \gamma_1, \gamma_2] - \phi[x-1, y, \gamma_1+1, \gamma_2]\, \theta_g[x-1, y, \gamma_1, \gamma_2] - \\
&\phi[x-1, y, \gamma_1, \gamma_2+1]\, \theta_g[x-1, y, \gamma_1, \gamma_2] + \phi[x-1, y, \gamma_1, \gamma_2]\, \theta_g[x-1, y, \gamma_1, \gamma_2] \\
&+ \\
&\phi[x+1, y, \gamma_1, \gamma_2]\, \theta_x[x, y, \gamma_1, \gamma_2] - \phi[x, y, \gamma_1, \gamma_2]\, \theta_x[x, y, \gamma_1, \gamma_2] \quad + \\
&\phi[x, y+1, \gamma_1, \gamma_2]\, \theta_y[x, y, \gamma_1, \gamma_2] - \phi[x, y, \gamma_1, \gamma_2]\, \theta_y[x, y, \gamma_1, \gamma_2] \quad + \\
&\phi[x, y, \gamma_1+1, \gamma_2+1]\, \theta_g[x, y, \gamma_1, \gamma_2] - \phi[x, y, \gamma_1+1, \gamma_2]\, \theta_g[x, y, \gamma_1, \gamma_2] - \\
&\phi[x, y, \gamma_1, \gamma_2+1]\, \theta_g[x, y, \gamma_1, \gamma_2] + \phi[x, y, \gamma_1, \gamma_2]\, \theta_g[x, y, \gamma_1, \gamma_2] \\
&+
\end{aligned}
$$

$$
\begin{aligned}
&\phi\left[x+2,y,\gamma_1,\gamma_2\right]\theta_x\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_x\left[x+1,y,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_y\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_y\left[x+1,y,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x+1,y,\gamma_1+1,\gamma_2+1\right]\theta_g\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x+1,y,\gamma_1+1,\gamma_2\right]\theta_g\left[x+1,y,\gamma_1,\gamma_2\right]- \\
&\phi\left[x+1,y,\gamma_1,\gamma_2+1\right]\theta_g\left[x+1,y,\gamma_1,\gamma_2\right]+\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y,\gamma_1,\gamma_2\right] \\
&+ \\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x-1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x-1,y+1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x-1,y+2,\gamma_1,\gamma_2\right]\theta_y\left[x-1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_y\left[x-1,y+1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x-1,y+1,\gamma_1+1,\gamma_2+1\right]\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right]- \\
&\phi\left[x-1,y+1,\gamma_1+1,\gamma_2\right]\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right]- \\
&\phi\left[x-1,y+1,\gamma_1,\gamma_2+1\right]\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right]+\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right] \\
&+ \\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x,y+1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x,y+2,\gamma_1,\gamma_2\right]\theta_y\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_y\left[x,y+1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x,y+1,\gamma_1+1,\gamma_2+1\right]\theta_g\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y+1,\gamma_1+1,\gamma_2\right]\theta_g\left[x,y+1,\gamma_1,\gamma_2\right]- \\
&\phi\left[x,y+1,\gamma_1,\gamma_2+1\right]\theta_g\left[x,y+1,\gamma_1,\gamma_2\right]+\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x,y+1,\gamma_1,\gamma_2\right] \\
&+ \\
&\phi\left[x+2,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x+1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x+1,y+1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x+1,y+2,\gamma_1,\gamma_2\right]\theta_y\left[x+1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_y\left[x+1,y+1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x+1,y+1,\gamma_1+1,\gamma_2+1\right]\theta_g\left[x+1,y+1,\gamma_1,\gamma_2\right]- \\
&\phi\left[x+1,y+1,\gamma_1+1,\gamma_2\right]\theta_g\left[x+1,y+1,\gamma_1,\gamma_2\right]- \\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2+1\right]\theta_g\left[x+1,y+1,\gamma_1,\gamma_2\right]+\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y+1,\gamma_1,\gamma_2\right]
\end{aligned}
\tag{B.3}
$$

According to the implementation of [5] for 'segmentation':

$$
\begin{aligned}
A^*(\theta) = &\theta_x\left[x,y,\gamma_1,\gamma_2\right]-\theta_x\left[x,y-1,\gamma_1,\gamma_2\right]+ \\
&\theta_y\left[x,y,\gamma_1,\gamma_2\right]-\theta_y\left[x,y-1,\gamma_1,\gamma_2\right]- \\
&\theta_g\left[x,y,\gamma_1-1,\gamma_2-1\right]+\theta_g\left[x,y,\gamma_1-1,\gamma_2\right]+\theta_g\left[x,y,\gamma_1,\gamma_2-1\right]-\theta_g\left[x,y,\gamma_1,\gamma_2\right]
\end{aligned}
\tag{B.4}
$$

Therefore, we have the right hand side of Equation B.7 as follows:

$$\left\langle \phi, A^*\left(\theta\right)\right\rangle =$$

$$
\begin{aligned}
&\phi\left[x-1,y-1,\gamma_1,\gamma_2\right]\theta_x\left[x-1,y-1,\gamma_1,\gamma_2\right]-\phi\left[x-1,y-1,\gamma_1,\gamma_2\right]\theta_x\left[x-2,y-1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x-1,y-1,\gamma_1,\gamma_2\right]\theta_y\left[x-1,y-1,\gamma_1,\gamma_2\right]-\phi\left[x-1,y-1,\gamma_1,\gamma_2\right]\theta_y\left[x-1,y-2,\gamma_1,\gamma_2\right] \ - \\
&\phi\left[x-1,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y-1,\gamma_1-1,\gamma_{2-1}\right]+ \\
&\phi\left[x-1,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y-1,\gamma_1-1,\gamma_2\right]+ \\
&\phi\left[x-1,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y-1,\gamma_1,\gamma_2-1\right]-\phi\left[x-1,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y-1,\gamma_1,\gamma_2\right] \\
&+ \\
&\phi\left[x,y-1,\gamma_1,\gamma_2\right]\theta_x\left[x,y-1,\gamma_1,\gamma_2\right]-\phi\left[x,y-1,\gamma_1,\gamma_2\right]\theta_x\left[x-1,y-1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x,y-1,\gamma_1,\gamma_2\right]\theta_y\left[x,y-1,\gamma_1,\gamma_2\right]-\phi\left[x,y-1,\gamma_1,\gamma_2\right]\theta_y\left[x,y-2,\gamma_1,\gamma_2\right] \ - \\
&\phi\left[x,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x,y-1,\gamma_1-1,\gamma_{2-1}\right]+\phi\left[x,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x,y-1,\gamma_1-1,\gamma_2\right]+ \\
&\phi\left[x,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x,y-1,\gamma_1,\gamma_2-1\right]-\phi\left[x,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x,y-1,\gamma_1,\gamma_2\right] \\
&+ \\
&\phi\left[x+1,y-1,\gamma_1,\gamma_2\right]\theta_x\left[x+1,y-1,\gamma_1,\gamma_2\right]-\phi\left[x+1,y-1,\gamma_1,\gamma_2\right]\theta_x\left[x,y-1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x+1,y-1,\gamma_1,\gamma_2\right]\theta_y\left[x+1,y-1,\gamma_1,\gamma_2\right]-\phi\left[x+1,y-1,\gamma_1,\gamma_2\right]\theta_y\left[x+1,y-2,\gamma_1,\gamma_2\right] \ - \\
&\phi\left[x+1,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y-1,\gamma_1-1,\gamma_{2-1}\right]+ \\
&\phi\left[x+1,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y-1,\gamma_1-1,\gamma_2\right]+ \\
&\phi\left[x+1,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y-1,\gamma_1,\gamma_2-1\right]-\phi\left[x+1,y-1,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y-1,\gamma_1,\gamma_2\right] \\
&+ \\
&\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_x\left[x-1,y,\gamma_1,\gamma_2\right]-\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_x\left[x-2,y,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_y\left[x-1,y,\gamma_1,\gamma_2\right]-\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_y\left[x-1,y-1,\gamma_1,\gamma_2\right] \ - \\
&\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y,\gamma_1-1,\gamma_{2-1}\right]+\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y,\gamma_1-1,\gamma_2\right]+ \\
&\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y,\gamma_1,\gamma_2-1\right]-\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y,\gamma_1,\gamma_2\right] \\
&+ \\
&\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_x\left[x,y,\gamma_1,\gamma_2\right]-\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_x\left[x-1,y,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_y\left[x,y,\gamma_1,\gamma_2\right]-\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_y\left[x,y-1,\gamma_1,\gamma_2\right] \ - \\
&\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_g\left[x,y,\gamma_1-1,\gamma_{2-1}\right]+\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_g\left[x,y,\gamma_1-1,\gamma_2\right]+ \\
&\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_g\left[x,y,\gamma_1,\gamma_2-1\right]-\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_g\left[x,y,\gamma_1,\gamma_2\right] \\
&+ \\
&\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_x\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_x\left[x,y,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_y\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_y\left[x+1,y-1,\gamma_1,\gamma_2\right] \ - \\
&\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y,\gamma_1-1,\gamma_{2-1}\right]+\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y,\gamma_1-1,\gamma_2\right]+ \\
&\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y,\gamma_1,\gamma_2-1\right]-\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y,\gamma_1,\gamma_2\right] \\
&+
\end{aligned}
$$

$$
\begin{aligned}
&\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x-1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x-2,y+1,\gamma_1,\gamma_2\right] \;+\\
&\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_y\left[x-1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_y\left[x-1,y,\gamma_1,\gamma_2\right] \;-\\
&\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y+1,\gamma_1-1,\gamma_{2-1}\right]+\\
&\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y+1,\gamma_1-1,\gamma_2\right]+\\
&\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y+1,\gamma_1,\gamma_2-1\right]-\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right]\\
&+\\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x-1,y+1,\gamma_1,\gamma_2\right] \;+\\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_y\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_y\left[x,y,\gamma_1,\gamma_2\right] \;-\\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x,y+1,\gamma_1-1,\gamma_{2-1}\right]+\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x,y+1,\gamma_1-1,\gamma_2\right]+\\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x,y+1,\gamma_1,\gamma_2-1\right]-\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x,y+1,\gamma_1,\gamma_2\right]\\
&+\\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x+1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_x\left[x,y+1,\gamma_1,\gamma_2\right] \;+\\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_y\left[x+1+1,y,\gamma_1,\gamma_2\right]-\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_y\left[x+1,y,\gamma_1,\gamma_2\right] \;-\\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y+1,\gamma_1-1,\gamma_{2-1}\right]+\\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y+1,\gamma_1-1,\gamma_2\right]+\\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y+1,\gamma_1,\gamma_2-1\right]-\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y+1,\gamma_1,\gamma_2\right]
\end{aligned}
\tag{B.5}
$$

Equation B.3 and Equation B.5 are expected to be the same. However, considering the implemented operator A and its adjoint (Equation B.1 and Equation B.4), there still remains a problem: On the left hand side of Equation B.7, there will be neighbors with the arguments $+2$, and on the right hand side -2, which cannot be matched.

Another possibility was also considered, according to [5]. Since we have two minimization variables, namely the deformation fields in x and y directions, $m = 2$ in Equation 3.19, I concluded the five dimensional vector $A(\phi)$ to be

$$
\begin{aligned}
A(\vec{\phi}) = (&\phi\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x,y,\gamma_1,\gamma_2\right],\\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y,\gamma_1,\gamma_2\right],\\
&\phi\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x,y,\gamma_1,\gamma_2\right],\\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y,\gamma_1,\gamma_2\right],\\
&\phi\left[x,y,\gamma_1+1,\gamma_2+1\right]-\phi\left[x,y,\gamma_1+1,\gamma_2\right]-\phi\left[x,y,\gamma_1,\gamma_2+1\right]+\phi\left[x,y,\gamma_1,\gamma_2\right])
\end{aligned}
\tag{B.6}
$$

This conclusion is based on the dimension and the codomain of $\vec{p}$, in Equation 3.21. Besides this conclusion, it was suggested to think about the

possibility that the gradients, mentioned in Equation 3.19, might be gradients with respect to x, y, γ_1 and γ_2. However, it seems not to be the case, because if we have $m = 3$, but these minimization variables are still a function of x and y, the definition of the operator A suggests to have six gradients, but the gradients with respect to x, y, γ_1, γ_2 and γ_3 are only five gradients. Before all, we considered that the four p's in Equation 3.21 is due to the isotropic gradient. Therefore, without isotropic gradient, we should have two gradients - instead of four - in Equation 3.19, with respect to x and y. However, since the number of these gradients are mentioned to be $2m$ ([5]), without isotropic gradient, this number will be m. And if we consider the gradients to be with respect to the variable, we need the assumption that the number of variables is always the same as the number of minimization parameters. Checking this assumption is also mentioned at the end of this appendix.

To find the adjoint of operator A, we should find an operator A^* which satisfies the definition of the adjoint operator

$$< A(\phi), \theta >=< \phi, A^*(\theta) > \tag{B.7}$$

Consider the five dimensional vector θ to be

$$\begin{aligned}\vec{\theta} = (&\theta_{x_1}[x, y, \gamma_1, \gamma_2],\\ &\theta_{y_1}[x, y, \gamma_1, \gamma_2],\\ &\theta_{x_2}[x, y, \gamma_1, \gamma_2],\\ &\theta_{y_2}[x, y, \gamma_1, \gamma_2],\\ &\theta_g[x, y, \gamma_1, \gamma_2])\end{aligned} \tag{B.8}$$

Normally, we have an idea of the adjoint and check that with the Equation B.7. We also had the idea that the adjoint of the 'discrete gradient, written with forward difference' might be the 'divergence, using backward difference'. However, this did not exactly satisfy Equation B.7. After discussions with the experts, I found out about the following basic extraction of the adjoint, from the operator, itself.

Note that we aim to satisfy the Equation B.7. On the left hand side of this equation, the operator A is applied to ϕ, and therefore the variations are on the variables of ϕ. In contrary, on the right hand side, the adjoint operator is applied to θ, and hence the variations are on the variables of θ. It is obvious that in order to have an equality of the form Equation B.7, the variations should be on the variables of the same functions in both sides. Considering the fact that ϕ or θ are vector fields (vectors at each point of the space), the left hand side of Equation B.7 was calculated on a small 3×3 neighborhood,

in order to become able to end in the right hand side. A 3×3 neighborhood was the smallest one we can choose while having the main pixel, (x, y), out of the boundaries. On a 3×3 neighborhood, with (x, y) be the pixel in the center, we have

$$\langle A(\phi), \theta \rangle =$$

$$
\begin{aligned}
&\phi[x, y-1, \gamma_1, \gamma_2]\, \theta_{x_1}[x-1, y-1, \gamma_1, \gamma_2] - \phi[x-1, y-1, \gamma_1, \gamma_2]\, \theta_{x_1}[x-1, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x-1, y, \gamma_1, \gamma_2]\, \theta_{y_1}[x-1, y-1, \gamma_1, \gamma_2] - \phi[x-1, y-1, \gamma_1, \gamma_2]\, \theta_{y_1}[x-1, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x, y-1, \gamma_1, \gamma_2]\, \theta_{x_2}[x-1, y-1, \gamma_1, \gamma_2] - \phi[x-1, y-1, \gamma_1, \gamma_2]\, \theta_{x_2}[x-1, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x-1, y, \gamma_1, \gamma_2]\, \theta_{y_2}[x-1, y-1, \gamma_1, \gamma_2] - \phi[x-1, y-1, \gamma_1, \gamma_2]\, \theta_{y_2}[x-1, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x-1, y-1, \gamma_1+1, \gamma_2+1]\, \theta_g[x-1, y-1, \gamma_1, \gamma_2] - \\
&\phi[x-1, y-1, \gamma_1+1, \gamma_2]\, \theta_g[x-1, y-1, \gamma_1, \gamma_2] - \\
&\phi[x-1, y-1, \gamma_1, \gamma_2+1]\, \theta_g[x-1, y-1, \gamma_1, \gamma_2] + \\
&\phi[x-1, y-1, \gamma_1, \gamma_2]\, \theta_g[x-1, y-1, \gamma_1, \gamma_2] \\
&+ \\
&\phi[x+1, y-1, \gamma_1, \gamma_2]\, \theta_{x_1}[x, y-1, \gamma_1, \gamma_2] - \phi[x, y-1, \gamma_1, \gamma_2]\, \theta_{x_1}[x, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x, y, \gamma_1, \gamma_2]\, \theta_{y_1}[x, y-1, \gamma_1, \gamma_2] - \phi[x, y-1, \gamma_1, \gamma_2]\, \theta_{y_1}[x, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x+1, y-1, \gamma_1, \gamma_2]\, \theta_{x_2}[x, y-1, \gamma_1, \gamma_2] - \phi[x, y-1, \gamma_1, \gamma_2]\, \theta_{x_2}[x, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x, y, \gamma_1, \gamma_2]\, \theta_{y_2}[x, y-1, \gamma_1, \gamma_2] - \phi[x, y-1, \gamma_1, \gamma_2]\, \theta_{y_2}[x, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x, y-1, \gamma_1+1, \gamma_2+1]\, \theta_g[x, y-1, \gamma_1, \gamma_2] - \\
&\phi[x, y-1, \gamma_1+1, \gamma_2]\, \theta_g[x, y-1, \gamma_1, \gamma_2] - \\
&\phi[x, y-1, \gamma_1, \gamma_2+1]\, \theta_g[x, y-1, \gamma_1, \gamma_2] + \\
&\phi[x, y-1, \gamma_1, \gamma_2]\, \theta_g[x, y-1, \gamma_1, \gamma_2] \\
&+ \\
&\phi[x+2, y-1, \gamma_1, \gamma_2]\, \theta_{x_1}[x+1, y-1, \gamma_1, \gamma_2] - \phi[x+1, y-1, \gamma_1, \gamma_2]\, \theta_{x_1}[x+1, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x+1, y, \gamma_1, \gamma_2]\, \theta_{y_1}[x+1, y-1, \gamma_1, \gamma_2] - \phi[x+1, y-1, \gamma_1, \gamma_2]\, \theta_{y_1}[x+1, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x+2, y-1, \gamma_1, \gamma_2]\, \theta_{x_2}[x+1, y-1, \gamma_1, \gamma_2] - \phi[x+1, y-1, \gamma_1, \gamma_2]\, \theta_{x_2}[x+1, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x+1, y, \gamma_1, \gamma_2]\, \theta_{y_2}[x+1, y-1, \gamma_1, \gamma_2] - \phi[x+1, y-1, \gamma_1, \gamma_2]\, \theta_{y_2}[x+1, y-1, \gamma_1, \gamma_2] \; + \\
&\phi[x+1, y-1, \gamma_1+1, \gamma_2+1]\, \theta_g[x+1, y-1, \gamma_1, \gamma_2] - \\
&\phi[x+1, y-1, \gamma_1+1, \gamma_2]\, \theta_g[x+1, y-1, \gamma_1, \gamma_2] - \\
&\phi[x+1, y-1, \gamma_1, \gamma_2+1]\, \theta_g[x+1, y-1, \gamma_1, \gamma_2] + \\
&\phi[x+1, y-1, \gamma_1, \gamma_2]\, \theta_g[x+1, y-1, \gamma_1, \gamma_2] \\
&+
\end{aligned}
$$

$$
\begin{aligned}
&\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_{x_1}\left[x-1,y,\gamma_1,\gamma_2\right]-\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_{x_1}\left[x-1,y,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x-1,y,\gamma_1,\gamma_2\right]-\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x-1,y,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x-1,y,\gamma_1,\gamma_2\right]-\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x-1,y,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x-1,y,\gamma_1,\gamma_2\right]-\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x-1,y,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x-1,y,\gamma_1+1,\gamma_2+1\right]\theta_g\left[x-1,y,\gamma_1,\gamma_2\right]-\phi\left[x-1,y,\gamma_1+1,\gamma_2\right]\theta_g\left[x-1,y,\gamma_1,\gamma_2\right]-\\
&\phi\left[x-1,y,\gamma_1,\gamma_2+1\right]\theta_g\left[x-1,y,\gamma_1,\gamma_2\right]+\phi\left[x-1,y,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y,\gamma_1,\gamma_2\right]\\
&+\\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x,y,\gamma_1,\gamma_2\right]-\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x,y,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x,y,\gamma_1,\gamma_2\right]-\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x,y,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x,y,\gamma_1,\gamma_2\right]-\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x,y,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x,y,\gamma_1+1,\gamma_2+1\right]\theta_g\left[x,y,\gamma_1,\gamma_2\right]-\phi\left[x,y,\gamma_1+1,\gamma_2\right]\theta_g\left[x,y,\gamma_1,\gamma_2\right]-\\
&\phi\left[x,y,\gamma_1,\gamma_2+1\right]\theta_g\left[x,y,\gamma_1,\gamma_2\right]+\phi\left[x,y,\gamma_1,\gamma_2\right]\theta_g\left[x,y,\gamma_1,\gamma_2\right]\\
&+\\
&\phi\left[x+2,y,\gamma_1,\gamma_2\right]\theta_{x_1}\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_{x_1}\left[x+1,y,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x+1,y,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x+2,y,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x+1,y,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x+1,y,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x+1,y,\gamma_1+1,\gamma_2+1\right]\theta_g\left[x+1,y,\gamma_1,\gamma_2\right]-\phi\left[x+1,y,\gamma_1+1,\gamma_2\right]\theta_g\left[x+1,y,\gamma_1,\gamma_2\right]-\\
&\phi\left[x+1,y,\gamma_1,\gamma_2+1\right]\theta_g\left[x+1,y,\gamma_1,\gamma_2\right]+\phi\left[x+1,y,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y,\gamma_1,\gamma_2\right]\\
&+\\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_{x_1}\left[x-1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_{x_1}\left[x-1,y+1,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x-1,y+2,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x-1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x-1,y+1,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x-1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x-1,y+1,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x-1,y+2,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x-1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x-1,y+1,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x-1,y+1,\gamma_1+1,\gamma_2+1\right]\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right]-\\
&\phi\left[x-1,y+1,\gamma_1+1,\gamma_2\right]\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right]-\\
&\phi\left[x-1,y+1,\gamma_1,\gamma_2+1\right]\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right]+\\
&\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right]\\
&+\\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_{x_1}\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_{x_1}\left[x,y+1,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x,y+2,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x,y+1,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x,y+1,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x,y+2,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x,y+1,\gamma_1,\gamma_2\right]\ +\\
&\phi\left[x,y+1,\gamma_1+1,\gamma_2+1\right]\theta_g\left[x,y+1,\gamma_1,\gamma_2\right]-\phi\left[x,y+1,\gamma_1+1,\gamma_2\right]\theta_g\left[x,y+1,\gamma_1,\gamma_2\right]-\\
&\phi\left[x,y+1,\gamma_1,\gamma_2+1\right]\theta_g\left[x,y+1,\gamma_1,\gamma_2\right]+\phi\left[x,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x,y+1,\gamma_1,\gamma_2\right]\\
&+
\end{aligned}
$$

$$
\begin{aligned}
&\phi\left[x+2,y+1,\gamma_1,\gamma_2\right]\theta_{x_1}\left[x+1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_{x_1}\left[x+1,y+1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x+1,y+2,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x+1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_{y_1}\left[x+1,y+1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x+2,y+1,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x+1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x+1,y+1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x+1,y+2,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x+1,y+1,\gamma_1,\gamma_2\right]-\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_{y_2}\left[x+1,y+1,\gamma_1,\gamma_2\right] \ + \\
&\phi\left[x+1,y+1,\gamma_1+1,\gamma_2+1\right]\theta_g\left[x+1,y+1,\gamma_1,\gamma_2\right]- \\
&\phi\left[x+1,y+1,\gamma_1+1,\gamma_2\right]\theta_g\left[x+1,y+1,\gamma_1,\gamma_2\right]- \\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2+1\right]\theta_g\left[x+1,y+1,\gamma_1,\gamma_2\right]+ \\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\theta_g\left[x+1,y+1,\gamma_1,\gamma_2\right]
\end{aligned}
\tag{B.9}
$$

Note that in this formula, each portion - separated by + signs - is related to a pixel. So, the portions belong to the pixels $(x-1,y-1)$, $(x,y-1)$, $(x+1,y-1)$, $(x-1,y)$, (x,y), $(x+1,y)$, $(x-1,y+1)$, $(x,y+1)$ and $(x+1,y+1)$, respectively.

In order to yield a combination of the form of the right hand side of Equation B.7, Equation B.9 should be regrouped with respect to ϕ, as follows:

$$
\begin{aligned}
&\phi\left[x,y-1,\gamma_1,\gamma_2\right]\left(\right. \\
&\theta_{x_1}\left[x-1,y-1,\gamma_1,\gamma_2\right]+\theta_{x_2}\left[x-1,y-1,\gamma_1,\gamma_2\right]-\theta_{x_1}\left[x,y-1,\gamma_1,\gamma_2\right]- \\
&\theta_{y_1}\left[x,y-1,\gamma_1,\gamma_2\right]-\theta_{x_2}\left[x,y-1,\gamma_1,\gamma_2\right]-\theta_{y_2}\left[x,y-1,\gamma_1,\gamma_2\right]+\theta_g\left[x,y-1,\gamma_1,\gamma_2\right]\left.\right) \\
&- \\
&\phi\left[x-1,y-1,\gamma_1,\gamma_2\right]\left(\right. \\
&\theta_{x_1}\left[x-1,y-1,\gamma_1,\gamma_2\right]+\theta_{y_1}\left[x-1,y-1,\gamma_1,\gamma_2\right]+\theta_{x_2}\left[x-1,y-1,\gamma_1,\gamma_2\right]+ \\
&\theta_{y_2}\left[x-1,y-1,\gamma_1,\gamma_2\right]-\theta_g\left[x-1,y-1,\gamma_1,\gamma_2\right]\left.\right) \\
&+ \\
&\phi\left[x-1,y,\gamma_1,\gamma_2\right]\left(\right. \\
&\theta_{y_1}\left[x-1,y-1,\gamma_1,\gamma_2\right]+\theta_{y_2}\left[x-1,y-1,\gamma_1,\gamma_2\right]-\theta_{x_1}\left[x-1,y,\gamma_1,\gamma_2\right]- \\
&\theta_{y_1}\left[x-1,y,\gamma_1,\gamma_2\right]-\theta_{x_2}\left[x-1,y,\gamma_1,\gamma_2\right]-\theta_{y_2}\left[x-1,y,\gamma_1,\gamma_2\right]-\theta_g\left[x-1,y,\gamma_1,\gamma_2\right]\left.\right) \\
&+ \\
&\phi\left[x-1,y-1,\gamma_1+1,\gamma_2+1\right]\left(\theta_g\left[x-1,y-1,\gamma_1,\gamma_2\right]\right) \\
&- \\
&\phi\left[x-1,y-1,\gamma_1+1,\gamma_2\right]\left(\theta_g\left[x-1,y-1,\gamma_1,\gamma_2\right]\right) \\
&- \\
&\phi\left[x-1,y-1,\gamma_1,\gamma_2+1\right]\left(\theta_g\left[x-1,y-1,\gamma_1,\gamma_2\right]\right)
\end{aligned}
$$

$$
+
$$
$$
\phi\left[x+1,y-1,\gamma_1,\gamma_2\right]\left(\right.
$$
$$
\theta_{x_1}\left[x,y-1,\gamma_1,\gamma_2\right]+\theta_{x_2}\left[x,y-1,\gamma_1,\gamma_2\right]-\theta_{x_1}\left[x+1,y-1,\gamma_1,\gamma_2\right]-
$$
$$
\theta_{y_1}\left[x+1,y-1,\gamma_1,\gamma_2\right]-\theta_{x_2}\left[x+1,y-1,\gamma_1,\gamma_2\right]-\theta_{y_2}\left[x+1,y-1,\gamma_1,\gamma_2\right]+
$$
$$
\left.\theta_g\left[x+1,y-1,\gamma_1,\gamma_2\right]\right)
$$
$$
+
$$
$$
\phi\left[x,y,\gamma_1,\gamma_2\right]\left(\right.
$$
$$
\theta_{y_1}\left[x,y-1,\gamma_1,\gamma_2\right]+\theta_{y_2}\left[x,y-1,\gamma_1,\gamma_2\right]-\theta_{x_1}\left[x,y,\gamma_1,\gamma_2\right]-\theta_{y_1}\left[x,y,\gamma_1,\gamma_2\right]-
$$
$$
\theta_{x_2}\left[x,y,\gamma_1,\gamma_2\right]-\theta_{y_2}\left[x,y,\gamma_1,\gamma_2\right]+\theta_g\left[x,y,\gamma_1,\gamma_2\right]+\theta_{x_1}\left[x-1,y,\gamma_1,\gamma_2\right]+
$$
$$
\left.\theta_{x_2}\left[x-1,y,\gamma_1,\gamma_2\right]\right)
$$
$$
+
$$
$$
\phi\left[x,y-1,\gamma_1+1,\gamma_2+1\right]\left(\theta_g\left[x,y-1,\gamma_1,\gamma_2\right]\right)
$$
$$
-
$$
$$
\phi\left[x,y-1,\gamma_1+1,\gamma_2\right]\left(\theta_g\left[x,y-1,\gamma_1,\gamma_2\right]\right)
$$
$$
-
$$
$$
\phi\left[x,y-1,\gamma_1,\gamma_2+1\right]\left(\theta_g\left[x,y-1,\gamma_1,\gamma_2\right]\right)
$$
$$
+
$$
$$
\phi\left[x+2,y-1,\gamma_1,\gamma_2\right]\left(\theta_{x_1}\left[x+1,y-1,\gamma_1,\gamma_2\right]+\theta_{x_2}\left[x+1,y-1,\gamma_1,\gamma_2\right]\right)
$$
$$
+
$$
$$
\phi\left[x+1,y,\gamma_1,\gamma_2\right]\left(\right.
$$
$$
\theta_{y_1}\left[x+1,y-1,\gamma_1,\gamma_2\right]+\theta_{y_2}\left[x+1,y-1,\gamma_1,\gamma_2\right]-\theta_{x_1}\left[x+1,y,\gamma_1,\gamma_2\right]-
$$
$$
\theta_{y_1}\left[x+1,y,\gamma_1,\gamma_2\right]-\theta_{x_2}\left[x+1,y,\gamma_1,\gamma_2\right]-\theta_{y_2}\left[x+1,y,\gamma_1,\gamma_2\right]+
$$
$$
\theta_g\left[x+1,y,\gamma_1,\gamma_2\right]+\theta_{x_1}\left[x,y,\gamma_1,\gamma_2\right]+\theta_{x_2}\left[x,y,\gamma_1,\gamma_2\right]\right)
$$
$$
+
$$
$$
\phi\left[x+1,y-1,\gamma_1+1,\gamma_2+1\right]\left(\theta_g\left[x+1,y-1,\gamma_1,\gamma_2\right]\right)
$$
$$
-
$$
$$
\phi\left[x+1,y-1,\gamma_1+1,\gamma_2\right]\left(\theta_g\left[x+1,y-1,\gamma_1,\gamma_2\right]\right)
$$
$$
-
$$
$$
\phi\left[x+1,y-1,\gamma_1,\gamma_2+1\right]\left(\theta_g\left[x+1,y-1,\gamma_1,\gamma_2\right]\right)
$$
$$
+
$$
$$
\phi\left[x-1,y+1,\gamma_1,\gamma_2\right]\left(\right.
$$
$$
\theta_{y_1}\left[x-1,y,\gamma_1,\gamma_2\right]+\theta_{y_2}\left[x-1,y,\gamma_1,\gamma_2\right]-\theta_{x_1}\left[x-1,y+1,\gamma_1,\gamma_2\right]-
$$
$$
\theta_{y_1}\left[x-1,y+1,\gamma_1,\gamma_2\right]-\theta_{x_2}\left[x-1,y+1,\gamma_1,\gamma_2\right]-\theta_{y_2}\left[x-1,y+1,\gamma_1,\gamma_2\right]+
$$
$$
\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right]\right)
$$
$$
+
$$

$$
\begin{aligned}
&\phi\left[x-1,y,\gamma_1+1,\gamma_2+1\right]\left(\theta_g\left[x-1,y,\gamma_1,\gamma_2\right]\right)\\
&-\\
&\phi\left[x-1,y,\gamma_1+1,\gamma_2\right]\left(\theta_g\left[x-1,y,\gamma_1,\gamma_2\right]\right)\\
&-\\
&\phi\left[x-1,y,\gamma_1,\gamma_2+1\right]\left(\theta_g\left[x-1,y,\gamma_1,\gamma_2\right]\right)\\
&+\\
&\phi\left[x,y+1,\gamma_1,\gamma_2\right]\left(\right.\\
&\theta_{y_1}\left[x,y,\gamma_1,\gamma_2\right]+\theta_{y_2}\left[x,y,\gamma_1,\gamma_2\right]-\theta_{x_1}\left[x,y+1,\gamma_1,\gamma_2\right]-\\
&\theta_{y_1}\left[x,y+1,\gamma_1,\gamma_2\right]-\theta_{x_2}\left[x,y+1,\gamma_1,\gamma_2\right]-\theta_{y_2}\left[x,y+1,\gamma_1,\gamma_2\right]+\\
&\left.\theta_g\left[x,y+1,\gamma_1,\gamma_2\right]+\theta_{x_1}\left[x-1,y+1,\gamma_1,\gamma_2\right]+\theta_{x_2}\left[x-1,y+1,\gamma_1,\gamma_2\right]\right)\\
&+\\
&\phi\left[x,y,\gamma_1+1,\gamma_2+1\right]\left(\theta_g\left[x,y,\gamma_1,\gamma_2\right]\right)\\
&-\\
&\phi\left[x,y,\gamma_1+1,\gamma_2\right]\left(\theta_g\left[x,y,\gamma_1,\gamma_2\right]\right)\\
&-\\
&\phi\left[x,y,\gamma_1,\gamma_2+1\right]\left(\theta_g\left[x,y,\gamma_1,\gamma_2\right]\right)\\
&+\\
&\phi\left[x+2,y,\gamma_1,\gamma_2\right]\left(\theta_{x_1}\left[x+1,y,\gamma_1,\gamma_2\right]+\theta_{x_2}\left[x+1,y,\gamma_1,\gamma_2\right]\right)\\
&+\\
&\phi\left[x+1,y+1,\gamma_1,\gamma_2\right]\left(\right.\\
&\theta_{y_1}\left[x+1,y,\gamma_1,\gamma_2\right]+\theta_{y_2}\left[x+1,y,\gamma_1,\gamma_2\right]-\theta_{x_1}\left[x+1,y+1,\gamma_1,\gamma_2\right]-\\
&\theta_{y_1}\left[x+1,y+1,\gamma_1,\gamma_2\right]-\theta_{x_2}\left[x+1,y+1,\gamma_1,\gamma_2\right]-\theta_{y_2}\left[x+1,y+1,\gamma_1,\gamma_2\right]+\\
&\left.\theta_g\left[x+1,y+1,\gamma_1,\gamma_2\right]+\theta_{x_1}\left[x,y+1,\gamma_1,\gamma_2\right]\theta_{x_2}\left[x,y+1,\gamma_1,\gamma_2\right]\right)\\
&+\\
&\phi\left[x+1,y,\gamma_1+1,\gamma_2+1\right]\left(\theta_g\left[x+1,y,\gamma_1,\gamma_2\right]\right)\\
&-\\
&\phi\left[x+1,y,\gamma_1+1,\gamma_2\right]\left(\theta_g\left[x+1,y,\gamma_1,\gamma_2\right]\right)\\
&-\\
&\phi\left[x+1,y,\gamma_1,\gamma_2+1\right]\left(\theta_g\left[x+1,y,\gamma_1,\gamma_2\right]\right)\\
&+\\
&\phi\left[x-1,y+2,\gamma_1,\gamma_2\right]\left(\theta_{y_1}\left[x-1,y+1,\gamma_1,\gamma_2\right]+\theta_{y_2}\left[x-1,y+1,\gamma_1,\gamma_2\right]\right)\\
&+\\
&\phi\left[x-1,y+1,\gamma_1+1,\gamma_2+1\right]\left(\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right]\right)\\
&-\\
&\phi\left[x-1,y+1,\gamma_1+1,\gamma_2\right]\left(\theta_g\left[x-1,y+1,\gamma_1,\gamma_2\right]\right)\\
&-
\end{aligned}
$$

$$
\begin{aligned}
&\phi\left[x-1, y+1, \gamma_1, \gamma_2+1\right]\left(\theta_g\left[x-1, y+1, \gamma_1, \gamma_2\right]\right) \\
&+ \\
&\phi\left[x, y+2, \gamma_1, \gamma_2\right]\left(\theta_{y_1}\left[x, y+1, \gamma_1, \gamma_2\right]+\theta_{y_2}\left[x, y+1, \gamma_1, \gamma_2\right]\right) \\
&+ \\
&\phi\left[x, y+1, \gamma_1+1, \gamma_2+1\right]\left(\theta_g\left[x, y+1, \gamma_1, \gamma_2\right]\right) \\
&- \\
&\phi\left[x, y+1, \gamma_1+1, \gamma_2\right]\left(\theta_g\left[x, y+1, \gamma_1, \gamma_2\right]\right) \\
&- \\
&\phi\left[x, y+1, \gamma_1, \gamma_2+1\right]\left(\theta_g\left[x, y+1, \gamma_1, \gamma_2\right]\right) \\
&+ \\
&\phi\left[x+2, y+1, \gamma_1, \gamma_2\right]\left(\theta_{x_1}\left[x+1, y+1, \gamma_1, \gamma_2\right]+\theta_{x_2}\left[x+1, y+1, \gamma_1, \gamma_2\right]\right) \\
&+ \\
&\phi\left[x+1, y+2, \gamma_1, \gamma_2\right]\left(\theta_{y_1}\left[x+1, y+1, \gamma_1, \gamma_2\right]+\theta_{y_2}\left[x+1, y+1, \gamma_1, \gamma_2\right]\right) \\
&+ \\
&\phi\left[x+1, y+1, \gamma_1+1, \gamma_2+1\right]\left(\theta_g\left[x+1, y+1, \gamma_1, \gamma_2\right]\right) \\
&- \\
&\phi\left[x+1, y+1, \gamma_1+1, \gamma_2\right]\left(\theta_g\left[x+1, y+1, \gamma_1, \gamma_2\right]\right) \\
&- \\
&\phi\left[x+1, y+1, \gamma_1, \gamma_2+1\right]\left(\theta_g\left[x+1, y+1, \gamma_1, \gamma_2\right]\right)
\end{aligned}
\tag{B.10}
$$

The regrouping of the Equation B.10 should reveal repeating patterns which are only different in the (x, y) arguments, for the nine neighbors. So that we can end in the desired expression for the adjoint.

List of Figures

Bibliography

[1] Barbara Zitova and Jan Flusser. Image registration methods: a survey. *Image and Vision Computing*, (21):9771000, June 2003. Corresponding author. Tel.: 420-2-6605-2390; fax: 420-2-8468-0730. E-mail address: zitova@utia.cas.cz (B. Zitova), flusser@utia.cas.cz(J. Flusser).

[2] Bart ter Haar Romeny. Medical image registration. Presentation technische universiteit eindhoven, 2009.

[3] Torsten Rohlfing, Robert Brandt, Randolf Menzel, Daniel B. Russakoff, and Jr. Calvin R. Maurer. *Quo Vadis, Atlas-Based Segmentation?*, volume III: Segmentation and Registration Models, Kluwer Academic of *The Handbook of Medical Image Analysis*, chapter 11, pages 435–486. Plenum Publishers, New York, NY, 2005.

[4] Subrahmanyam Gorthi, Valrie Duay, Xavier Bresson, Meritxell Bach Cuadra, Sanchez Castro, F. Javier, Claudio Pollo, Abdelkarim S. Allal, and Jean-Philippe Thiran. Active deformation fields: Dense deformation field estimation for atlas-based segmentation using the active contour framework. *Medical Image Analysis*, 2010.

[5] Ethan S. Brown, Tony F. Chan, and Xavier Bresson. A convex relaxation method for a class of vector-valued minimization problems with applications to mumford-shah segmentation. cam report 10-43, University of California, Los Angeles, July 2010. Authors are with the Department of Mathematics, University of California, Los Angeles. Email: ethan,chan,xbresson@math.ucla.edu. This work was supported in part by NSF IIS-0914580 and ONR N00014-09-1-0105.

[6] Tom Goldstein, Xavier Bresson, and Stanley Osher. Global minimization of markov random fields with applications to optical flow. cam report 09-77, University of California, Los Angeles, September 2009.

[7] Tony F. Chan and Selim Esedoglu snd Mila Nikolova. Algorithms for finding global minimizers of image segmentation and denoising model.

Siam Journal on Applied Mathematics, 66(5):1632–1648, June 2006. Received by the editors September 17, 2004; accepted for publication (in revised form) November 21, 2005; published electronically June 19, 2006.

[8] Tony F. Chan and Luminita A. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–276, February 2001.

[9] Gilbert Strang. Maximal flow through a domain. *Mathematical Programming*, pages 123–143, 1983.

[10] Antonin Chambolle, Stacey E. Levine, Bradley, and J. Lucier. Upwind and multiscale finite-difference methods for total variationbased image smoothing.

[11] Ethan S. Brown, Tony F. Chan, and Xavier Bresson. Globally convex chan-vese image segmentation. cam report 10-44, University of California, Los Angeles, July 2010. Authors are with the Department of Mathematics, University of California, Los Angeles. Email: ethan,chan,xbresson@math.ucla.edu. This work was supported in part by NSF IIS-0914580 and ONR N00014-09-1-0105.

[12] Arnaud Le Carvennec. Gpu-accelerated convex multi-phase image segmentation. Internship report, cole Polytechnique Fdrale de Lausanne, Septembre 2010.

[13] Christoph Schnorr. Convex and non-convex optimization. Presentation University of Mannheim, Germany, 2006.

[14] Thomas Pock, Thomas Schoenemann, Gottfried Graber, Horst Bischof, and Daniel Cremers. A convex formulation of continuous multi-label problems. *ECCV 2008, LNCS 5304*, Part III:792–805, 2008.

[15] R. Ravi. Approximations algorithms. web, November 2005. Carnegie Mellon School of Computer Science Lecture.

[16] Antonin Chambolle. Total variation minimization and applications. In *Inverse Problems Workshop Series*, II. CEREMADE - CNRS UMR 7534 Universite Paris Dauphine and CMAP, Ecole Polytechnique, IPAM, UCLA, 2003.

[17] Xin Sun, Hongxun Yao, Zhongqian Sun, and Bineng Zhong. A determined binary level set method based on mean shift for contour tracking. *PCM 2010, Part I, LNCS 6297*, pages 425–436, 2010.

[18] Ethan S. Brown, Tony F. Chan, and Xavier Bresson. A convex approach for multi-phase piecewise constant mumford-shah image segmentation. cam report 09-66, University of California, Los Angeles, July 2010. Revised July 2010.

[19] Ernie Esser, Xiaoqun Zhang, and Tony Chan. A general framework for a class of first order primal-dual algorithms for tv minimization. cam report 09-67, University of California, Los Angeles, August 2009.

[20] Erlend Hodneland, Xue-Cheng Tai, Joachim Weickert, Nickolay V. Bukoreshtliev, Arvid Lundervold, and Hans-Hermann Gerdes. Level set methods for watershed image segmentation. cam report 06-67, University of California, Los Angeles, December 2006.

[21] Determining optical flow. *Artifical intelligence*, (17):185203, 1981.

[22] Xavier Bresson, Selim Esedoglu, Pierre Vandergheynst, Jean-Philippe Thiran, and Stanley Osher. Fast global minimization of the active contour/snake model. *Journal of Mathematical Imaging and Vision*, (28):151–167, 2007.

[23] R. Malladi, R. Kimmel, D. Adalsteinsson, G. Sapiro, V. Caselles, and J. A. Sethian. A geometric approach to segmentation and analysis of 3d medical images. *MMBIA '96: Proceedings of the 1996 Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA '96), IEEE Computer Society*, page 244. Washington, DC, USA.

[24] Guy Gilboa and Stanley Osher. Nonlocal linear image regularization and supervised segmentation. *Multiscale Modeling and Simulation*, 6:595–630, July 2007.

[25] ao P. Oliveira, Jo Jos M. Bioucas-Dias, and Mario A. T. Figueiredo. Adaptive total variation image deblurring : A majorization-minimization approach. *Signal Processing*, 9(89):1683–1693, 2009.

[26] Y Wang, W Yin, and Y Zhang. A fast algorithm for image deblurring with total variation regularization. Technical report, Department of Computational And Applied Mathematics (CAAM), Rice University, 2007.

[27] L Rudin, S Osher, and E Fatemi. Nonlinear total variation based noise removal algorithms. *Physica. D.*, (60):259–268, 1992.

[28] Stanley Osher, Martin Burger, Donald Goldfarb, Jinjun Xu, and Wotao Yin4. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling and Simulation*, pages 460–489, 2005.

[29] Tony F. Chan, Gene H. Golub, and Pep Mulet. A nonlinear primal-dual method for total variation-based image restoration. *SIAM Journal on Scientific Computing*, (20):1964–1977, 1999.

[30] Antonin Chambolle. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision*, 1-2(20):89–97, 2004.

[31] Jean-Franois Aujol and Antonin Chambolle. Dual norms and image decomposition models. *International Journal of Computer Vision*, 1(63):85–104, 2005.

[32] Tom Goldstein and Stanley Osher. The split bregman method for l1 regularized problems. cam report 08-29, University of California, Los Angeles, 2008.

[33] Simon Setzer. Split bregman algorithm, douglas-rachford splitting and frame shrinkage. *Second International Conference on Scale Space Methods and Variational Methods in Computer Vision*, 2009.

[34] Hiroshi Ishikawa. Exact optimization for markov random fields with convex priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (25):1333–1336, 2003.

[35] David Mumford and Jayant Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(5):577–685, 1989.

[36] Thomas Pock, Antonin Chambolle, Horst Bischof, and Daniel Cremers. An algorithm for minimizing the piecewise smooth mumford-shah functional. *IEEE Conference on Computer Vision (ICCV)*, 2009.

[37] Paul A. Yushkevich, Joseph Piven, Heather Cody Hazlett, Rachel Gimpel Smith, Sean Ho, James C. Gee, and Guido Gerig. User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *Neuroimage*, 31(3):1116–1128, 2006.