

Investigation of Intensity Correction in the Context of Image Registration

by

Piotr Urban

A thesis submitted in partial fulfilment for the requirements of the degree of
Master of Philosophy at the University of Central Lancashire

University of Central Lancashire, Preston UK
School of Computing Engineering and Physical Sciences
Applied Digital Signal and Image Processing Research Centre

The research was part of Metrology Guided Radiation Therapy project
which was funded by Engineering and Physical Science Research Council

June 2010

Contents

1	Introduction	9
1.1	Background	9
1.2	Image registration	10
1.2.1	Single and multi-modal images	10
1.2.2	Examples of image registration	10
1.3	Similarity measure and intensity mapping	11
1.4	Aims of the research	15
2	Investigation of image registration	16
2.1	Intensity and landmark based registration	16
2.2	Definitions	16
2.2.1	Image and grid	17
2.2.2	Interpolation	17
2.2.3	Spatial transformation with reverse mapping	18
2.2.4	Image registration and similarity measure	19
2.3	Parametric registration	19
2.3.1	General algorithm	19
2.3.2	Multi-resolution	21
2.3.3	Deformation models	22
2.3.4	Affine transformation	24
2.3.5	B-Spline transformation	25
2.3.6	Similarity measure	30
2.3.7	Image registration and optimisation	45
2.4	Affine 2-d registration	50
2.4.1	The algorithm implemented	50
2.4.2	Experimental results	52
2.4.3	Concluding remarks	53
2.5	Nonrigid B-Spline 1-d registration	61
2.5.1	The algorithm implemented	61
2.5.2	Experimental results	62
2.5.3	Concluding remarks	64
2.6	Summary	71

3	Investigation of intensity correction	72
3.1	Fitting non-linear function	72
3.2	Histogram matching	78
3.3	Bi-functional mapping	78
3.4	Intensity inhomogeneity correction	83
3.5	Summary	85
4	Image registration with intensity correction	86
4.1	Intensity mapping and image registration	86
4.2	Affine 2-d registration with intensity correction	87
4.2.1	Linear intensity correction	87
4.2.2	Non-linear intensity correction	88
4.2.3	Additive inhomogeneity correction	93
4.2.4	Concluding remarks	96
4.3	Nonrigid B-Spline 1-d registration	96
4.3.1	Global non-linear intensity correction	96
4.3.2	Registration examples	97
4.3.3	Affine inhomogeneity correction	102
4.3.4	Registration examples	103
4.3.5	Concluding remarks	104
4.4	Summary	110
5	Conclusions and future work	111

Abstract

The research aims to develop means to increase the accuracy of radiation therapy treatment. Radiation therapy is planned based on radiation therapy planning computed tomography (RTPCT) and often validated using cone beam computed tomography (CBCT) images. The RTPCT image of a patient is used to outline a tumour that is to be irradiated. Before each radiation session the body is positioned on the treatment machine to match the geometry of the delivered beams with the planned treatment. The CBCT images are taken just before or after the treatment session to check if the patient position and internal anatomy is in agreement with the treatment plan. Registration of current CBCT to reference RTPCT image is required to visualise the effect of irradiation. Such registration allows effective irradiation according to the planned treatment. A particular goal is to develop algorithms for non-rigid image registration.

Intensity based image registration has been investigated because it is automatic and works without human expert interaction. Such an algorithm optimises an intensity based similarity measure between images to be registered with respect to spatial transformation. The similarity measure is a cost function which reaches maximum when images are aligned. This cost function is based on the average difference between intensity values of corresponding pixels. Contrary to a landmark based similarity measure, it does not depend on the choice of characteristic points. If images to be registered are very similar in terms of intensity values and differ only in spatial transformation, then sum of squared differences (SSD) can be chosen as a similarity measure. Choice of similarity measure is difficult in the case of multi-modal images. There are multi-modal similarity measures but they work properly only for some specific classes of images. To avoid using multi-modal similarity measures, the author decided to include intensity correction in the registration algorithm. The algorithm optimises SSD with respect to spatial transformation and intensity correction simultaneously. As a result the problem was shifted from choice of similarity measure to choice of model for intensity correction. In fact each multi-modal similarity measure contains implicit estimation of intensity mapping. However, the model of the mapping is fixed and may not fit to the class of images to be registered.

An image registration algorithm with intensity correction was developed.

The algorithm utilises common Levenberg-Marquardt optimisation. The author has chosen two dimensional affine and one dimensional B-Spline model as spatial transformation, as well as intensity correction models specific to CT images. They are global non-linear mapping and smooth local affine correction. The algorithm was tested experimentally using a wide class of simulated images and a limited class of medical images. Affine registration works properly even for deformations which exceed typical deformation encountered in medical practice. B-Spline registration works properly for small deformations and requires further development to increase capture range. Although there is a limitation in lack of discontinuity in intensity correction models, it can be overcome by implementing segmentation based intensity correction.

The idea of separating intensity correction mapping from similarity measure is shown to have advantages. Choosing intensity correction model can make the registration algorithm specific to the image class of interest.

Acknowledgement

I would like to thank directors of studies Dr Bogdan Matuszewski and Prof. Lik-Kwan Shark for their patience, support and guidance throughout the research and preparation of the thesis. Careful review by Dr Martin Varley and Dr Guoping Qiu allowed improve quality of the thesis.

Thanks to University of Central Lancashire I had opportunity to work on this fascinating project. I greatly appreciate afford of all research administrative staff especially: Clare Altham, Louise Price, Rachel Alen, Alison Brown, Anne Marie, Jean Chapman, June Hazzard, Carol Mills, Fiona Mair, Martin Varley, Peter Lumsden, Mike Holmes and their colleagues for allowing me to attend interesting lectures, workshops and conferences.

I would like to thank UK Engineering and Physical Sciences Research Council (EPSRC) for financial support. My work is part of Metrology Guided Radiation Therapy (MEGURATH) project whose collaborators motivated me to research. I especially appreciate Mahbubunnabi Tamal, Jian Kun-Shen, Zhang Yan and Andrzej Skalski. I was impressed by presentations given by Tomasz Zielinski and Christopher Moore.

I appreciate support of all people from Applied Digital Signal and Image Processing Research Centre (ADSIP). The discussion group moderated by Peter Dunne, Bogdan Matuszewski, Xin Chen and Wei Quan allowed me to learn essential techniques in image processing. Informal meetings with Marine Breuily, Andrzej Skalski, Sarma Volety, Kaushik Nag, Bartlomiej Papiez and other students were also invaluable. Leisure time spent with badminton team and mountaineering club helped me keep good mood and be better prepared to everyday work.

I would like to thank all administrative and technical staff of School of Computing Engineering and Physical Sciences especially Liz Higham and her colleagues, Apple Faisiri, Susan Laidlaw, Andrew Wildman, Warren Melia, Shabbir Badat, Mandeep Kandola, Christopher Finnigan, Brian Lee and Peter Bull. They keep bureaucracy, computers, printers, network and other equipment in perfect order.

Notations

CT, RTPCT, CBCT - computed tomography image, radiotherapy planning
computed tomography image, cone beam computed tomography image

MRI - magnetic resonance image

(x, y) - coordinates of pixel in 2-d image

L - 2-d grid of pixel positions of image $L = \{1, 2, \dots, w\} \times \{1, 2, \dots, h\}$
where w and h are width and height of image

L^c - 2-d continuous region $L^c = [1, w] \times [1, h]$ containing pixel grid L

u - reference image, the image which is the target for registration

v - source image, the image which during registration is to be transformed
to match reference image u

I_v - interpolation function which agree with image v at grid points and
extends v to whole L^c

Id - identity function

$\lfloor x \rfloor$, $\lceil x \rceil$, $[x]$ - the biggest integer which is $\leq x$, the smallest integer which
is $\geq x$, the closest integer value to x

$T(x, y) = (T^x(x, y), T^y(x, y)) = (x', y')$ - spatial transformation $T : R^2 \rightarrow R^2$

$v_T, T \circ v$ - superposition $v(T(x, y))$ of spatial transformation T and image
 v

T_P - spatial transformation parametrised by vector \mathbf{p}

T_A, T_B - affine spatial transformation, B-Spline spatial transformation

u - this subscript denotes simulated spatial transformation (e.g. T_u) or intensity distortion (e.g. F_u) in order to transform source image v to reference image u for experimental purposes

v - spatial transformation (e.g. T_v) or intensity distortion (e.g. F_v) which is estimated during registration to recover simulated one (simulated is denoted by index u)

F - non-linear intensity distortion

$A(x, y)$ - 2-d additive inhomogeneity distortion

$A(x)$ - multiplicative part of 1-d affine inhomogeneity distortion

$B(x)$ - additive part of 1-d affine inhomogeneity distortion

$b^l(\frac{x}{h} - i)$ - value of i -th B-Spline basis function at x , the basis function is of l -th degree and has h scaling factor

c_i - value of i -th B-Spline control point

$T_B(x) = \mathbf{b}^T(x) \cdot \mathbf{c}$ - inner product form of B-Spline function

$\mathbf{p} = [p_1, p_2, \dots, p_n]^T$ - vector of parameters

$f(\mathbf{p})$ - usually denotes the cost function in the least-square optimisation

$\mathbf{r}(\mathbf{p}) = [r_1(\mathbf{p}), r_2(\mathbf{p}), \dots, r_m(\mathbf{p})]^T$ - residual vector in the least-square cost function where $m = |L|$ is the number of pixels in image

$\nabla f(x, y) = \left[\frac{\partial f(x, y)}{\partial x}, \frac{\partial f(x, y)}{\partial y} \right]^T$ - gradient of $f(x, y)$ with respect to (x, y)

$\nabla_{\mathbf{p}} f(\mathbf{p}) = \left[\frac{\partial f(\mathbf{p})}{\partial p_1}, \frac{\partial f(\mathbf{p})}{\partial p_2}, \dots, \frac{\partial f(\mathbf{p})}{\partial p_n} \right]^T$ - gradient of $f(\mathbf{p})$ with respect to \mathbf{p}

$J_{\mathbf{r}}(\mathbf{p}) = J(\mathbf{p}) = \left[\frac{\partial r_i(\mathbf{p})}{\partial p_j} \right]_{i=1, \dots, m, j=1, \dots, n}$ - Jacobian of $\mathbf{r}(\mathbf{p})$ with respect to \mathbf{p}

SSD, NCC, CR, MI, NMI - similarity measures: sum of square differences, normalised cross-correlation coefficient, correlation ratio, mutual information, normalised mutual information

$D_{SSD}(u, v)$ - sum of square differences between images u and v which is used as a cost function for image registration

u, U - image u is treated as an instance of the random variable U

$p(u), p(u, v), p(v|u)$ - probability mass function, joint probability mass function, conditional probability mass function of image intensity

$E(X), Var(X), H(X)$ - expected value, variance and entropy of random variable X

$h(u), h_u(u_i)$ - intensity histogram of image u (u_i is i -th intensity level of u)

$H(u), H_u(u_i)$ - cumulative histogram of image u (integral of $h(u)$)

$h(u, v)$ - joint intensity histogram (JIH) of two images u and v

$f(u)$ - non-linear intensity mapping of image u

$f(u, \theta)$ - non-linear intensity mapping of image u parametrised by vector θ

$f_\epsilon(u), \epsilon = 1, 2$ - non-linear intensity mapping of image u parameterised by vector θ (when $\epsilon = 1$) or vector ψ (when $\epsilon = 2$)

$N(\mu, \sigma)$ - Normal distribution with μ mean and σ standard deviation

$G_\sigma(\cdot)$ - Gaussian function with σ standard deviation and 0 mean

Chapter 1

Introduction

1.1 Background

A typical radiotherapy treatment is carried out based on one radiation therapy planning computed tomography (RTPCT) [1] which provides a high quality 3-d image of the patient. Usually this data is captured for each patient before the treatment starts. The RTPCT image of the patient is used to outline the tumour that is to be irradiated. The radiation therapy treatment is planned on the basis of the outlined image, whereby the patient body is exposed to radiation according to a prescribed plan that can consist of many radiation sessions and can last for a few days. Before each radiation session the body is positioned in the treatment machine using point source markers. However, the positioning itself is not repeatable and also internal organs can deform during the treatment (for example bladder can have different volume each day). Therefore the radiation can be incorrect and can accidentally damage healthy tissue instead of irradiating the tumour.

To improve the accuracy of radiotherapy, additional computed tomography (CT) [1] images can be taken before or after radiation to give the most up-to-date information about current body shape and position. Contrary to RTPCT only cone beam computed tomography (CBCT) [1] capturing device can be installed on a treatment machine. Having radiation treatment and image capture units in the same machine, CBCT images can be taken directly before or after each radiation, thereby allowing interleave of imaging and radiation. Although RTPCT and CBCT images are of the same modality, they have different resolution, different noise ingredient, and different intensity mapping. They are also spatially deformed. To compare them efficiently they should be put in spatial correspondence so that the same positions on both images correspond to the same locations of the physical object. The process of aligning two images of the same object is called image registration.

1.2 Image registration

In biomedical domain there is a frequent need to compare images for diagnosis and analysis purposes. However, before comparison they should be registered. The images usually represent the same object (or objects of the same type) taken at different times, or by different devices. Object variability in time and different imaging devices introduce geometrical deformations that make comparison difficult. If in addition images are of different modality (they have different intensity mappings), then comparison is even more difficult. *Image registration* is a process of finding geometrical transformation that maps points from one image to points of another image so that the same features of the object match on both images.

Registration is frequently used to compare medical images of human internal body parts. Different acquisition techniques allow different physical properties to be measured. While computed tomography (CT) and magnetic resonance imaging (MRI) [3] are complementary techniques designed to see internal anatomy, positron emission tomography (PET) [3] shows the activity of internal structures.

1.2.1 Single and multi-modal images

Capturing devices can use different ways of imaging techniques like different frequency bands of electromagnetic waves (X-ray and optical cameras). In that case they produce images that have significantly different intensity mappings. Usually they provide complementary information about an object. Images taken by different imaging techniques and which significantly differ in intensity mapping are called *multi-modal* images. Figure 1.1 shows two colour components of the same RGB image which are multi-modal. They are spatially identical but differ in intensity mapping. Both of them provide complementary information about objects.

Images taken by the same imaging technique are called *single-modal images*. Intensity mappings of such images are approximately the same. Figure 1.2 shows two slices of a skull taken by RTPCT and CBCT imaging devices, respectively. They are single-modal. They differ in resolution, noise ingredient and are slightly deformed. For these two images the difference between their intensity mappings is minor.

1.2.2 Examples of image registration

Considering two images of the same object, taken by different capturing devices or at different times, although both images should be identical, they usually differ. The object can be spatially deformed due to object variation in time, position change of the image capturing device or difference in the capturing devices used. Apart from spatial deformation they can differ in

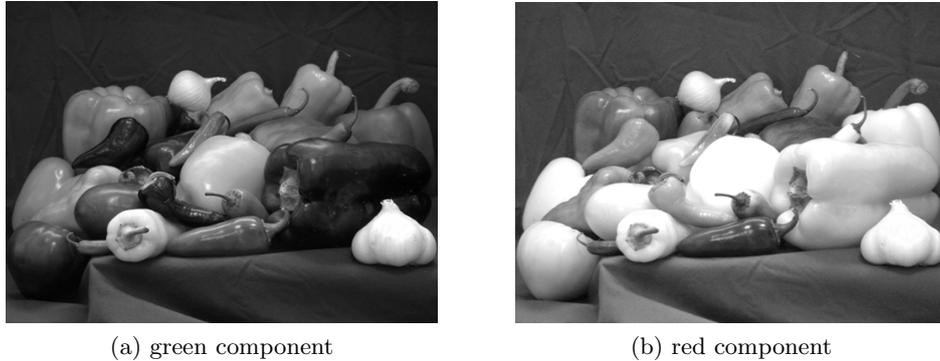


Figure 1.1: Example of multi-modal images - two colour components of the same RGB image

intensity mapping.

In practice there is often need to compare different images of the same object. Apart from medical images, satellite images of the same region form a good example. They can be taken by different imaging techniques and at different times. To compare them and catch significant changes, they have to be overlapped and aligned to match the geometrical position of real objects. This requires estimation of unknown geometrical transformation and applying this transformation to one of them.

Figure 1.3 shows an example of rigid images registration, based on geometrical transformation, where images (a) and (b) represent two satellite images of the same region taken at different times. After image registration with affine geometrical transformation, it is seen from image (c) that salient features like roads and buildings in the two images match each other.

Figure 1.4 shows another example of deformable image registration based on spatial transformation, where image (a) and image (b) differ only in spatial deformation. They are identical as far as intensity mapping is concerned. In this case registration consists of identifying unknown spatial transformation that can be applied to image (a) to make it same as image (b).

Registration of multi-modal images contains implicit estimation of intensity mapping. This makes registration process more difficult and requires statistical analysis of coincidence between intensities of the same pixel positions.

1.3 Similarity measure and intensity mapping

An essential component of intensity based registration is similarity measure. Sum of squared differences (SSD) [4],[5] is a common choice. It is simple

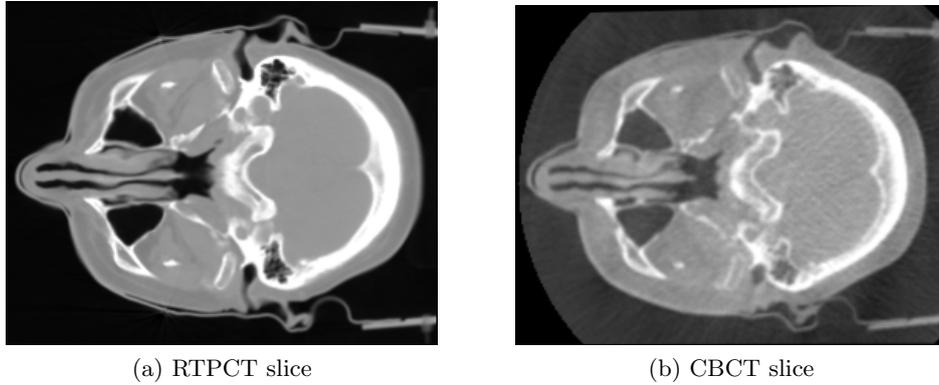


Figure 1.2: Example of single-modal images - slice of skull taken by different imaging devices using the same imaging technique

and fast. However it can be applied only to registration of single-modal images. For general SSD based registration, intensity correction is required, and it can be done by application of position dependent non-linear intensity mapping.

However, images can be registered without explicit intensity correction. Instead of SSD other similarity measures can be used.

Normalised cross-correlation coefficient (NCC) [6] can be applied to images for which one has intensity linearly distorted with respect to the other.

Normalised mutual information (NMI) [8], [9], [5] and correlation ratio (CR) [11] are most popular multi-modal similarity measures. NMI is more robust than SSD but it treats intensity values in a pure qualitative way [11]. There is no optimal choice of similarity measure because each similarity measure can outperform others on a specific class of images.



(a)



(b)

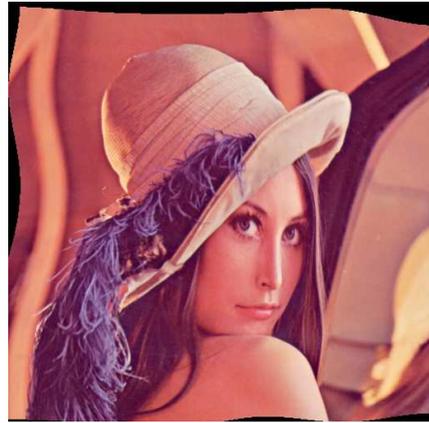


(c)

Figure 1.3: Example of rigid image registration, (a) (b) - two satellite images of the same region, (c) - both images overlapped with image (b) matched to image (a) by affine geometrical transformation (Source of images: Matlab Image Processing Toolbox).



(a) original image



(b) deformed image

Figure 1.4: Example of deformable image registration that consists of finding unknown spatial transformation to map image (a) to match image (b)

1.4 Aims of the research

CR and NMI contain implicit estimation of intensity mapping. The research question is if SSD together with intensity correction can be applied to multi-modal images as well.

The main goal of the research is to check how estimation of intensity mapping influences image registration, and this involves

- Review of registration methods.
- Review of different similarity measures and their robustness to differences of image intensity mapping.
- Examination of the sensitivity of the existing registration methods with respect to distortion of intensity mappings.
- Investigation of intensity correction methods.
- Investigation of influence of intensity correction on image registration.

Chapter 2

Investigation of image registration

2.1 Intensity and landmark based registration

Registration techniques can be divided into two general categories [5]: feature based registration and intensity based registration.

The first category relies on landmark or feature selection which is a process difficult to automate [4], and is normally implemented as a semi-automatic procedure supported by human expert interaction when necessary. During registration pairs of corresponding landmarks (or features) are selected. Then the displacement field is approximated by minimising sum of squared distances between corresponding landmarks. An additional regularisation term is added to cost function to make displacement smooth.

Intensity based registration depends on averaged similarity between pixel intensities instead of features which are difficult to select [4], [7]. Differences of intensity values of corresponding pixels are taken into account instead of distances between corresponding landmarks. It is based on optimising a cost function but the cost function renders the difference between pixel intensities. The method is sensitive to image contents and can fail for some specific images. However, in most practical cases, it works properly and is easy to automate.

The intensity based registration can be combined with feature based registration. This is usually achieved by using a more complex cost function [4]. Such blend gives a more reliable solution. The work focuses on intensity based registration.

2.2 Definitions

Definitions of grid, image, interpolation, spatial transformation and similarity measure are given in this section. Since all definitions and algorithms

can be generalised to 3-dimensional (3-d) images, only 2-dimensional (2-d) images are considered for simplicity.

2.2.1 Image and grid

A set denoted by $L = \{1, 2, \dots, w\} \times \{1, 2, \dots, h\}$ is called a *2-d grid*, and a function denoted by $v : L \rightarrow R$ is called a *2-d image* [4]. Grid L is a finite set of points uniformly distributed over a rectangle of size $w \times h$ with an incremental step of 1. Each element of L is represented by 2-dimensional *coordinates* (x, y) of some pixel, which has an *intensity value* $v(x, y)$. In other words, an image consists of $w \times h$ discrete pixels with coordinates (x, y) and intensity values $v(x, y)$. In practice image intensity values are limited to the range and precision of computer representation.

2.2.2 Interpolation

Image v is determined on a discrete grid from L . During spatial transformations grid positions of an image are mapped onto non-grid positions. To recover pixel values in non-grid positions *interpolation function* is introduced [4]. Interpolation function extends image domain from a discrete lattice $L = \{1, 2, \dots, w\} \times \{1, 2, \dots, h\}$ to a continuous rectangular 2-d region $L^c = [1, w] \times [1, h]$.

Function $I_v : L^c \rightarrow R$ which assigns a value to each point $(x, y) \in L^c$, and agrees with v in grid points $(x, y) \in L \Rightarrow v(x, y) = I_v(x, y)$ is called interpolation function. If image $v(x, y)$ is assumed to be a sampled version of a continuous image, then interpolation should recover the original continuous image. It is possible that the original image is bandlimited to frequency $2/T_{grid}$, where T_{grid} is the distance between grid points (Nyquist-Shanon theorem [40]). There are several ways to approximate interpolation functions [15].

The simplest example of a discontinuous interpolation function is *nearest neighbour interpolation*: $I_v(x, y) = v(\lfloor x \rfloor, \lfloor y \rfloor)$, where $\lfloor x \rfloor$ denotes the closest integer to x . Another example is *bilinear interpolation* which is continuous but not continuously differentiable:

$$I_v(x, y) = v(\lfloor x \rfloor, \lfloor y \rfloor) \cdot (1 - s) \cdot (1 - t) + v(\lfloor x \rfloor + 1, \lfloor y \rfloor) \cdot s \cdot (1 - t) + v(\lfloor x \rfloor, \lfloor y \rfloor + 1) \cdot (1 - s) \cdot t + v(\lfloor x \rfloor + 1, \lfloor y \rfloor + 1) \cdot s \cdot t \quad (2.1)$$

where $s = x - \lfloor x \rfloor$, $t = y - \lfloor y \rfloor$ and $\lfloor x \rfloor$ means the biggest integer not greater than x .

Commonly applied is basis function interpolation [15],[36],[18] which is continuously differentiable to certain degree:

$$I_v(x, y) = \sum_{i,j} b_{i,j}(x, y) \cdot c_{i,j} \quad (2.2)$$

where $b_{i,j}$ are basis functions and $c_{i,j}$ constant coefficients. If basis function is $\text{sinc}(x) = \sin(\pi \cdot x)/\pi \cdot x$, then $c_{i,j} = v(x_i, y_j)$. Otherwise $c_{i,j}$ can be obtained by solving equation (2.2) for grid points $(x, y) \in L$, [15]. Differentiability of I_v depends on the choice of basis functions.

For notational simplicity, v and I_v are not distinguished later on. v is often used as if it was extended to continuous domain by I_v and was determined for each $(x, y) \in L^c$.

2.2.3 Spatial transformation with reverse mapping

Spatial transformation $T : R^2 \rightarrow R^2$ is a function which transforms coordinates of image pixels and is denoted as follows:

$$T(x, y) = (T^x(x, y), T^y(x, y)) = (x', y') \quad (2.3)$$

Application of transformation T to grid points $(x, y) \in L$ usually gives non-grid points $T(x, y) \notin L$. In new image $v_T = v(T) = T \circ v$, transformed grid points of v do not correspond to grid points of v_T .

To deal with this problem, reverse mapping and interpolation function I_v are applied as illustrated in Figure 2.1 [4]. Grid positions of v_T can be transformed to source image v with T^{-1} . This procedure gives non-grid points in v . Then interpolation function I_v is used to find intensity values of these points.

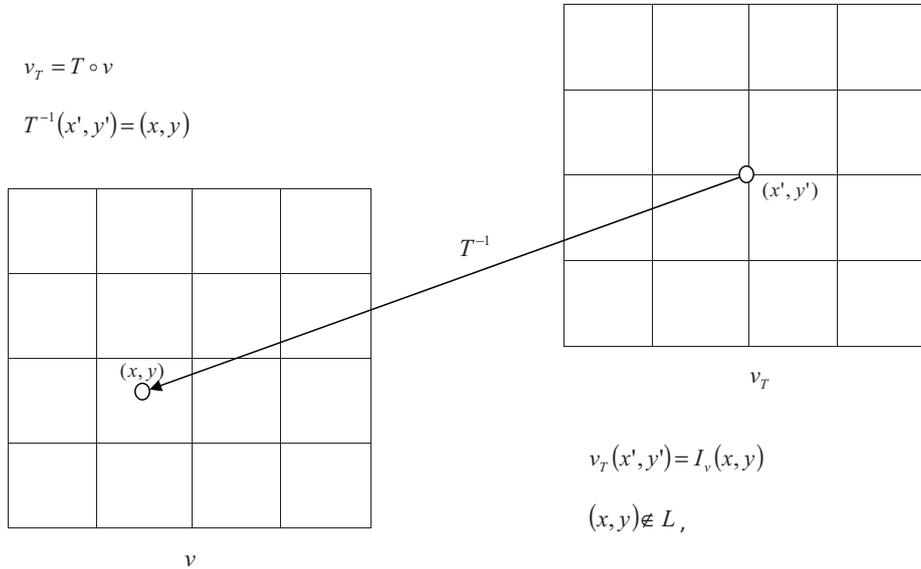


Figure 2.1: Spatial transformation T of image v onto v_T

Spatial transformation T can be decomposed as $T = Id + DF$ where Id is the identity function and $DF = T - Id$ is called the *displacement field* [4].

2.2.4 Image registration and similarity measure

A registration problem occurs when it is required to match two images containing the same object (or objects of the same type). Although object is the same, it can be spatially deformed. Registration consists of finding a suitable spatial transformation that makes two images *more similar*. Assuming two single-modal images with one called *reference* u and the second called *source* v , the registration algorithm look for *spatial transformation* $T : R^2 \rightarrow R^2$ which is applied to source image v to give *registered* image $v_T = T \circ v$, and which together with reference u form a pair of *similar* images. To formalise the notion of similarity, sum of squared differences (SSD), $D_{SSD}(u, v_T)$, is introduced as the similarity measure between the two images:

$$D_{SSD}(u, v_T) = \sum_{(x,y) \in L} (u(x, y) - v_T(x, y))^2 = \|u - v_T\|^2 \quad (2.4)$$

Now the registration problem can be formalised: given the similarity measure D_{SSD} , find spatial transformation $\hat{T} : R^2 \rightarrow R^2$ that minimises the similarity measure between the images, $D_{SSD}(u, v_T)$.

D_{SSD} is used as a cost function $f(T) = D_{SSD}(u, v_T)$ for optimisation during registration and represents the distance between two images. If both images u and v_T are identical, then $D_{SSD}(u, v_T) = 0$. If they differ, then $D_{SSD}(u, v_T) > 0$. The smaller the distance $D_{SSD}(u, v_T)$, the more similar are the images. For single-modal images, SSD is a good similarity measure, and also has the advantage of being very fast.

Usually registration algorithms consist of two stages, global and local. In the first stage global registration is applied to find the overall image position and orientation. Then local registration is applied as the second stage to refine final transformation. Both stages usually work in a multi-resolution framework.

2.3 Parametric registration

2.3.1 General algorithm

The name, parametric registration, is taken from the parametric nature of spatial transformation T_P . No matter the transformation is global or local, it is controlled by a vector of a limited number of parameters $\mathbf{p} = [p_1, p_2, \dots, p_n]$. At the input there are two images, *reference* image u and *source* image v . During registration source image v is transformed to $v_{T_P} = T_P \circ v$ known as the *registered* image. Parametric registration consists of

finding vector $\hat{\mathbf{p}}$ which minimises similarity measure $D_{SSD}(u, v_{T_P})$ [20], [21], [22].

Parametric registration can be decomposed into four separate components. They are: similarity measure or cost function (assume D_{SSD} for simplicity), interpolation $I_{v_{T_P}}$, spatial transformation model T_P and optimisation procedure. All these components are implemented separately but they are interleaved as shown in Figure 2.2. Reference and source images represent the same object but differ in some spatial deformation T_T . The registration can be seen as estimation of unknown deformation $T_P \approx T_T$ that transforms the source image to the reference one. The algorithm starts with identity transformation $T_P = I$ and iterates in the loop. Each iteration makes transformation closer to the target deformation.

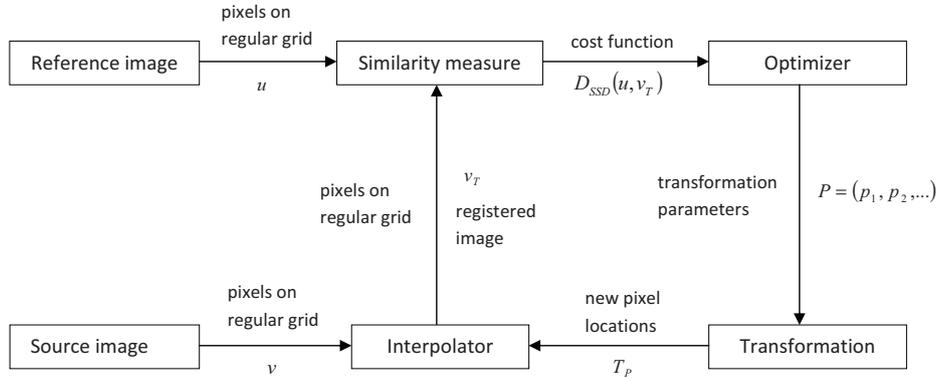


Figure 2.2: Intensity based registration algorithm

Registered and reference image are compared in the similarity measure block. As a result cost value $D_{SSD}(u, v_{T_P})$ is obtained that represents similarity between compared images. On the basis of the cost value and transformation model, optimiser calculates a new version of parameter vector \mathbf{p} . New pixel positions of the registered image are calculated. Since they are at non-grid positions, interpolation block $I_{v_{T_P}}$ is required to obtain the image at grid points. New registered image v_{T_P} is then compared to reference image u in the similarity measure block and gives smaller cost value $D_{SSD}(u, v_{T_P})$. Based on the new cost value and deformation model, optimiser calculates new transformation parameters \mathbf{p} and the whole procedure iterates again. When the end condition is met (e.g. insufficient improvement of cost value), the algorithm stops and the last found parameter values $\hat{\mathbf{p}}$ are taken as the solution.

The algorithm is applied few times for different resolutions of the input images. It starts from the lowest resolution and proceeds to the highest one.

This is called multi-resolution and is applied to recover bigger deformations.

2.3.2 Multi-resolution

The use of multi-resolution approaches is to accelerate registration and increase the capture range [3], [4], [14]. Increased capture range means that the registration algorithm is able to register significantly bigger deformation than those without using multi-resolution.

A registration algorithm minimises similarity measure between images and an optimisation procedure usually searches for the closest local minimum. As a result it can get stuck in some unwanted local minima. The higher the frequency components of image the higher the probability of getting stuck in a local minimum, which can be far from the global one. This significantly decreases the capture range of registration.

Maximum displacement possible for correct registration is limited by the half-period of the highest frequency component in the image. As it is shown in Figure 2.3, if two waves of the same frequency are to be registered, then the highest displacement possible for correct registration is exactly half-period of the wave frequency. It is called the *displacement limit* between two images. Applying a low-pass filter to images allows to recover a bigger displacement but at the expense of reduced accuracy. According to the Nyquist-Shanon theorem [40] low-pass filtering of an image can be followed by resampling at a frequency twice higher than the highest frequency in the filtered image. Such resampling reduces resolution which accelerates image processing.

Registration with multi-resolution consists of applying a typical registration algorithm to images at reduced resolution. The algorithm starts by registering images of lowest resolution and each next stage consists of registering images with doubled resolution. So registration is applied to images with gradually increasing resolution, until the highest - original resolution is reached. The results of registration from every stage is passed on as the initialization of the next stage. So in the first stage transformation is estimated roughly to register images with big displacements and then gradually stage by stage registration accuracy is increased together with the decreasing range of displacements. Pyramid is normally built based on images with resolutions reduced by powers of 2 and registration is applied in reverse order from the lowest resolution to the highest. Reducing image resolution by 2 means that every square has twice less pixels in each direction. Figure 2.4 shows an example of such a pyramid.

In the case of B-spline registration multi-resolution is more complicated [20]. Spatial transformation is determined by a grid of control points. There is no need to have a dense grid of control points for images with low resolution. So the number of pixels between two control points can be kept approximately constant for every image in multi-resolution pyramid.

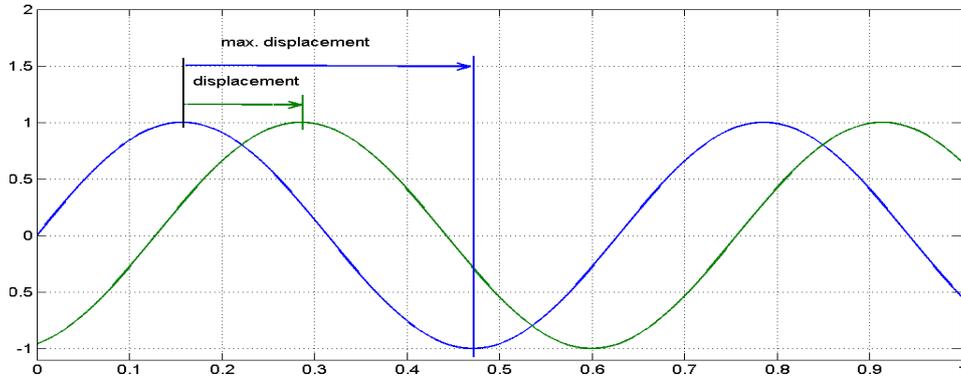


Figure 2.3: Two waves of the same frequency, where the green one is shifted by *displacement* with respect to the blue one. If displacement is less than the half period of the wave then it is possible to uniquely find the correct displacement. Otherwise the displacement error can be equal to the integer multiples of the period.

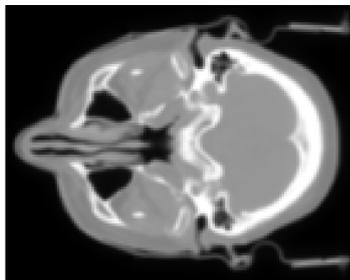
2.3.3 Deformation models

Spatial transformation is a significant part of the general image registration algorithm (Figure 2.2). The transformation model should be able to cover the considered class of geometrical deformations. The deformation can be classified with respect to range as global or local [5].

Global deformation models are parametric in nature, for example rigid, affine and projective transformations [13]. They deform whole image using a warping function controlled by a few parameters. The rigid and affine model is most commonly applied for medical image registration. The projective model is more general among linear models, but due to singularities, it is more difficult to implement and not justified in the case of medical images.

Local models can be divided into the following categories [19]:

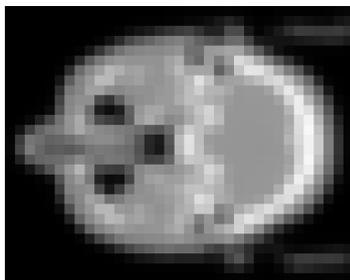
- physical non-parametric models based on partial differential equations [28]
 - linear elasticity [4]
 - viscous fluid flow [4], [23]
 - optical flow [24], [25], [26], [27]
- basis function parametric models
 - B-Spline [17],[20],[21],[22]
 - radial basis functions [4], [19]



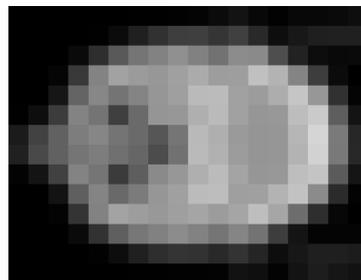
(a) original image



(b) 2x - reduction



(c) 3x - reduction



(d) 4x - reduction

Figure 2.4: Example of 3 level multi-resolution pyramid. (a) - original image, (b),(c),(d) - images with reduced resolution

– wavelets [19]

Local deformation models can change a small part of an image almost without influencing the remaining parts in the image. If they are parametric, then one parameter can influence only a local region. As a result there are more parameters than those in the global models.

Physical models usually act directly on the displacement field. In physical models it is common to impose a regularisation constraint to make the displacement field smooth.

B-Spline models are an attractive choice for medical image registration [17], [20], [21], [22]. In the case of smooth functions it has good approximation capabilities and is computationally efficient. Affine and B-Spline deformation models are described in the following Sections.

2.3.4 Affine transformation

An affine transformation expressed by $T_A = A \circ T_R$ is a superposition of linear transformation A and translation T_R . In 2-d this is described by the following equation:

$$T_A(x, y) = A \cdot \begin{bmatrix} x \\ y \end{bmatrix} + T_R \quad (2.5)$$

where

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix}, T_R = \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (2.6)$$

There are 6 parameters that uniquely describe T_A . Figure 2.5 shows an example of 2-d affine transformation of an image. In 3-d, matrix A has size 3×3 and vector T_R has size 3×1 . The main equation is unchanged.

The linear part of affine transformation can be decomposed as follows:

$$A = R(\theta) \cdot R(-\phi) \cdot D(\lambda_x, \lambda_y) \cdot R(\phi) \quad (2.7)$$

where

$$R(\alpha) = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix} \quad (2.8)$$

is rotation by α angle, and

$$D(\lambda_x, \lambda_y) = \begin{bmatrix} \lambda_x & 0 \\ 0 & \lambda_y \end{bmatrix} \quad (2.9)$$

is a diagonal matrix of anisotropic scaling by factors λ_x and λ_y . Matrices $R(\cdot)$ and $D(\cdot, \cdot)$ can be found by singular value decomposition (SVD):

$$\begin{aligned} A &= U \cdot D \cdot V^T = (U \cdot V^T) \cdot (V \cdot D \cdot V^T) \\ &= R(\theta) \cdot R(-\phi) \cdot D(\lambda_1, \lambda_2) \cdot R(\phi) \end{aligned} \quad (2.10)$$

where V and U are orthogonal matrices of rotations.



Figure 2.5: Example of 2-d affine transformation. (a) - original image, (b) - transformed image

2.3.5 B-Spline transformation

B-spline models are a good choice for deformable transformation [17], [18]. It is a special case of the basis function model and has good approximation - interpolation capabilities. Figure 1.4 shows an example of B-Spline transformation applied to deform an image. A common basis function is the cubic B-Spline. Finite support of B-Spline basis functions makes computation fast.

Spline model $T_P(x)$ is the way to approximate function $f(x)$ as a piecewise polynomial. If function $f(x)$ is to be approximated on interval $[a, b]$, then it is divided into a sufficient number of small intervals $[x_i, x_{i+1}]$ with $a = x_0 \leq x_1 \leq \dots \leq x_k = b$. Border points of the intervals are called knots. Polynomial pieces of degree l are defined on each interval to approximate $f(x)$ in each interval:

$$T_P(x) = \sum_{i=1}^l x^{i-1} \cdot c_{i,j}, \quad x \in [x_j, x_{j+1}] \quad (2.11)$$

Sum of all pieces forms the approximation function

$$f(x) \approx T_P(x) = \sum_{j=1}^k \sum_{i=1}^l x^{i-1} \cdot c_{i,j} \quad (2.12)$$

Coefficients $c_{i,j}$ of the polynomial pieces are chosen to keep $T_P(x)$ continuous and $l - 1$ differentiable at break points.

B-Spline models are equivalent to piecewise polynomials. A B-Spline curve is a linear combination of polynomial basis functions with one basis function spreading over few consecutive intervals. A B-Spline basis function is defined in terms of knots. Knots can overlap and overlapping knots are called multiple knots. Multiple knots reduce smoothness of the basis function and consequently the B-Spline curves. Multiple knots (except border points) are not taken into account because only smooth curves are considered. Knots and degree uniquely determine the basis function. For knots 0, 1, 2, 3, 4 the following 4-th degree (cubic) basis function $b^3(x)$ (Figure 2.6a) [17] is defined as follows:

$$b^3(x) = \begin{cases} (1-x)^3/6 & x \in [0, 1) \\ (3x^3 - 6x^2 + 4)/6 & x \in [1, 2) \\ (-3x^3 + 3x^2 + 3x + 1)/6 & x \in [2, 3) \\ (x^3)/6 & x \in [3, 4) \end{cases} \quad (2.13)$$

The B-Spline curve or function $T_B(x)$ is a linear combination of basis functions b^l with coefficients c_i :

$$T_B = \sum_i b^l \left(\frac{x}{h} - i \right) \cdot c_i \quad (2.14)$$

where $b^l(x)$ denotes the l -th degree B-Spline basis function, c_i are weights which are called control points, and h is a scaling factor which controls resolution of T_B . Coefficients c_i of T_B are not the same coefficients as in the case of T_P . They are weights for linear combination of basis functions.

Equation (2.14) can be rewritten in scalar product form:

$$T_B(x) = \mathbf{b}^T(x) \cdot \mathbf{c} \quad (2.15)$$

where $\mathbf{b}(\mathbf{x})$ is the vector of B-Spline basis functions given by

$$\mathbf{b}(x) = [b_1(x), b_2(x), \dots, b_n(x)]^T, \quad b_i(x) = b^l \left(\frac{x}{h} - i \right) \quad (2.16)$$

and \mathbf{c} is the vector of control points given by

$$\mathbf{c} = [c_1, c_2, \dots, c_n]^T \quad (2.17)$$

Coefficients c_i are called control points because they describe the shape of curve T_B . As shown in Figure 2.7b, the curve smoothly follows the control points. When all control points are identical and equal to 1, curve T_B is a constant, equal to 1. In this case T_B fulfills partition of unity $T_B = \sum_i b^i(x - i) = 1$ as shown in Figure 2.7a. Although knots are uniformly distributed, there are 2 multiple knots 0 and 1. All basis function which are

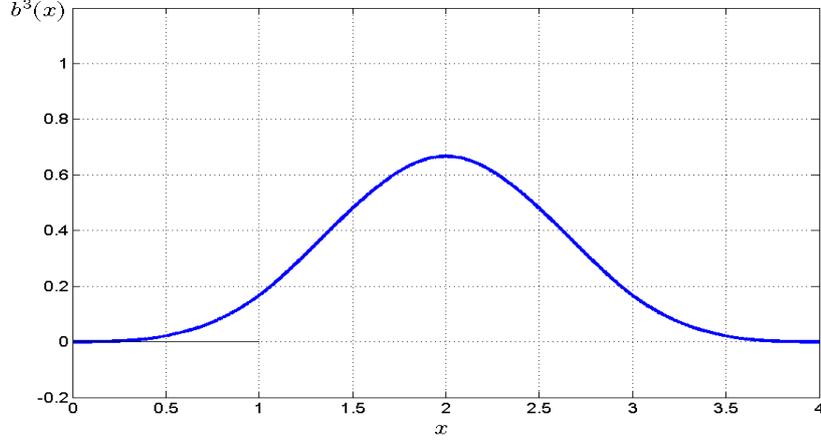


Figure 2.6: B-Spline cubic basis function $\beta^3(x)$ with knots 0, 1, 2, 3, 4, which are uniformly distributed along $[0, 4]$.

spanned over multiple knots are different than the others. In Figure 2.7a, there are 6 such basis functions and they adjoin the border points of the range $[0, 1]$.

B-spline basis functions are separable so in 2-d they are products of 1-d basis functions with $b^l(x, y) = b^l(x) \cdot b^l(y)$. 2-d cubic basis function $b^3(x, y)$ is shown in Figure 2.8.

Equation (2.14) in 2-d can be rewritten as follows:

$$T_B(x, y) = \sum_{i,j} b^l\left(\frac{x}{h_x} - i\right) \cdot b^l\left(\frac{y}{h_y} - j\right) \cdot c_{i,j} \quad (2.18)$$

$$T_B(x, y) = \mathbf{b}^T(x, y) \cdot \mathbf{c} \quad (2.19)$$

where the vector of B-Spline basis function is

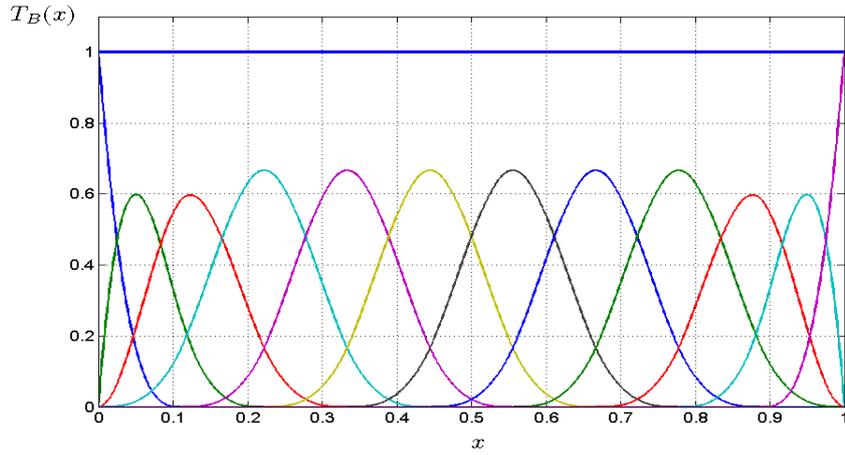
$$\mathbf{b}^T(x, y) = [b_{1,1}(x, y), b_{1,2}(x, y), \dots, b_{m,n}(x, y)] \quad (2.20)$$

$$b_{i,j}(x, y) = b^l\left(\frac{x}{h_x} - i\right) \cdot b^l\left(\frac{y}{h_y} - j\right) \quad (2.21)$$

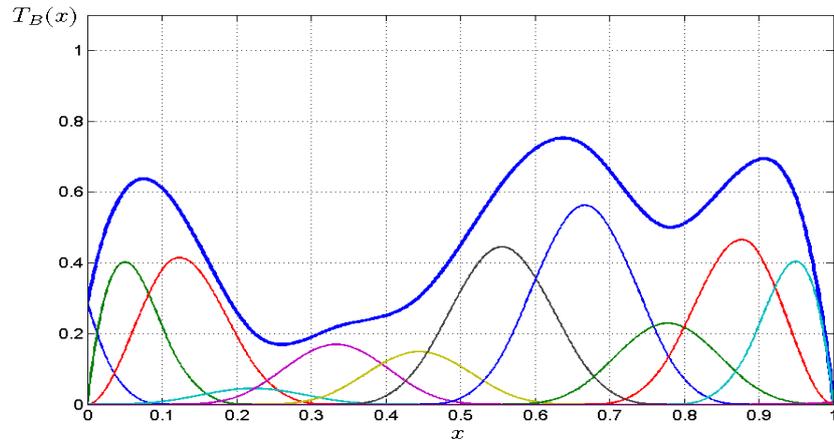
where the vector of control points is

$$\mathbf{c}^T = [c_{1,1}, c_{1,2}, \dots, c_{m,n}] \quad (2.22)$$

Components of vector \mathbf{b} and \mathbf{c} are sorted according to lexicographical order. Figure 2.9 shows an example of a B-Spline surface determined by control points.



(a) constant line and its basis functions



(b) curve and its basis functions

Figure 2.7: Examples of curves which are formed as a linear combination of basis functions. Knots are uniformly distributed except borders. Each basis function is vertically scaled by its control point. In (a) all control points are equal to 1, and in (b) control points correspond to curve.

During image registration deformable transformation is represented as two B-spline components for each axis:

$$T_B(x, y) = (T_B^x(x, y), T_B^y(x, y)) \quad (2.23)$$

The transformation is implemented as additive displacement field $T = Id + T_B$. All considerations can be generalised to 3-d.

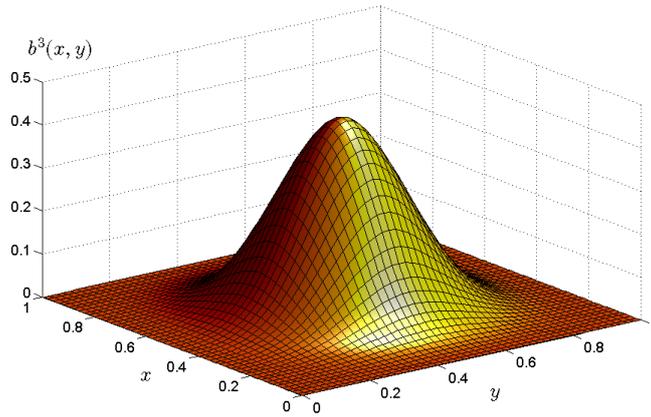


Figure 2.8: 2-d cubic B-Spline basis function $b^3(x, y) = b^3(x) \cdot b^3(y)$.

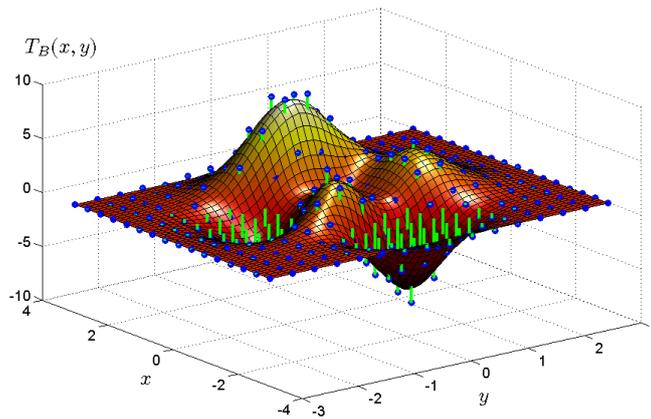


Figure 2.9: B-Spline surface and its control points (green bars).

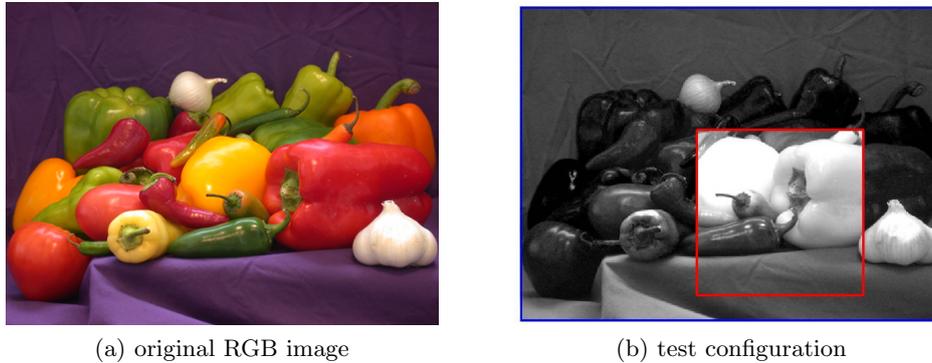


Figure 2.10: Images used in similarity measures experiment. (a) - original RGB image whose components were used for tests, (b) - small rectangular piece of red component translated over the background of blue component

2.3.6 Similarity measure

In Section 2.2.4 and Section 2.3.1 definitions and function of similarity measures were presented. If images to be registered have very similar intensities for corresponding pixels, then SSD is a good choice. There are various possibilities to choose a similarity measure. If intensity mapping between images is linear, the normalised cross-correlation coefficient (NCC) [6] can be applied. If intensity mapping is non-linear, then a suitable choice is the correlation ratio (CR) [11]. Finally the most general one is the normalised mutual information (NMI) [8], [9]. NMI exploits the entropy notion to measure statistical dependence between intensities of corresponding pixels.

Similarity measure and modality of images

In this Section different similarity measures are presented to demonstrate their properties using a simple experiment. The experiment consists of translating a small rectangular image v_T over bigger background image u and comparing their similarity. The translation is inside the image so that the area of overlap is constant. The test images are based on the blue and red components of the same RGB image (Figure 2.10). The blue component was used as the background image. Plots of each similarity measure versus translation are presented for each similarity measure. In the case of the single-modality experiment the small rectangular image was cut out of the blue component and in the case of multi-modality experiment the small rectangular image was cut out of the red component.

Sum of squared differences (SSD) SSD is an intensity based similarity measure which means that it is based on the average difference between intensities of corresponding pixels. Contrary to a landmark based similarity measure, it does not depend on the choice of characteristic points so it is easy to automate. *SSD* is sensitive to image contents, but it works in most practical cases for single-modal images. The definition is presented below:

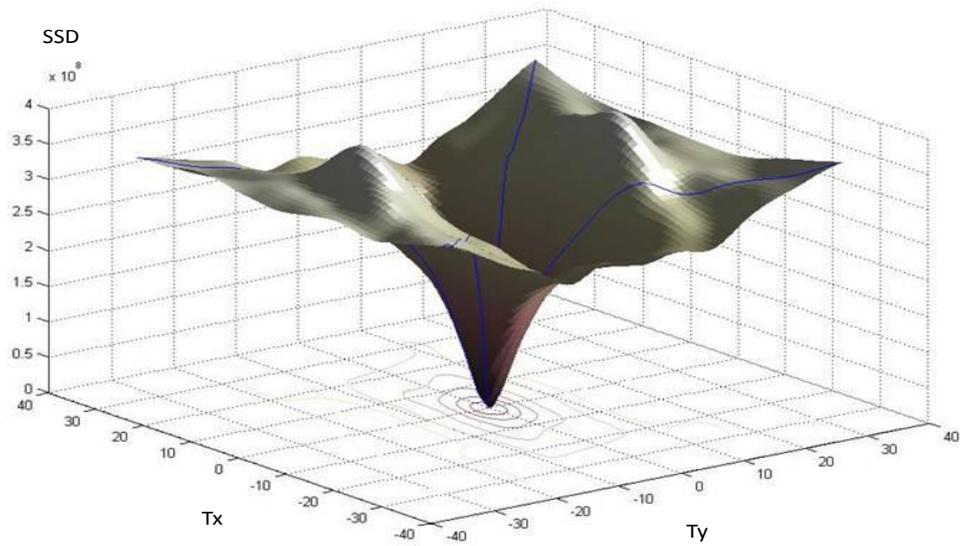
$$SSD(u, v_T) = \sum_{(x,y) \in L} (u(x, y) - v_T(x, y))^2 = \|u - v_T\|^2 \quad (2.24)$$

In (2.24) (x, y) denotes pixel position and L denotes the set of all pixels. On the plot shown in Figure 2.11, it is shown that SSD fails to give minimum difference at zero translation in the case of multi-modal images.

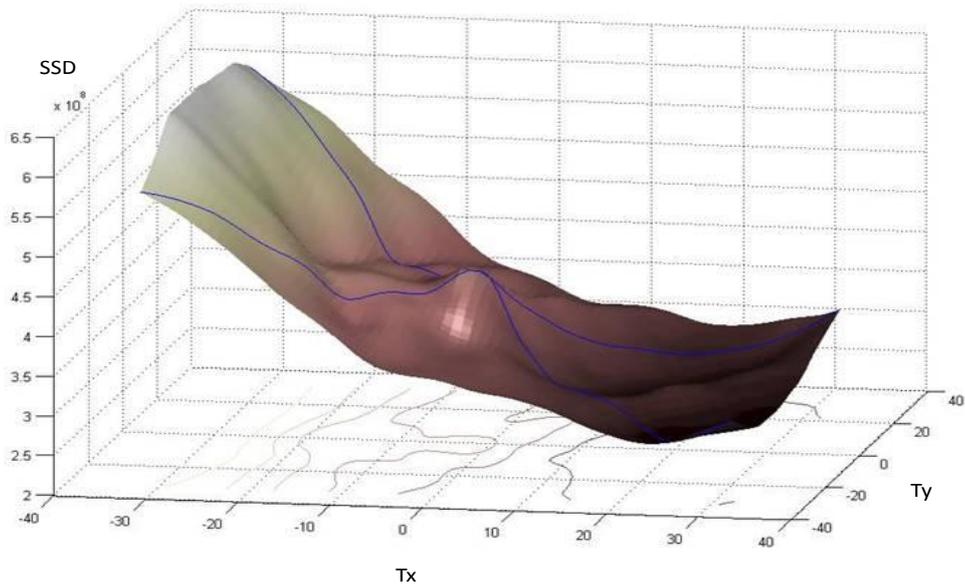
Normalised cross-correlation coefficient (NCC) NCC is insensitive to changes of brightness and contrast [6] and is defined as follows:

$$NCC(u, v_T) = \frac{Cov(u, v_T)^2}{Var(u) \cdot Var(v_T)} \quad (2.25)$$

NCC is unsuitable to compare multi-modal images as shown in Figure 2.12.



(a) single-modal images



(b) multi-modal images

Figure 2.11: Sum of squared differences (SSD) versus translation. The smaller the value of SSD, the more similar are the images. In the case of single-modal images shown in (a) SSD attains minimum at $(0,0)$ translation, but in the case of multi-modal images shown in (b) SSD does not have minimum at $(0,0)$ translation, which is incorrect.

Image as random variable Before describing the next similarity measure, stochastic interpretation of images is presented, whereby pixel intensities of images u and v are assumed to be instances of random variables U and V . Probability mass functions $p(u)$, $p(v)$ and $p(u, v)$ are calculated from intensity histograms $h(u)$, $h(v)$ and joint intensity histogram $h(u, v)$, respectively. Intensity levels are enumerated by integer index $i = 1, 2, \dots, l$ where l is the number of intensity levels and u_i denotes the i -th intensity level.

Given image u or v , probability $p(u_i)$ or $p(v_i)$ are computed by normalising the corresponding intensity histogram:

$$p(u_i) = \frac{h(u_i)}{\sum_j h(u_j)}, \quad p(v_i) = \frac{h(v_i)}{\sum_j h(v_j)} \quad (2.26)$$

Given images u and v , joint probability $p(u_i, v_j)$ is given by normalising the joint intensity histogram:

$$p(u_i, v_j) = \frac{h(u_i, v_j)}{\sum_{j,k} h(u_j, v_k)} \quad (2.27)$$

Correlation ratio (CR) If intensities between registered images are mapped by a global non-linear function, then the correlation ratio (CR) is a good choice for similarity measurement [11]. Contrary to SSD and NCC, it can be applied to a wide class of multi-modal images. Figure 2.13 shows examples of correct similarity measure plots for both single and multi-modal images. The maximum is at (0,0) translation and comparing to Figure 2.11 and Figure 2.12 the CR plots have better shapes with less local extremes which makes optimisation easier.

The CR definition is based on the assumption that when images are registered intensity mapping between corresponding pixels of both images is predictable and independent of coordinates. This mapping function is denoted by F , and the average dispersion around mean value $Var(V - F(U))$ is chosen as a measure of similarity between V and $F(U)$. However, function F is unknown and has to be estimated by minimisation

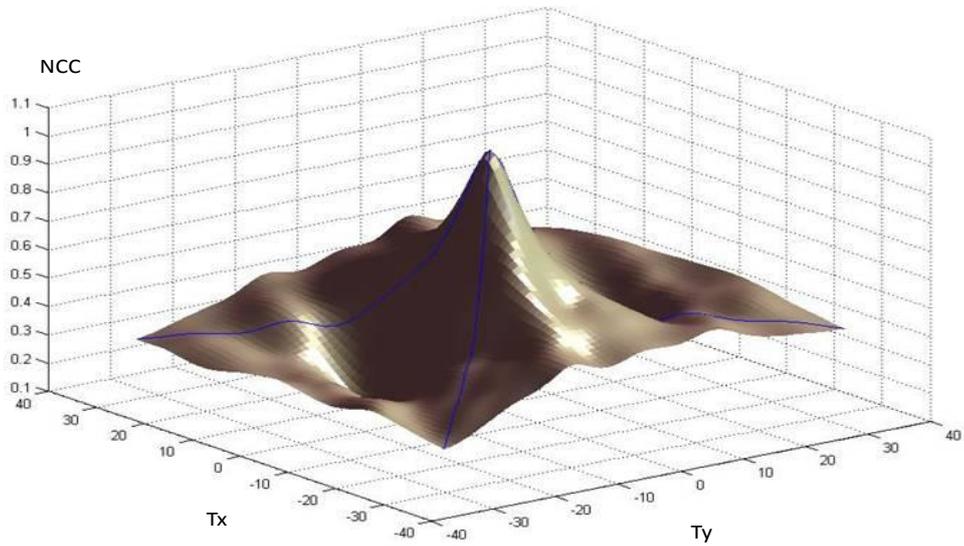
$$F = \arg \min_f Var(V - f(U)) \quad (2.28)$$

(2.28) is an unconstrained regression problem [36] which has the following solution:

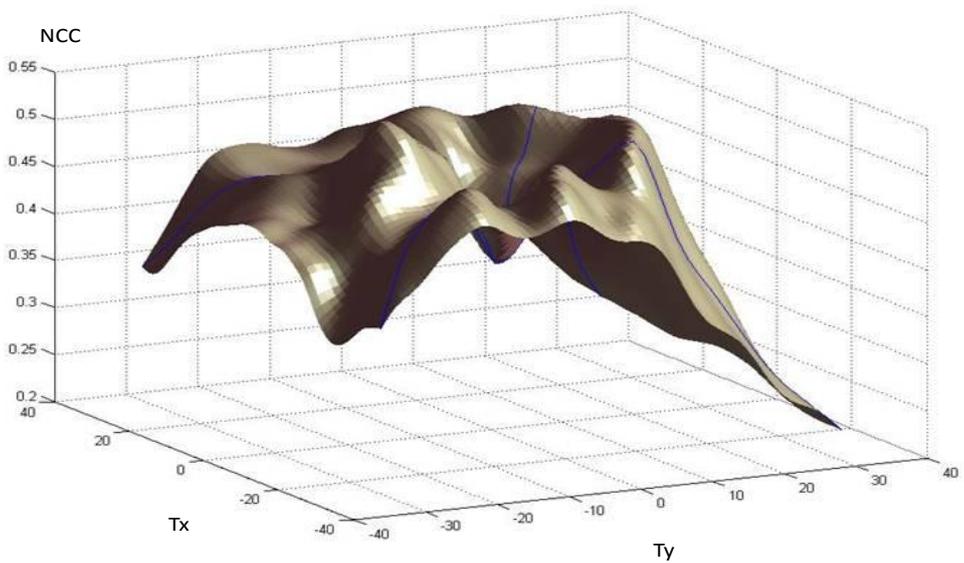
$$F(U) = E(V|U), \quad F(u) = \sum_v v \cdot p(v|u) \quad (2.29)$$

$Var(V)$ can be decomposed according to the orthogonality principle [12]:

$$Var(V) = Var[E(V|U)] + Var[V - E(V|U)] \quad (2.30)$$

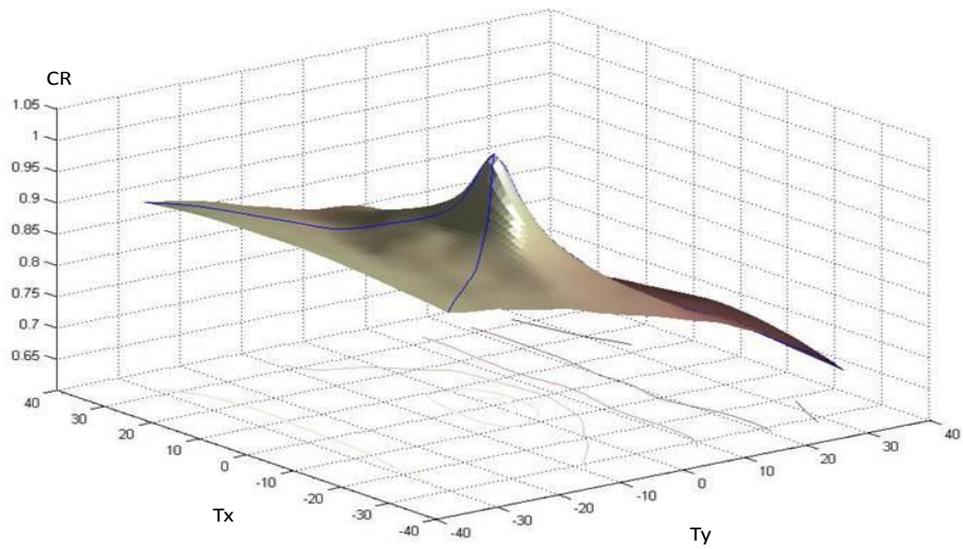


(a) single-modal images

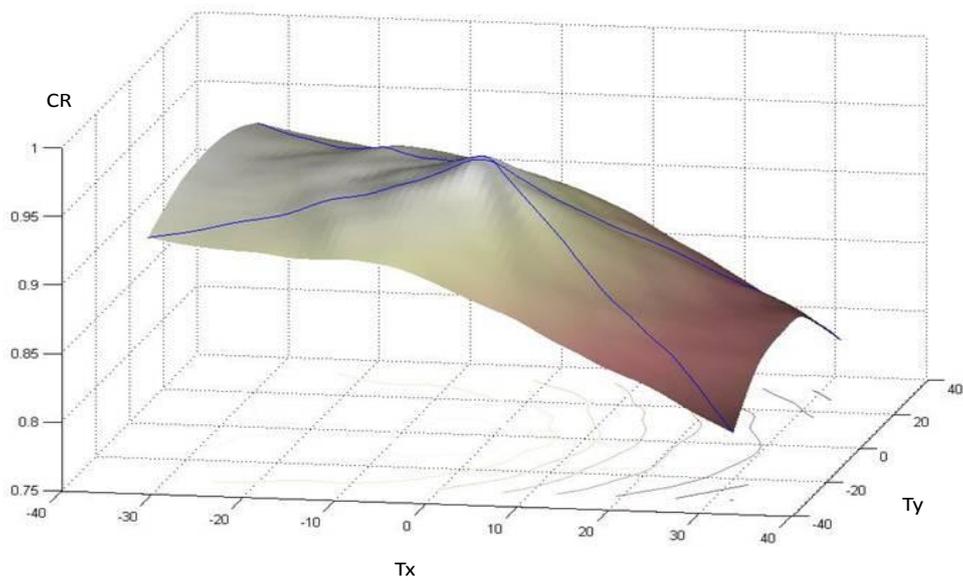


(b) multi-modal images

Figure 2.12: Normalised cross-correlation coefficient (NCC) versus translation. The higher the value of NCC, the more similar are the images. In the case of single-modal images shown in (a) NCC attains maximum at $(0,0)$ translation, but in the case of multi-modal images shown in (b) NCC does not have maximum at $(0,0)$ translation, which is incorrect.

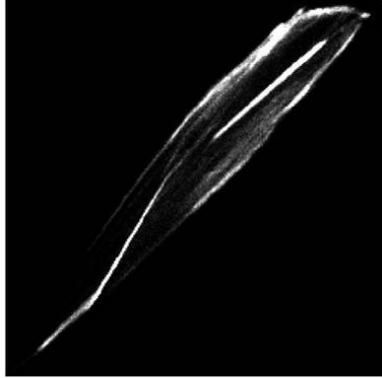


(a) single-modal images

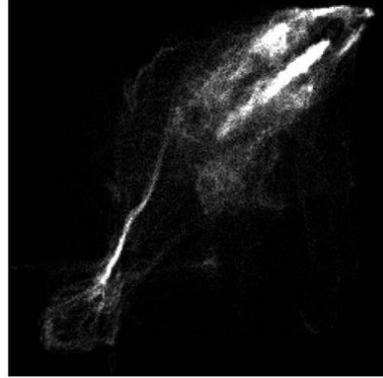


(b) multi-modal images

Figure 2.13: Correlation ratio (CR) versus translation. The higher the value of CR, the more similar are the images. CR attains maximum for (0,0) translation in both cases of single and multi-modal images. CR function has a better shape than SSD (Figure 2.11) and NCC (Figure 2.12) because it has less local extremes.



(a) aligned images



(b) translated images

Figure 2.14: Joint intensity histogram (JIH) of aligned and misaligned multi-modal images. For aligned images shown in (a) JIH is sharp which means that they are correlated. For misaligned images shown in (b) JIH is blurred which means that they are less correlated.

With the components of summation being orthogonal, the first term $Var[E(V|U)]$ measures how much V is predicted by U , and the second term $Var[V - E(V|U)]$ measures functional independence of V from U . Having in mind (2.30) and the above interpretation, the authors of [11] defined the following similarity measure:

$$CR(V|U) = \frac{Var[E(V|U)]}{Var(V)} = 1 - \frac{Var[V - E(V|U)]}{Var(V)} \quad (2.31)$$

Denominator $Var(V)$ was introduced to normalise the CR value. The CR similarity measure is asymmetric. Practical experiments [11] showed that CR can be applied to a wide class of multi-modal images.

Mutual information (MI) While CR assumes that intensities between images to be registered are related by function, the definition of mutual information (MI) uses a more general assumption. To assess the similarity between images, the authors of [8], [9] estimated the joint intensity histogram (JIH) instead of mapping function. JIH is more general because it represents a relation which is not restricted to function. Figure 2.14 shows JIH for aligned and misaligned images. Misalignment disperses JIH and makes it less sharp. Less sharp JIH means that images are less correlated. As a result level of dependency between images can be deduced from their JIH.

It is assumed that intensities of images u and v are instances of random variables U and V . Probabilities $p(u)$, $p(v)$, $p(u, v)$ are calculated from JIH.

Relative entropy or the Kullback-Leibler distance between $p(u)$ and $q(u)$ is defined as follows [40]:

$$D(p||q) = \sum_u p(u) \log \frac{p(u)}{q(u)} \quad (2.32)$$

The relative entropy, $D(p||q)$, is a measure of the inefficiency, assuming that the distribution is p when the true distribution is q . Mutual information $MI(u, v)$ is the Kullback-Liebler distance between joint probability $p(u, v)$ and probability $p(u) \cdot p(v)$ which assumes that U and V are independent random variables:

$$MI(u, v) = D(p(u, v)||p(u) \cdot p(v)) = \sum_{u,v} p(u, v) \cdot \log \frac{p(u, v)}{p(u) \cdot p(v)} \quad (2.33)$$

Mutual information is a robust similarity measure which can be applied to a wide class of multi-modal images. In Figure 2.15 the plot of MI versus translation is presented. The plot has maximum at $(0, 0)$ translation and a better shape than SSD and NCC.

However, MI depends on area of overlap. The author of [10] introduced normalised mutual information (NMI). NMI is less sensitive to overlap than MI. To demonstrate this sensitivity, a simple experiment was conducted. Red and green components of RGB image (shown in Figure 2.10a) was translated by 4 pixels with respect to each other. Then the vertical stripe positioned at the centre of the horizontal coordinate was cut out of both images and compared using MI and NMI. The width of the stripe was varied and dependence of MI and NMI versus area of overlap is shown in Figure 2.16.

In order to define NMI, the entropy notion is introduced [40]. Given random variable X , the entropy of the variable is defined as:

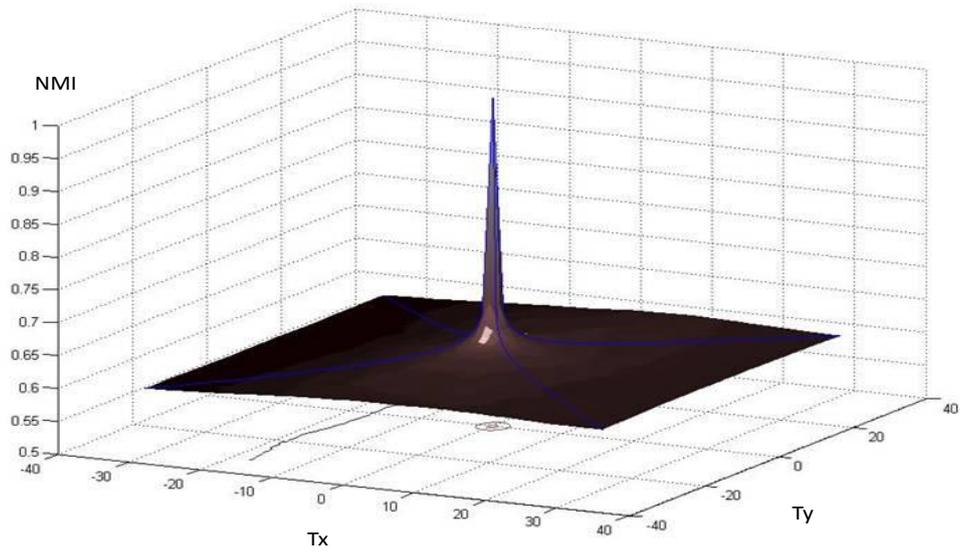
$$H(X) = - \sum_x p(x) \cdot \log p(x) \quad (2.34)$$

Entropy measures the uncertainty of a random variable. The greater the entropy, the more unpredictable is the random variable. A more unpredictable variable can convey more information. Shannon showed that it is possible to construct a code for X that will take $H(X)$ bits on the average [40]. Joint Entropy $H(X, Y)$ is defined as follows:

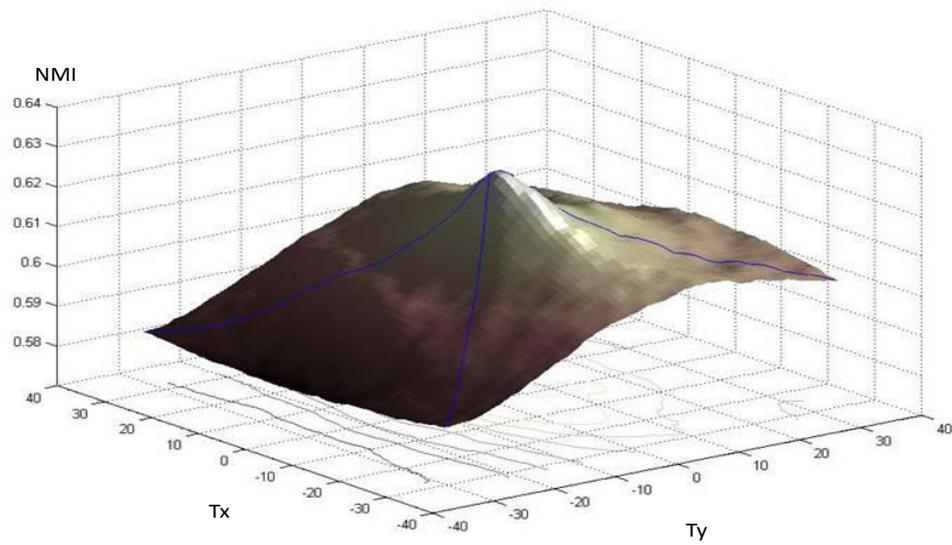
$$H(X, Y) = - \sum_{x,y} p(x, y) \log p(x, y) \quad (2.35)$$

Now the definition of mutual information can be reformulated as:

$$MI(u, v) = H(U) + H(V) - H(U, V) \quad (2.36)$$



(a) single-modal images



(b) multi-modal images

Figure 2.15: Mutual information (MI) versus translation. The higher the value of MI, the more similar are the images. For both cases (a) and (b) maximum is at $(0,0)$ translation.

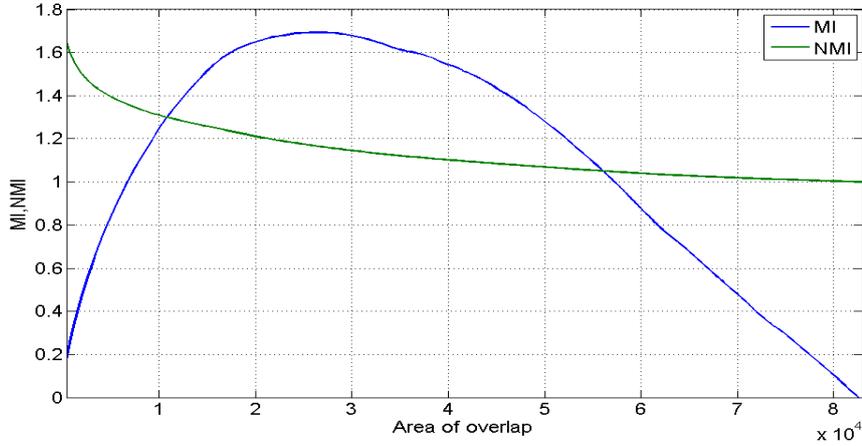


Figure 2.16: Dependence of MI (blue) and NMI (green) on overlap.

The author of [10] discovered that the following similarity measure is less sensitive to area of the overlap:

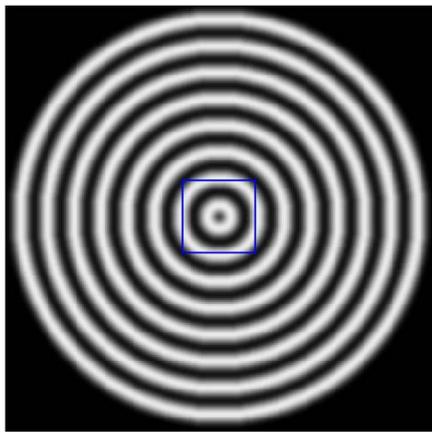
$$NMI(U, V) = \frac{H(U) + H(V)}{H(U, V)} \quad (2.37)$$

The measure is called normalised mutual information (NMI).

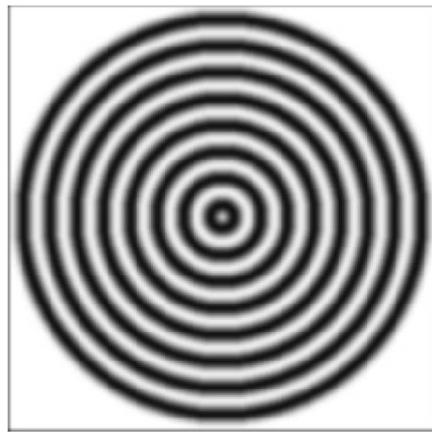
Interpolation and similarity measure During registration, one of the images is spatially transformed (Section 2.2.3). For the transformed image, its pixel values have to be interpolated (Section 2.2.2). There are many interpolation methods but most of them are sources of grid artefacts for similarity measures.

Grid artefacts make optimisation difficult. To avoid grid artefacts random sampling was introduced [16]. Normally only the transformed image has to be interpolated. When random sampling is used both images are interpolated. Intensity values which are taken to estimate JIH are interpolated at random positions which are independent of image grid. More samples can be taken than pixels to avoid grid artefacts and increase resolution.

To show the influence of random sampling on NMI, a simple experiment was conducted. Figure 2.17 shows images used in the experiment. A small square image was extracted from one of them and translated over the other to compare similarity. Graphs of NMI versus translation for different number of samples are presented in Figure 2.18. Comparison of Figure 2.18a and Figure 2.18d shows that random sampling can remove grid artefacts. Comparison of Figure 2.18b,c and Figure 2.18d shows that random sampling can increase resolution of NMI.

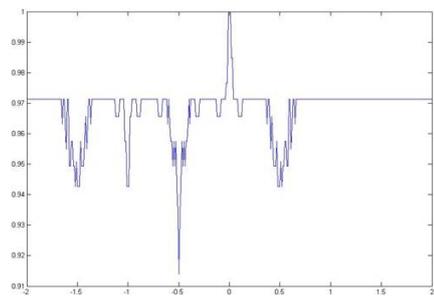


(a)

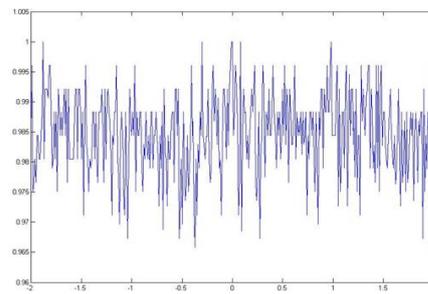


(b)

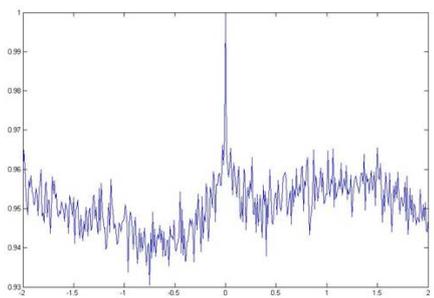
Figure 2.17: Images used for sampling experiment. Small rectangle shown in (a) is translated over (b) as a background. The results are presented in Figure 2.18.



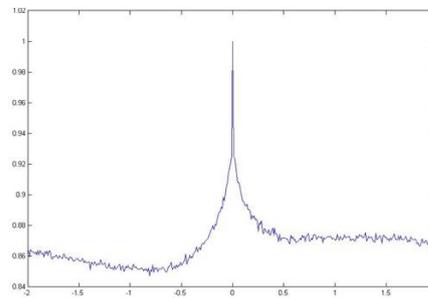
(a) grid artefacts for linear interpolation



(b) 1 sample per pixel



(c) 4 samples per pixel



(d) 32 samples per pixel

Figure 2.18: Influence of random sampling on NMI. (a) - linear interpolation causes grid artefacts, (b),(c),(d) - increasing number of random samples removes grid artefacts, makes graph smoother and increases resolution.

Comparison of NMI and CR Each similarity measure is suitable for some classes of images and can be outperformed by other similarity measures on a different class of images [11]. Although NMI appears to be the most general similarity measure, it does not work for all images as shown in the following example.

Two similar images are shown at the top of Figure 2.19. Although they represent the same pattern: three sectors with a light area sandwiched between two dark areas, the second image in the middle of Figure 2.19 is modulated by 10 pixel wide vertical stripes with each having a constant intensity. As shown in the intensity graph of both images along the horizontal direction at the bottom of Figure 2.19, the intensities of 10 pixel wide vertical stripes in the second image change monotonously within each section. Since the pattern of both images roughly agree, they should have a unique matching point.

Figure 2.20 shows graphs of NMI, CR and SSD versus translation for images shown in Figure 2.19. Although NMI is more robust than CR and SSD, it treats intensity values in a pure qualitative way [11]. As a result NMI fails to register the images compared with CR and SSD which have a unique extremum for the matching point.

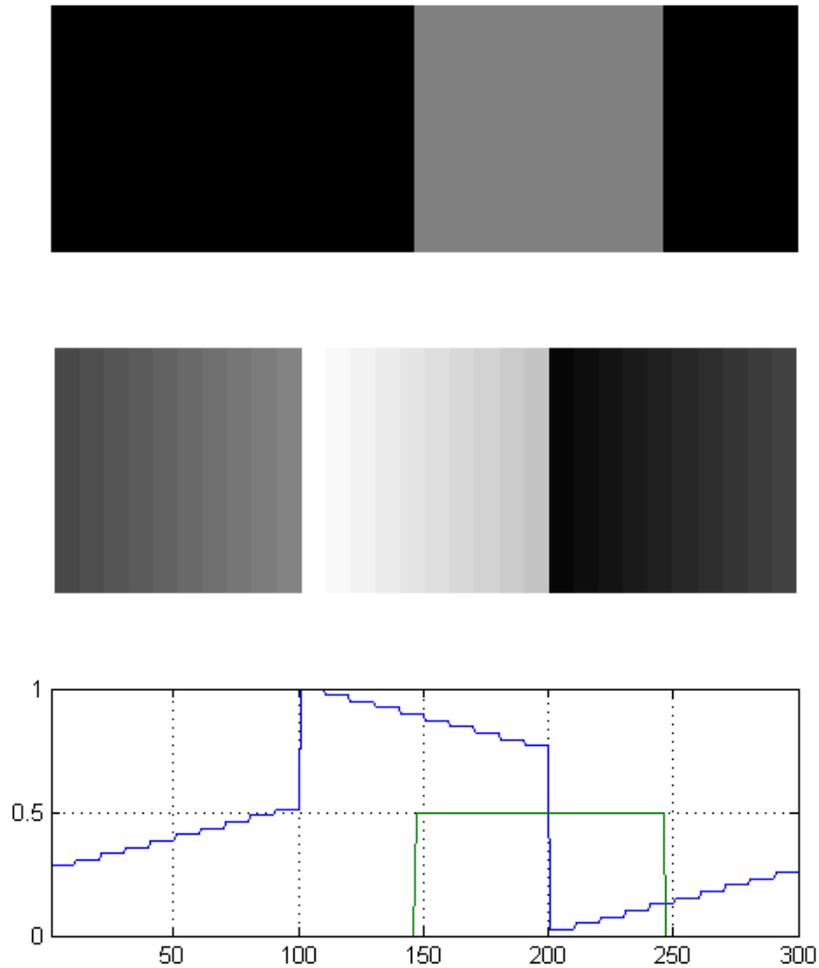


Figure 2.19: Comparison of CR and NMI. Vertical grey stripe from the top image is translated horizontally and compared to middle image. Graphs of NMI, CR and SSD versus translation are presented in Figure 2.20. Top image shows a gray vertical stripe on black background. Middle image is similar to top one but consists of 10 pixel wide vertical stripes. Every stripe has a unique intensity. Intensity of stripes are modulated so that middle image resembles top one. Graph of intensity of both images versus horizontal coordinate i presented at the bottom with green line showing the intensity profile of the top image, and blue line showing the intensity profile of the middle image.

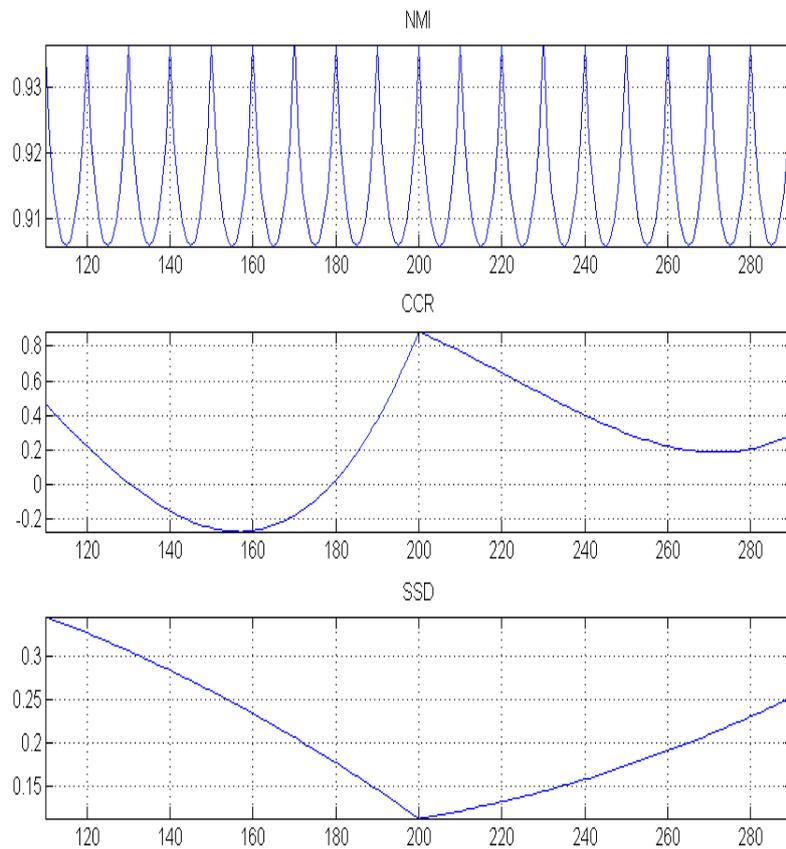


Figure 2.20: Graphs of NMI, CR and SSD versus translation for experiment based on images from Figure 2.19. NMI is periodic and does not render intensity pattern of compared images. On the other hand CR and SSD have an extremum for the matching point.

2.3.7 Image registration and optimisation

Intensity based image registration of two images u and v consists of minimising cost function $D_{SSD}(u, v_T)$ where registered image $v_T(x, y) = v(T(x, y)) = v(x', y')$ is a transformed version of source image $v(x, y)$. By assuming that T is parameterised by vector of parameters $\mathbf{p} = [p_1, p_2, \dots, p_k]^T$, the transformation is denoted as $T_P(x, y) = T(\mathbf{p}, x, y) = (x', y')$. As a result cost function $D_{SSD}(u, v_{T_P})$ is also parameterised:

$$f(\mathbf{p}) = D_{SSD}(u, v_{T_P}) = \sum_{(x,y) \in L} (u(x, y) - v_{T_P}(x, y))^2 = \|u - v_{T_P}\|^2 \quad (2.38)$$

Pixels can be ordered so that their positions (x_i, y_i) can be replaced by their integer index i . To simplify notation substitution $r_i(\mathbf{p}) = v(x_i, y_i) - v_{T_P}(x_i, y_i)$ yields the cost function in the form $f(\mathbf{p}) = \sum_i r_i^2(\mathbf{p})$. After changing notation the cost function is given in the form of sum of squares:

$$f(\mathbf{p}) = \frac{1}{2} \sum_i^m r_i^2(\mathbf{p}) \quad (2.39)$$

where m is the number of pixels in the image. As a result intensity based registration becomes a least squares optimisation problem [29], [31].

Least squares optimisation

Least squares minimisation is a task of finding a local minimum of cost function f which has the form:

$$f(\mathbf{p}) = \frac{1}{2} \sum_{i=1}^m r_i^2(\mathbf{p}) = \frac{1}{2} \|\mathbf{r}(\mathbf{p})\|^2 \quad (2.40)$$

where $\mathbf{p} = [x_1, x_2, \dots, x_n]^T$ is a vector, and $\mathbf{r} = [r_1, r_2, \dots, r_m]^T$ is a valued function $\mathbf{r} : R^n \rightarrow R^m$. The values of r_i are called residuals. Gradient based minimisation is considered. The optimization starts with some arbitrary location \mathbf{p}_0 and searches for the closest local minimum.

Usually minimisation is an iterative algorithm which produces a series of points $\mathbf{p}^1, \mathbf{p}^2, \dots, \mathbf{p}^t$ such that $f(\mathbf{p}^1) \geq f(\mathbf{p}^2) \geq \dots \geq f(\mathbf{p}^t)$ which are convergent to local minimum $\hat{\mathbf{p}}$ of objective function f . The series is built by taking steps in descent direction \mathbf{h} such that $f(\mathbf{p}^{t+1}) = f(\mathbf{p}^t + \lambda \cdot \mathbf{h}) < f(\mathbf{p}^t)$ where λ is step length. In gradient based algorithms descent direction is chosen so that $\mathbf{h} \cdot \nabla f(\mathbf{p}) < 0$. Local minimum $\hat{\mathbf{p}}$ of function f is a stationary point.

If f has continuous second order derivative then at the local minimum the first order derivative is equal to zero $\nabla f(\hat{\mathbf{p}}) = 0$ and the second order derivative $\nabla^2 f(\hat{\mathbf{p}})$ is positive definite. Since function f is in a special form,

calculation of ∇f and $\nabla^2 f$ is simplified. To calculate both derivatives it is enough to have Jacobian of \mathbf{r} . The Jacobian matrix J of \mathbf{r} with respect to \mathbf{p} is defined as:

$$J(\mathbf{p}) = \left[\frac{\partial r_i(\mathbf{p})}{\partial x_j} \right]_{i=1, \dots, m, j=1, \dots, n} \quad (2.41)$$

Knowing the Jacobian J it is possible to calculate gradient ∇f and Hessian $\nabla^2 f$ as follows:

$$\nabla f(\mathbf{p}) = \sum_{i=1}^m r_i(\mathbf{p}) \cdot \nabla r_i(\mathbf{p}) = J^T(\mathbf{p}) \cdot \mathbf{r}(\mathbf{p}) \quad (2.42)$$

$$\begin{aligned} \nabla^2 f(\mathbf{p}) &= \sum_{i=1}^m \nabla r_j(\mathbf{p}) \cdot \nabla r_i(\mathbf{p}) + \sum_{i=1}^m r_i(\mathbf{p}) \cdot \nabla^2 r_i(\mathbf{p}) \\ &= J^T(\mathbf{p}) \cdot J(\mathbf{p}) + \sum_{i=1}^m r_i(\mathbf{p}) \cdot \nabla^2 r_i(\mathbf{p}) \end{aligned} \quad (2.43)$$

If point x is close to local minimum \hat{x} , then the term $\sum_{i=1}^m r_i(\mathbf{p}) \cdot \nabla^2 r_i(\mathbf{p})$ in (2.43) is small and Hessian can be approximated in the following way:

$$\nabla^2 f(\mathbf{p}) \approx J^T(\mathbf{p}) \cdot J(\mathbf{p}) \quad (2.44)$$

Gradient descent minimisation

Gradient descent is a simple and intuitive iterative minimisation method. The parameter update is performed according to the negative gradient direction:

$$\mathbf{p}^{t+1} = \mathbf{p}^t - \lambda \cdot \nabla f(\mathbf{p}) \quad (2.45)$$

where λ is an adaptable scaling parameter. λ is increased when update is successful and decreased in the case of failure which is described in Algorithm 2.3.1.

Algorithm 2.3.1: GRADIENT DESCENT MINIMISATION($f, \mathbf{p}_0, \epsilon$)

```

t ← 0, pt ← p0
while ‖∇f(pt)‖ > ε
do
  pt+1 ← pt - λ · ∇f(pt)
  if f(pt+1) < f(pt)
  then
    t ← t + 1
    λ ← 2 · λ
  else
    λ ← λ / 2

```

The problem is that the search step should be longer if gradient is small and shorter when gradient is high which is opposite to (2.45). Another issue is that curvature of f might be different in different directions so search step should be different in different directions. As a result gradient descent optimisation is relatively slow (linear convergence) but robust.

Adaptation of λ can be replaced by linear search. In linear search [29] vector $-\nabla f$ is treated as a search direction and in each step the algorithm searches 1-d function $f(\mathbf{p}^t - \lambda \cdot \nabla f(\mathbf{p}^t))$ of λ for minimum. However, optimisation of 1-d function is time consuming.

Gauss-Newton minimisation

Disadvantages of the gradient descent algorithm are partially improved in the Newton algorithm. The algorithm models f around \mathbf{p}_0 as a quadratic function according to the Taylor expansion:

$$f(\mathbf{p}) = f(\mathbf{p}_0) + (\mathbf{p} - \mathbf{p}_0)^T \cdot \nabla f(\mathbf{p}_0) + \frac{1}{2} \cdot (\mathbf{p} - \mathbf{p}_0)^T \cdot \nabla^2 f(\mathbf{p}_0) \cdot (\mathbf{p} - \mathbf{p}_0) + o(\|\mathbf{p} - \mathbf{p}_0\|^3) \quad (2.46)$$

After leaving out the last term, replacing \mathbf{p}_0 by \mathbf{p}^t and \mathbf{p} by \mathbf{p}^{t+1} the quadratic model is obtained:

$$f(\mathbf{p}^{t+1}) \approx q(\mathbf{p}^{t+1}) = f(\mathbf{p}^t) + (\mathbf{p}^{t+1} - \mathbf{p}^t)^T \cdot \nabla f(\mathbf{p}^t) + \frac{1}{2} \cdot (\mathbf{p}^{t+1} - \mathbf{p}^t)^T \cdot \nabla^2 f(\mathbf{p}^t) \cdot (\mathbf{p}^{t+1} - \mathbf{p}^t) \quad (2.47)$$

To minimise (2.47) it is set $\nabla q(\mathbf{p}^{t+1}) = 0$:

$$\nabla f(\mathbf{p}^t) + \nabla^2 f(\mathbf{p}^t) \cdot (\mathbf{p}^{t+1} - \mathbf{p}^t) = 0 \quad (2.48)$$

and solve (2.48) for \mathbf{p}^{t+1} to obtain the Newton updating rule:

$$\mathbf{p}^{t+1} = \mathbf{p}^t - (\nabla^2 f(\mathbf{p}^t))^{-1} \cdot \nabla f(\mathbf{p}^t) \quad (2.49)$$

After replacing (2.45) by (2.49) and setting $\lambda = 1$ in Algorithm 2.3.1 the Newton algorithm (Algorithm 2.3.2) is obtained.

Algorithm 2.3.2: NEWTON MINIMISATION($f, \mathbf{p}_0, \epsilon$)

```

 $t \leftarrow 0, \mathbf{p}^t \leftarrow \mathbf{p}_0$ 
 $\mathbf{p}^{t+1} = \mathbf{p}^t - (\nabla^2 f(\mathbf{p}^t))^{-1} \cdot \nabla f(\mathbf{p}^t)$ 
while  $f(\mathbf{p}_t) - f(\mathbf{p}_{t+1}) \geq \epsilon$ 
  do  $\{\mathbf{p}^{t+1} = \mathbf{p}^t - (\nabla^2 f(\mathbf{p}^t))^{-1} \cdot \nabla f(\mathbf{p}^t)$ 

```

The algorithm can be applied only when Hessian $H = \nabla^2 f(\mathbf{p}^t)$ has full rank. In least square optimisation Hessian can be approximated by $J^T(\mathbf{p}) \cdot J(\mathbf{p})$ according to (2.44). Such approximation saves significantly computation time but is possible only for \mathbf{p} which is close to local minimum $\hat{\mathbf{p}}$ because for such \mathbf{p} function f can be approximated by the quadratic model (2.47). Newton least square optimisation together with Hessian approximation (2.44) is called the Gauss-Newton algorithm. For \mathbf{p} close to $\hat{\mathbf{p}}$, the Gauss-Newton algorithm has quadratic convergence [29].

Levenberg-Marquardt algorithm

While the gradient descent algorithm is robust with linear convergence, the Gauss-Newton algorithm is faster (quadratic convergence) with less robustness. The Levenberg-Marquardt (LM) algorithm [29], [30] is a blend of the gradient descent and Gauss-Newton optimisation algorithms. It is a heuristic algorithm which works like gradient descent for points far from minimum and works like the Gauss-Newton algorithm for points close to minimum. The LM update step [29], [30] is a mixture of the gradient descent step and Gauss-Newton step which is controlled by *damping parameter* μ :

$$\mathbf{p}^{t+1} \leftarrow \mathbf{p}^t - (\nabla^2 f(\mathbf{p}^t) + \mu \cdot I)^{-1} \cdot \nabla f(\mathbf{p}^t) \quad (2.50)$$

According to (2.42) and (2.44) calculation of gradient $\nabla f(\mathbf{p}^t) = J^T(\mathbf{p}^t) \cdot r(\mathbf{p}^t)$ and Hessian $\nabla^2 f(\mathbf{p}^t) = J^T(\mathbf{p}^t) \cdot J(\mathbf{p}^t)$ is simplified. If $\mu = 0$, then LM update is equivalent to Gauss-Newton update (2.49). On the other hand, if μ is big, then LM update is close to gradient descent update (2.45). Rank deficiency of Hessian is compensated by the term $\mu \cdot I$.

μ is an adaptable parameter whose adaptation is controlled by factor $\rho = (f(\mathbf{p}^{t+1}) - f(\mathbf{p}^t)) / (q(\mathbf{p}^{t+1}) - q(\mathbf{p}^t))$. $\rho \approx 1$ shows high accordance of f with quadratic model q at \mathbf{p}^t . High value of ρ means that μ can be decreased to move the algorithm towards Gauss-Newton step. On the other hand, small value of ρ means that μ should be increased to move the algorithm towards gradient descent step.

Algorithm 2.3.3: LEVENBERG-MARQUARDT MINIMIZ. $(f, \mathbf{p}_0, \epsilon)$

```

 $t \leftarrow 0, \nu \leftarrow 2, \mathbf{p} \leftarrow \mathbf{p}_0$ 
 $A \leftarrow \nabla^2 f(\mathbf{p}^t), \mu \leftarrow \max\{a_{ii}\}$ 
while  $\|\nabla f(\mathbf{p}^t)\| > \epsilon$ 
   $\mathbf{p}^{t+1} \leftarrow \mathbf{p}^t - (\nabla^2 f(\mathbf{p}^t) + \mu \cdot I)^{-1} \cdot \nabla f(\mathbf{p}^t)$ 
   $\rho \leftarrow \frac{f(\mathbf{p}^{t+1}) - f(\mathbf{p}^t)}{q(\mathbf{p}^{t+1}) - q(\mathbf{p}^t)}$ 
  if  $\rho > 0$ 
    then  $\begin{cases} t \leftarrow t + 1 \\ \mu \leftarrow \mu \cdot \max\{\frac{1}{3}, 1 - (2 \cdot \rho - 1)^3\} \\ \nu \leftarrow 2 \end{cases}$ 
  else  $\begin{cases} \mu \leftarrow \mu \cdot \nu \\ \nu \leftarrow 2 \cdot \nu \end{cases}$ 

```

If the value of μ is high, then Hessian is not used and the LM algorithm is equivalent to gradient descent. Replacing I by $\text{diag}(H)$ make scaling proportional to the curvature of f . The following update proposed by Marquardt

$$\mathbf{p}^{t+1} \leftarrow \mathbf{p}^t - (\nabla^2 f(\mathbf{p}^t) + \mu \cdot \text{diag}(\nabla^2 f(\mathbf{p}^t)))^{-1} \cdot \nabla f(\mathbf{p}^t) \quad (2.51)$$

results in a larger step along the direction for which gradient is smaller.

Approximation of Jacobian

Objective function (2.38) is optimized with respect to vector of parameters \mathbf{p} :

$$f(\mathbf{p}) = \sum_{i=1}^m r_i^2(\mathbf{p}) = \sum_{(x,y) \in L} (u(x,y) - v_{T_P}(x,y))^2 \quad (2.52)$$

where:

$$v_{T_P}(x,y) = v(T_P(x,y)) = v(T_P^x(x_i, y_i), T_P^y(x_i, y_i)) = v(x', y') \quad (2.53)$$

Jacobian of $r(\mathbf{p})$ is defined as:

$$J(\mathbf{p}) = \left[\frac{\partial r_i(\mathbf{p})}{\partial p_j} \right]_{i=1, \dots, m, j=1, \dots, n} \quad (2.54)$$

where:

$$\begin{aligned} \frac{\partial r_i(\mathbf{p})}{\partial p_j} &= \frac{\partial (u(x_i, y_i) - v_{T_P}(x_i, y_i))}{\partial p_j} = - \frac{\partial v(T_P^x(x_i, y_i), T_P^y(x_i, y_i))}{\partial p_j} \\ &= - \left(\frac{\partial v(x', y')}{\partial x} \cdot \frac{\partial T_P^x(x_i, y_i)}{\partial p_j} + \frac{\partial v(x', y')}{\partial y} \cdot \frac{\partial T_P^y(x_i, y_i)}{\partial p_j} \right) \end{aligned} \quad (2.55)$$

Image derivatives at any point can be approximated by central difference:

$$\frac{\partial v(x, y)}{\partial x} = \frac{v(x + \Delta x, y) - v(x - \Delta x, y)}{2 \cdot \Delta x} \quad (2.56)$$

If the distance between neighbouring pixels is equal to 1, then (2.56) can be simplified to:

$$\frac{\partial v(x, y)}{\partial x} = \frac{1}{2}(v(x + 1, y) - v(x - 1, y)) \quad (2.57)$$

2.4 Affine 2-d registration

2.4.1 The algorithm implemented

Affine registration works according to the general algorithm described in Section 2.3.1 and was implemented based on the following components:

- affine transformation T_A (Section 2.2.3, Section 2.3.4)
- similarity measure D_{SSD} (Section 2.2.4)
- Levenberg-Marquardt minimisation (Section 2.3.7)
- multi resolution procedure (Section 2.3.2)

The algorithm implemented assumes that images u and v to be registered are single-modal so that D_{SSD} can be used as similarity measure. The algorithm minimises $D_{SSD}(u, v_{T_P})$ with respect to affine transformation $T_A = T_P$ which is parameterised by vector \mathbf{p} . The algorithm implementation is presented in terms of the 3 main components of registration, the essential parts are cost function and calculation of Jacobian which are input data to the Levenber-Marquardt algorithm (Section 2.3.7).

Transformation Affine transformation $T_P(x, y)$ is defined as follows:

$$T_P(x, y) = (I + A) \cdot \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} = (x', y')$$

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{bmatrix} \quad (2.58)$$

Identity matrix I was introduced to show that only displacement field is parameterised by a vector of 6 parameters:

$$\mathbf{p} = [p_1, p_2, \dots, p_6] = [a_{1,1}, a_{1,2}, t_x, a_{2,1}, a_{2,2}, t_y] \quad (2.59)$$

Using \mathbf{p} (2.58) can be rewritten as

$$\begin{aligned}
x' &= x + a_{1,1} \cdot x + a_{1,2} \cdot y + t_x = x + p_1 \cdot x + p_2 \cdot y + p_3 \\
y' &= y + a_{2,1} \cdot x + a_{2,2} \cdot y + t_y = y + p_4 \cdot x + p_5 \cdot y + p_6
\end{aligned} \tag{2.60}$$

Cost function The cost function is defined as follows:

$$\begin{aligned}
f(\mathbf{p}) &= \sum_{(x,y) \in L} (u(x,y) - v_{T_P}(x,y))^2 \\
&= \sum_{(x,y) \in L} (u(x,y) - v(x',y'))^2 \\
&= \sum_{(x,y) \in L} r^2(\mathbf{p}, x, y)
\end{aligned} \tag{2.61}$$

where:

$$\begin{aligned}
r(\mathbf{p}, x, y) &= u(x, y) - v(x', y') \\
&= u(x, y) - v(x + p_1 \cdot x + p_2 \cdot y + p_3, y + p_4 \cdot x + p_5 \cdot y + p_6)
\end{aligned}$$

Since pixels (x, y) are at grid positions, they can be sorted and enumerated, thereby enabling summation over $(x, y) \in L$ in (2.61) to be replaced by summation over integer index $i = 1, 2, \dots, m$ where $m = |L|$ is number of pixels.

$$f(\mathbf{p}) = \sum_{(x,y) \in L} r^2(\mathbf{p}, x, y) = \sum_{i=1}^m r^2(\mathbf{p}, x_i, y_i) = \sum_{i=1}^m r_i^2(\mathbf{p}) = \|\mathbf{r}(\mathbf{p})\|^2 \tag{2.62}$$

Jacobian Jacobian of $\mathbf{r}(\mathbf{p})$ is defined as:

$$J(\mathbf{p}) = \left[\frac{\partial r_i(\mathbf{p})}{\partial p_j} \right]_{i=1, \dots, m, j=1, \dots, 6} \tag{2.63}$$

The i -th row of Jacobian can be calculated according to:

$$\begin{aligned}
(\nabla_p r_i(\mathbf{p}))^T &= (\nabla_p [u(x_i, y_i) - v(x'_i, y'_i)])^T = \\
&= -(\nabla_p v(x_i + p_1 \cdot x_i + p_2 \cdot y_i + p_3, y_i + p_4 \cdot x_i + p_5 \cdot y + p_6))^T \\
&= -[v_x \cdot x_i, v_x \cdot y_i, v_x, v_y \cdot x_i, v_y \cdot y_i, v_y] \tag{2.64}
\end{aligned}$$

where:

$$\begin{aligned} v_x &= v_x(x'_i, y'_i) = \frac{\partial v(x'_i, y'_i)}{\partial x} \\ v_y &= v_y(x'_i, y'_i) = \frac{\partial v(x'_i, y'_i)}{\partial y} \end{aligned} \quad (2.65)$$

and ∇_p stands for gradient with respect to \mathbf{p} .

Jacobian $J(\mathbf{p})$ and vector $\mathbf{r}(\mathbf{p})$ are necessary data for the Levenberg-Marquardt algorithm.

2.4.2 Experimental results

Figure 2.21 shows examples of 2-d affine registration. Slices of RTPCT and CBCT of size 226x282 pixels were chosen as test images. In the beginning both slices were manually registered. Then the CBCT image was deformed by affine deformation with randomly chosen parameters. Deformed CBCT images were used as reference images and RTPCT images were used as source images. The maximal value of the displacement field of simulated deformation was 46 pixels. After registration average discrepancy between simulated and registered displacement fields was below 1.25 pixel. Difference between registered and reference images and their contours was also displayed to demonstrate the quality of registration. To obtain contours, Canny edge detector was used.

Figure 2.22 and Figure 2.23 show results of similar registration with intensity distortion added (global non-linear intensity mapping in Figure 2.22, and smooth additive distortion in Figure 2.23). In both cases simulated spatial deformation was the same as the deformation shown in Figure 2.21b. The intensity distortion is shown in Figure 2.22b and Figure 2.23b. In the case of non-linear mapping, the quality of registration was found to deteriorate; and in the case of additive distortion, registration was found to fail. Figure 2.24 illustrates discrepancy between two displacement fields: correct registration (shown in Figure 2.21) and failed registration (shown in Figure 2.23).

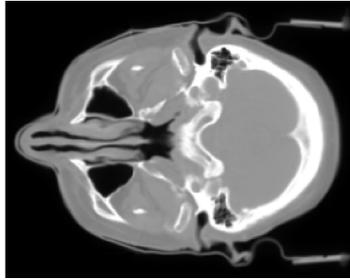
Figure 2.25 shows the graph of the maximum value in the simulated displacement field versus the maximum value in the displacement field after registration. The displacement field was simulated by random choice of affine parameters with uniform distribution. In the beginning the displacement field was small and was gradually increased by multiplying the same affine parameters by a gradually increasing value. The plot of difference between the simulated and registered displacement fields versus the maximum value of the simulated displacement field is shown in Figure 2.25. The displacement field after registration is small and almost constant until simulated deformation reaches the displacement limit of the image (Section 2.3.2). When initial deformation exceeds the displacement limit, the

value of the difference significantly increases showing failed affine registration. The displacement limit of the image is much bigger than those typical encountered in medical practice. The biggest deformation which can still be recovered based on the graph shown in Figure 2.25 is shown in Figure 2.26.

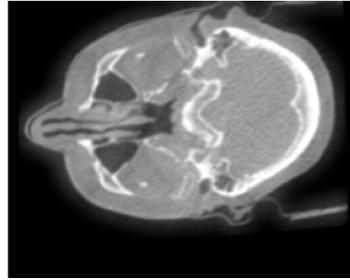
Figure 2.27 and Figure 2.28 show examples of 3-d affine registration which are similar to 2-d registration shown in Figure 2.21. RTPCT and CBCT volumes were used as source images. 3-d volumes were represented as isosurface generated by suitable threshold selection. Cross section slice was used to visualise the results as an alternative representation to the isosurface. One slice shows edges of images and the other the difference between images. Average discrepancy between simulated and registered displacement fields was below 3 pixels.

2.4.3 Concluding remarks

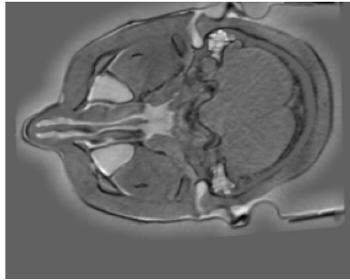
- Sum of squared differences SSD is a suitable cost function for affine registration. Minimisation of SSD by Levenberg-Marquardt optimisation gives an efficient registration algorithm. It is able to recover much bigger deformations than those encountered in medical practice. Since affine registration is fast and robust, it is frequently applied as a normalisation step before deformable registration.
- SSD based affine registration is sensitive to intensity distortion. Intensity distortions simulated in experiments were similar to those encountered in medical practice. The algorithm often fails when intensity distortion is introduced so SSD based affine registration should be supplemented by intensity correction.
- The algorithm can be easily generalised to register 3-d volumes.



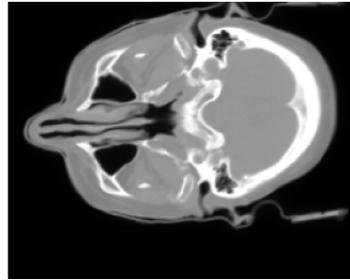
(a) source image



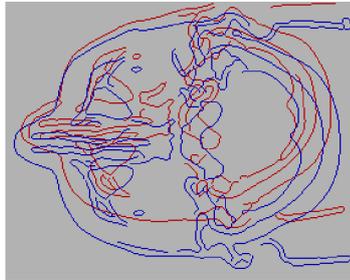
(b) reference image



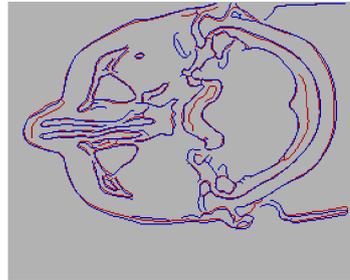
(c) difference between reference and registered



(d) registered image



(e) contours before registration



(f) contours after registration

Figure 2.21: 2-d affine registration of RTPCT and CBCT slices of size 226x282 pixels. Reference image is an affine transformed version of CBCT slice. The deformation was known before registration and maximum value of displacement field of this deformation was 46 pixels. Registration reduced maximum discrepancy between simulated and registered displacement fields to 1.25 pixels.

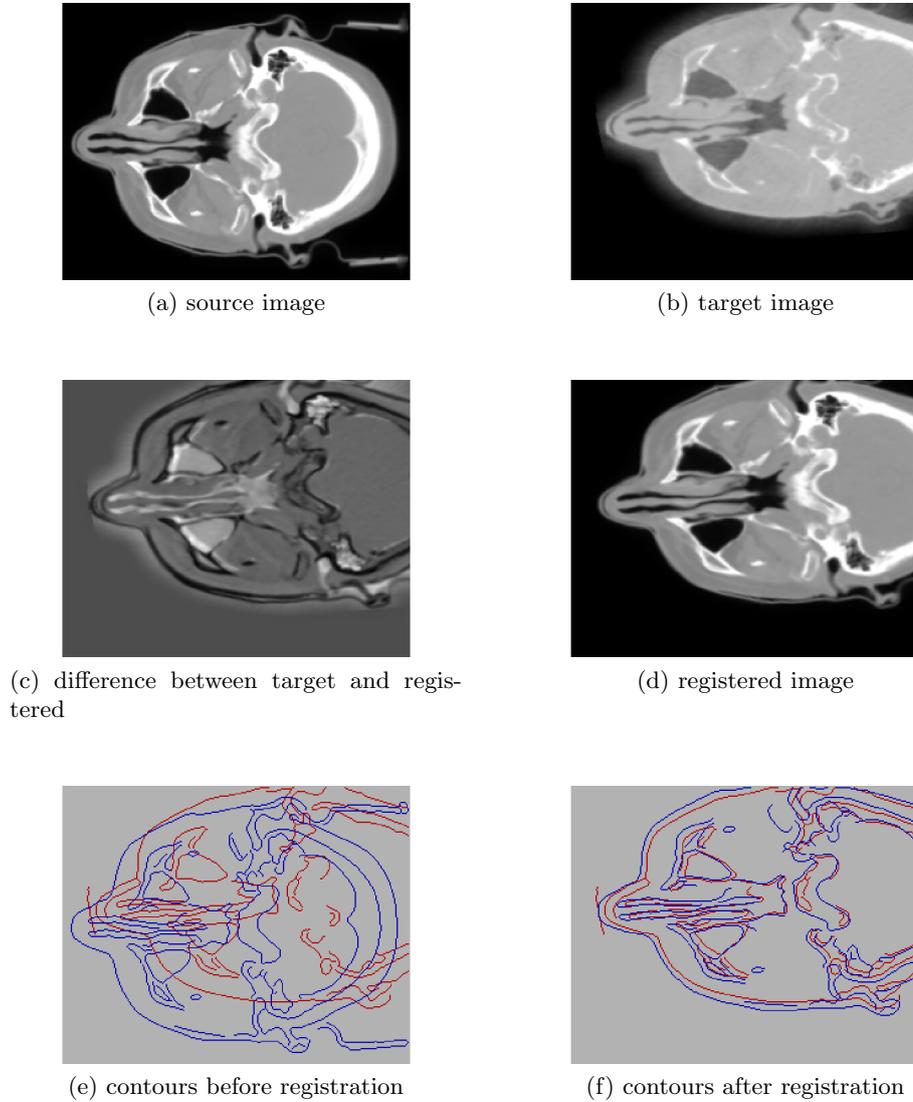


Figure 2.22: 2-d affine registration of RTPCT and CBCT slices of size 226x282 pixels with non-linear intensity distortion. Reference image is an affine transformed version of CBCT slice. Spatial deformation was known before registration and it was the same as in Figure 2.21. Maximum value of displacement field of this deformation was 46 pixels. Before spatial transformation non-linear intensity distortion was applied to reference image. It was low quality registration because maximum discrepancy between simulated and registered displacement fields was 9 pixels.

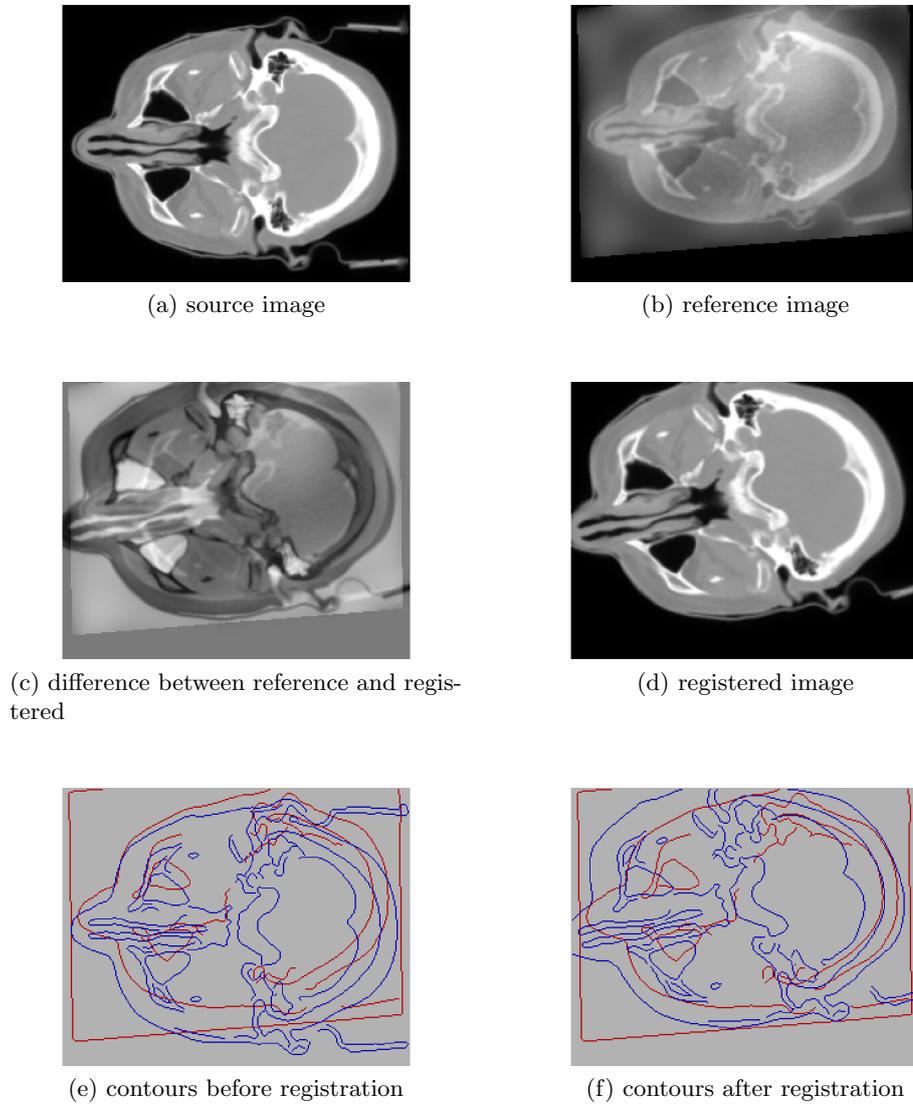
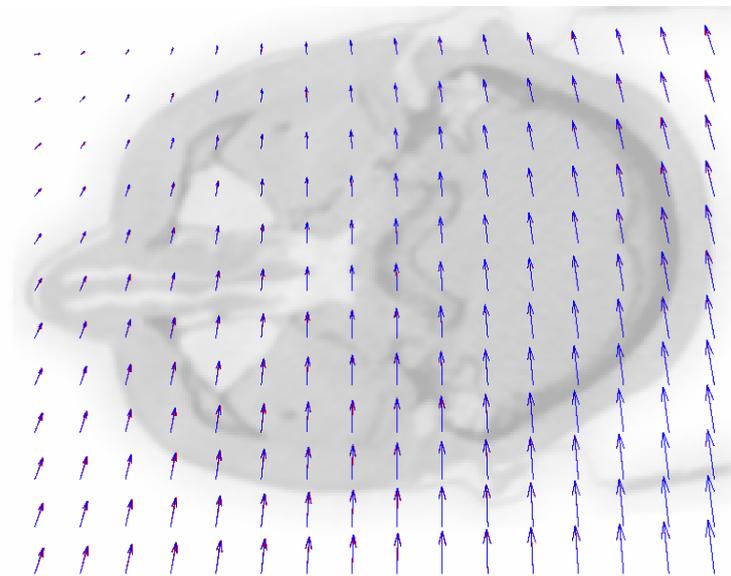
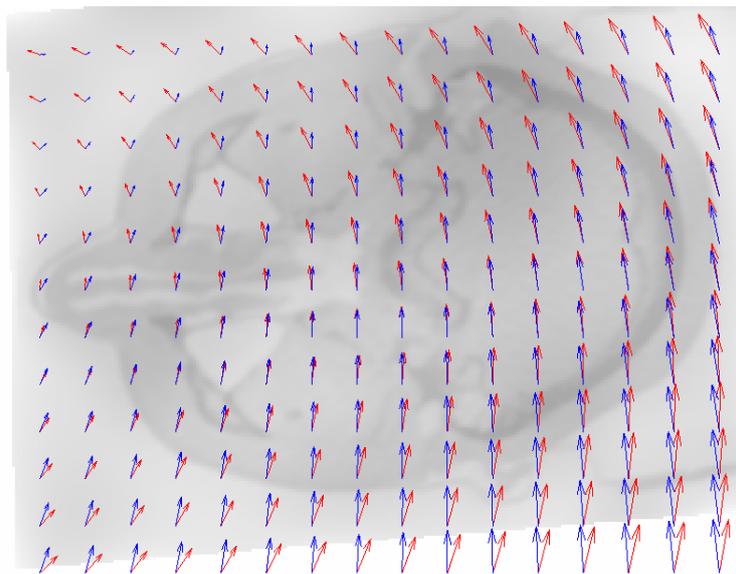


Figure 2.23: 2-d affine registration of RTPCT and CBCT slices of size 226x282 pixels with smooth intensity distortion. Reference image is an affine transformed version of CBCT slice. Spatial deformation was known before registration and it was the same as in Figure 2.21. Maximum value of displacement field of this deformation was 46 pixels. Before spatial transformation smooth intensity distortion was added to reference image. Registration failed because maximum discrepancy between simulated and registered displacement fields was 58 pixels.



(a) registration without intensity distortion



(b) registration with intensity distortion

Figure 2.24: Displacement fields after registration of images in Figure 2.21 and Figure 2.23. Blue colour - simulated displacement field, red colour - estimated displacement field. In case of registration without intensity distortion both displacement fields are equal (they overlap) as shown in (a). In case of registration with intensity distortion displacement fields are different as shown in (b).

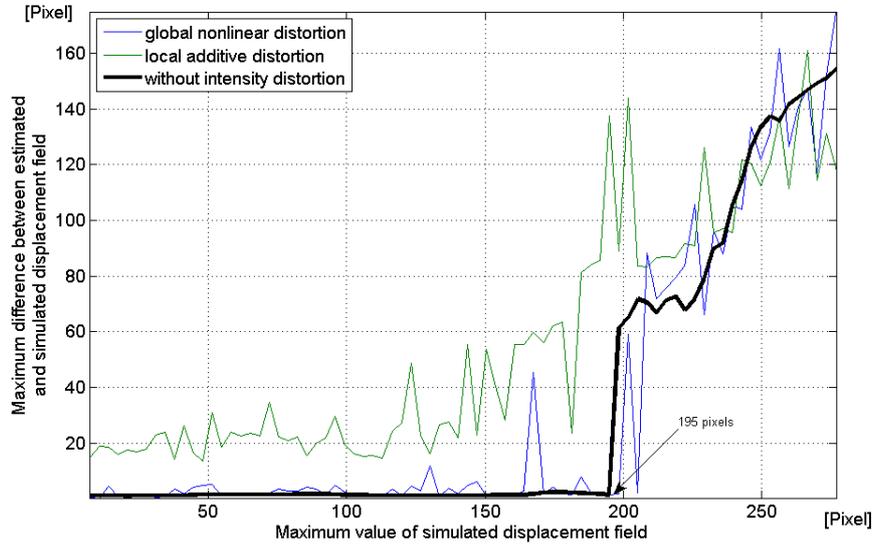
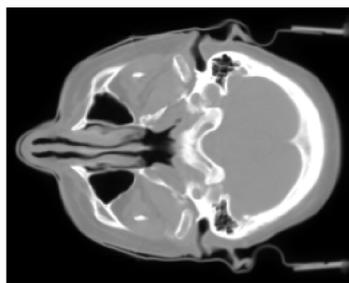
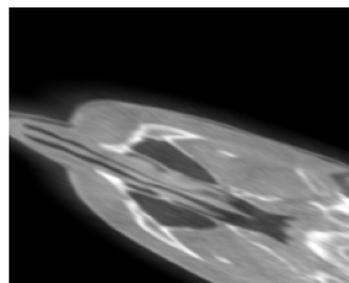


Figure 2.25: Maximum difference between simulated and estimated displacement fields versus maximum value of simulated displacement field. Affine coefficients were randomly generated and then multiplied by a gradually increasing factor. When maximum value of displacement field of simulated deformation exceeds 195 pixels registration fails.

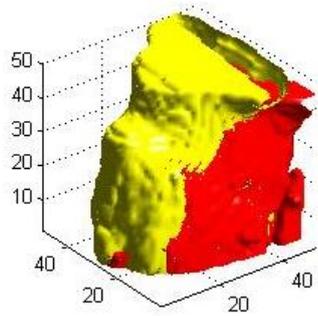


(a) source image

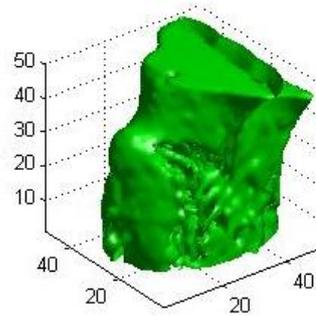


(b) maximal deformation that was still properly registered

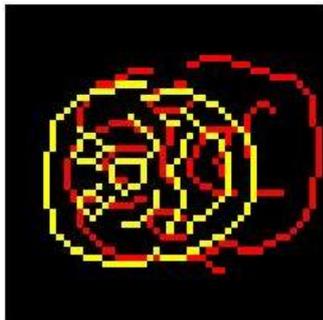
Figure 2.26: Illustration of maximally deformed image for which deformation can still be recovered. It is a demonstration of displacement limit of image shown in Figure 2.25. (a) source image, (b) maximal deformation of reference image which was properly registered.



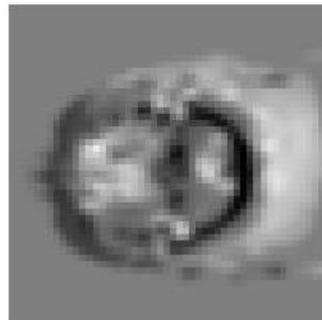
(a) source and reference (isosurface)



(b) source - reference (isosurface)

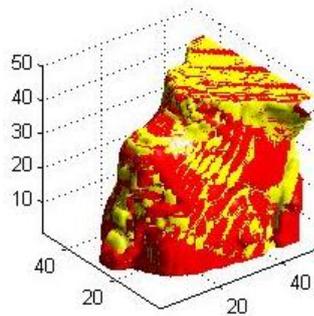


(c) source and reference (slice - edges)

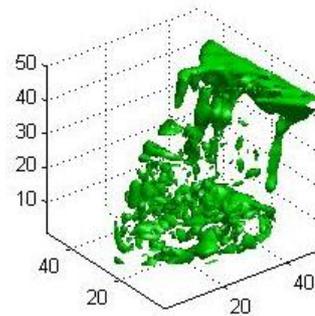


(d) source - reference (slice)

Figure 2.27: RTPCT and CBCT volumes before registration. Reference image is an affine transformed version of CBCT volume. (a) yellow - source (RTPCT) and red - reference (CBCT) images displayed as isosurface, (b) - difference between images displayed as isosurface. (c) - one slice of volume displayed after Canny edge detector, (d) - one slice of volume displayed as difference between images.



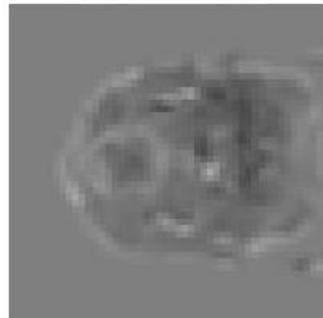
(a) registered and reference (isosurface)



(b) registered - reference (isosurface)



(c) registered and reference (edges)



(d) registered - reference (slice)

Figure 2.28: 3-d affine registration of volumes shown in Figure 2.27. Reference image is affine transformed version of CBCT volume. (a) yellow - registered (RTPCT) and red - reference (CBCT) images displayed as isosurface, (b) - difference between images displayed as isosurface. (c) - one slice of volume displayed after Canny edge detector, (d) - one slice of volume displayed as difference between images.

2.5 Nonrigid B-Spline 1-d registration

2.5.1 The algorithm implemented

In this section all data for registration are restricted to artificial 1-d images, represented as a sequence of numbers. It is called image to be consistent with 2-d registration but they can also be called signals or curves. Consideration is restricted to smooth curves with limited high frequency components.

During registration the displacement field is estimated. The displacement field is also a smooth curve. It is assumed that the highest frequency component of the displacement field is significantly lower than that in the images to be registered.

Algorithm for B-Spline deformable registration is similar to affine registration (Section 2.4.1). The difference is that affine transformation is replaced by B-Spline. The algorithm works like the general algorithm described in Section 2.3.1 and consists of the following components:

- B-Spline transformation T_B (Section 2.2.3, Section 2.3.5)
- similarity function D_{SSD} (Section 2.2.4)
- Levenberg-Marquardt minimisation (Section 2.3.7)
- multi resolution procedure (Section 2.3.2)

Reference image u and source image v are single-modal so D_{SSD} is used. Transformation $T_P = Id + T_B$ is parameterised by B-Spline control points that form parameter vector \mathbf{p} . Calculation of cost function and Jacobian is described below. These are necessary data for the Levenber-Marquardt algorithm (Section 2.3.7).

Transformation 1-d B-Spline transformation is defined as follows

$$T_P(x) = x + T_B(x) = x + \sum_{i=1}^n b^l \left(\frac{x}{h} - i \right) \cdot p_i \quad (2.66)$$

Here n is number of control points and l is degree of the basis function (in our case 3). Knots are distributed uniformly. The term x is used because only the displacement field is parameterised (Section 2.2.3). Equation (2.66) can be rewritten in inner product form:

$$T_B(x) = x + \mathbf{b}^T(x) \cdot \mathbf{p} = x' \quad (2.67)$$

where $\mathbf{b}(\mathbf{x})$ is the vector of B-Spline basis functions and \mathbf{p} the vector of control points (2.15).

Cost function The definition of cost function is as follows:

$$\begin{aligned}
f(\mathbf{p}) &= \sum_{x \in L} (u(x) - v_{T_P}(x))^2 \\
&= \sum_{x \in L} \left(u(x) - v(x + \mathbf{b}^T(x) \cdot \mathbf{p}) \right)^2 \\
&= \sum_{x \in L} r^2(\mathbf{p}, x)
\end{aligned}$$

Replacing summation over x by summation over integer index $i = 1, 2, \dots, m$, where $m = |L|$ is number of points, gives

$$f(\mathbf{p}) = \sum_{x \in L} r^2(\mathbf{p}, x) = \sum_{i=1}^m r^2(\mathbf{p}, x_i) = \sum_{i=1}^m r_i^2(\mathbf{p}) = \|\mathbf{r}(\mathbf{p})\|^2 \quad (2.68)$$

Jacobian Jacobian of $\mathbf{r}(\mathbf{p})$ is defined as:

$$J(\mathbf{p}) = \left[\frac{\partial r_i(\mathbf{p})}{\partial p_j} \right]_{i=1, \dots, m, j=1, \dots, n} \quad (2.69)$$

The i -th row of Jacobian can be calculated according to:

$$\begin{aligned}
(\nabla_p r_i(\mathbf{p}))^T &= (\nabla_p [u(x_i) - v(x_i + \mathbf{b}^T(x_i) \cdot \mathbf{p})])^T \\
&= -\frac{\partial v(x_i)}{\partial x} \cdot \mathbf{b}^T(x_i)
\end{aligned} \quad (2.70)$$

where ∇_p stands for gradient with respect to \mathbf{p} .

Jacobian $J(\mathbf{p})$ and vector $\mathbf{r}(\mathbf{p})$ are necessary data for the Levenberg-Marquardt algorithm.

2.5.2 Experimental results

The registration algorithm was evaluated using simulated 1-d image $v(x)$. B-Spline curves with uniformly distributed knots were used in experiments to represent 1-d images (Section 2.3.5). Values of the control points were randomly generated on uniformly distributed set of knots. Consideration was restricted to smooth images whose regularity was controlled by a number of control points.

The control points can be considered as discrete samples of the image and B-Spline curve its approximation. Since knots are distributed uniformly, the distance between knots is a sampling period. According to Nyquist-Shannon theorem sampling period determines frequency bandwidth of the image. As

a result maximum frequency component of the image can be adjusted by changing distance between subsequent knots.

For constant image estimation of any displacement field is impossible because spatial deformation do not change the image. Texture allow to estimate simulated spatial deformation. Therefore to make registration feasible heuristic rule was applied. Registered images were expected to have significantly more variety than the displacement field. Therefore frequency components of registered images and displacement fields were related. The maximum frequency component of the displacement field was kept twice bigger than the maximum frequency component of the image. Apart from this the maximum value of the displacement field was kept smaller than half period of the maximum frequency component of the image (Figure 2.3). The rules are given in terms of knot spacing and are formalized as follows:

- Knots were uniformly distributed.
- The image had twice as many knots than the displacement field.
- The maximum value of the displacement field was less than the distance between knots in image (Figure 2.3)
- The displacement field had twice as many knots than additive intensity distortion.

The distortions were applied to generate deformed images as follows:

- smooth additive $u_a(x) = v(x) + A(x)$, $u(x) = u_a(T_u(x))$
- non-linear mapping $u(x) = F(v(T_u(x)))$.

For experimental purposes the displacement field $T_u(x)$ was randomly simulated like images. It was used to generate reference image $u(x) = v(T_u(x))$. Our goal was to recover simulated displacement field $T_u(x)$ by image registration. It was done by estimating deformation $T_v(x)$ so that $v(T_u(x)) \approx v(T_v(x))$. It is expected that estimated deformation should be close to the simulated one. In Figure 2.29 an example of succesful deformable registration is presented.

Intensity distortions $A(\cdot)$ and $F(\cdot)$ were applied to the reference image in another experiment. Homogeneous nonlinear intensity distortion $F(\cdot)$ was applied according to equation $u(x) = F(v(T_u(x)))$. Inhomogeneous additive intensity distortion $A(\cdot)$ was applied according to equation $u(x) = v(T_u(x)) + A(x)$. Both distortions $A(\cdot)$ and $F(\cdot)$ were B-Spline curves with randomly generated control points. Knots were distributed uniformly. Density of knots of additive distortion $A(\cdot)$ was twice smaller than density of knots of spatial deformation. More detailed description of both deformations is postponed to Section 4.3.

Figure 2.30 and Figure 2.31 show registration with intensity distortion. It shows that intensity distortion can prevent the algorithm from achieving correct registration.

Figure 2.32 presents how deformation influences accuracy of registration. If amplitudes of deformation do not exceed the displacement limit of the image (Section 2.3.2), then registration is correct. If deformation exceed the displacement limit then registration fails. The value of the displacement limit depends on image content.

All experiments so far shows registration results when simulated and registered models of the displacement fields agree. The goal of next experiment was to demonstrate how sensitive the registration algorithm is to deformation model. Therefore number of knots used for estimation deformation was altered. Additionally knot positions used to simulate deformation were randomly deviated from uniform positions. The deviation did not change knots order. Figure 2.33 shows how accuracy of registration depends on number of knots in the registration model. There is a local minimum when the number of knots in the registration model is 32 which is equal to the number of knots in the simulation model. However, the minimum is narrow and normally when the best fitting number of knots is unknown no one can rely on guess. However, the graph shows that choosing big enough number of knots can give acceptable results.

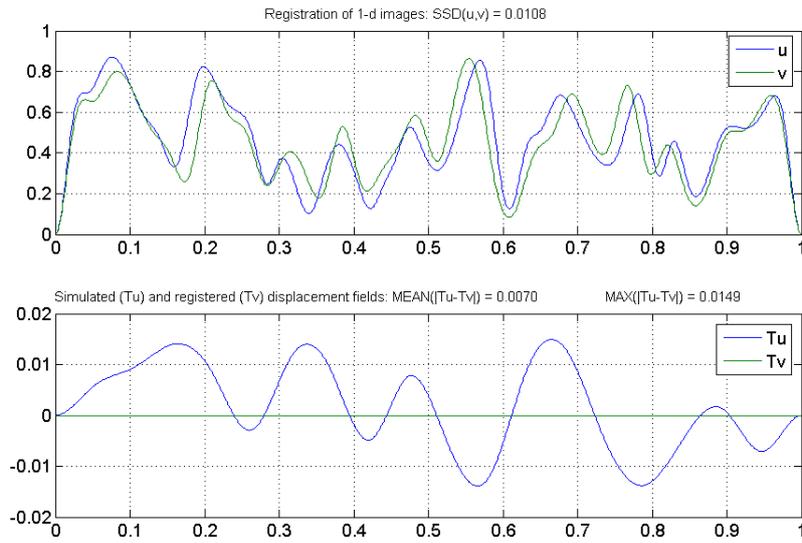
The results presented in Figure 2.32 and Figure 2.33 depend on image contents. The reason is that random deformation and random image used for experiments are not correlated. Deformation can not be recovered with high accuracy if the deformed image does not vary enough in positions of high deformation.

2.5.3 Concluding remarks

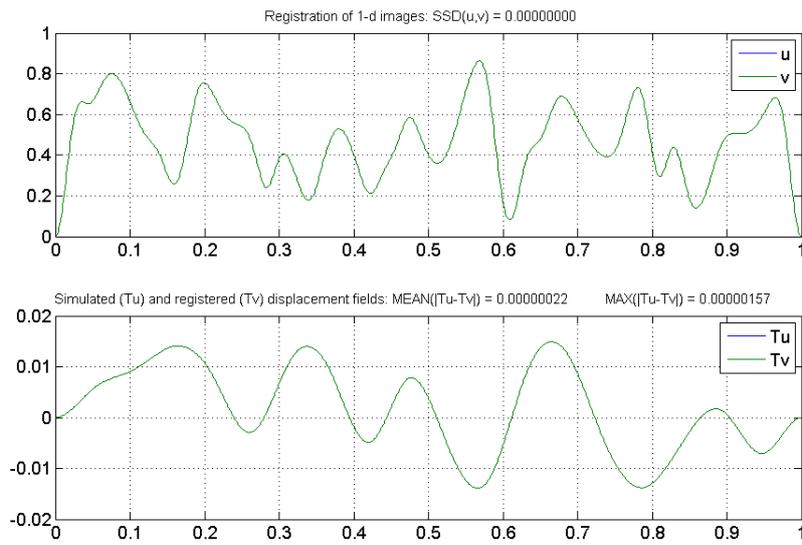
- Sum of squared differences SSD is a suitable cost function for B-Spline deformable registration. Minimisation of SSD by applying the Levenberg-Marquardt optimisation gives an efficient registration algorithm.
- If image and displacement field fulfil the following conditions:
 - the maximum frequency component of the image is twice bigger than the maximum frequency component of the displacement field
 - the maximum value of the displacement field is smaller than half period of the maximum frequency component of the image (Figure 2.3)

then simulated deformation can be recovered.

- If the optimum number of knots for best fitting is unknown, good results can be achieved by choosing large number of knots.
- In presence of intensity distortion the registration algorithm can fail.

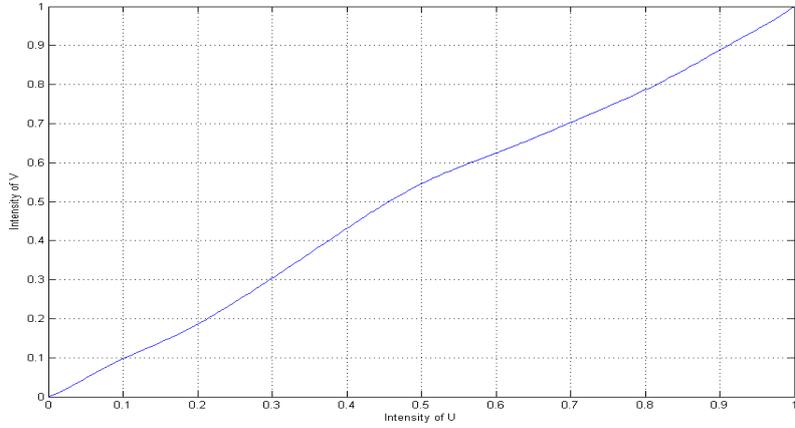


(a) before registration

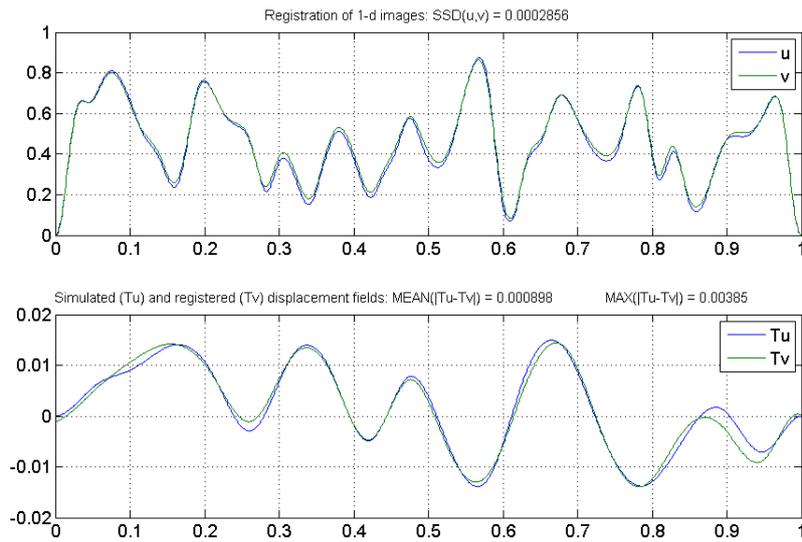


(b) after registration

Figure 2.29: 1-d deformable registration. Source image v is registered to reference image u . Reference image is a transformed version of source image $u(x) = v(T_u(x))$. The deformation T_u is known before registration. T_v is a displacement field estimated by registration algorithm. In the beginning T_v is initialised to constant function. After registration estimated displacement field is approximately equal to simulated $T_u \approx T_v$.

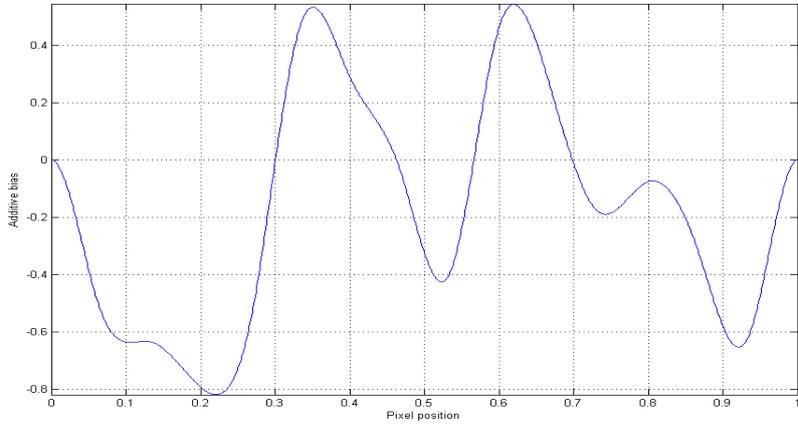


(a) non-linear intensity distortion

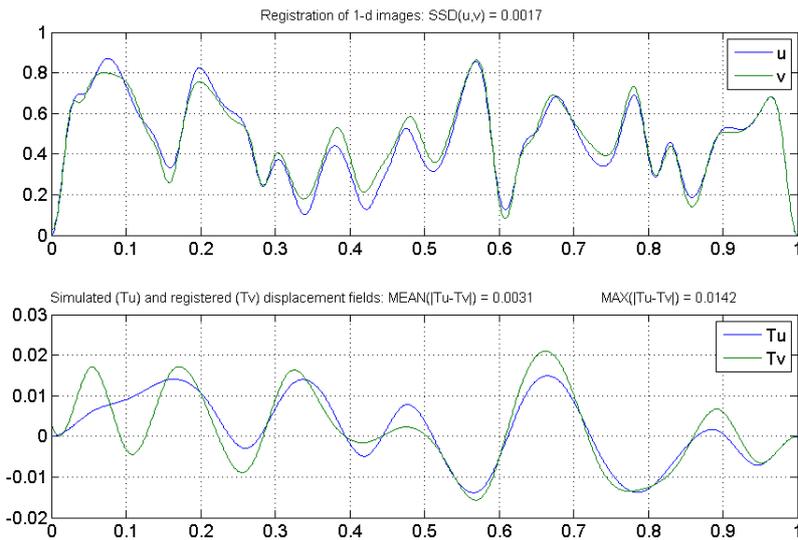


(b) registered with non-linear distortion

Figure 2.30: Deformable registration with non-linear intensity distortion. The experiment was similar to that from Figure 2.29 with simulated non-linear intensity distortion introduced. As a result reference image is equal to $u(x) = F(v(T_u(x)))$ where F is intensity distortion and T_u is spatial deformation. T_v is displacement field estimated during registration. This displacement field was initialised to constant function in the beginning. After registration $T_u \neq T_v$ because of intensity distortion F .

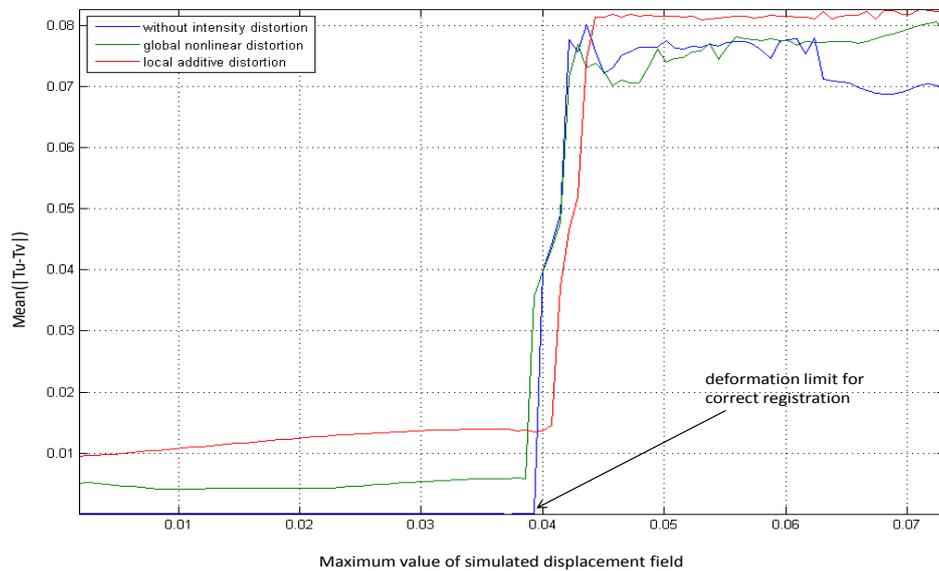


(a) additive bias

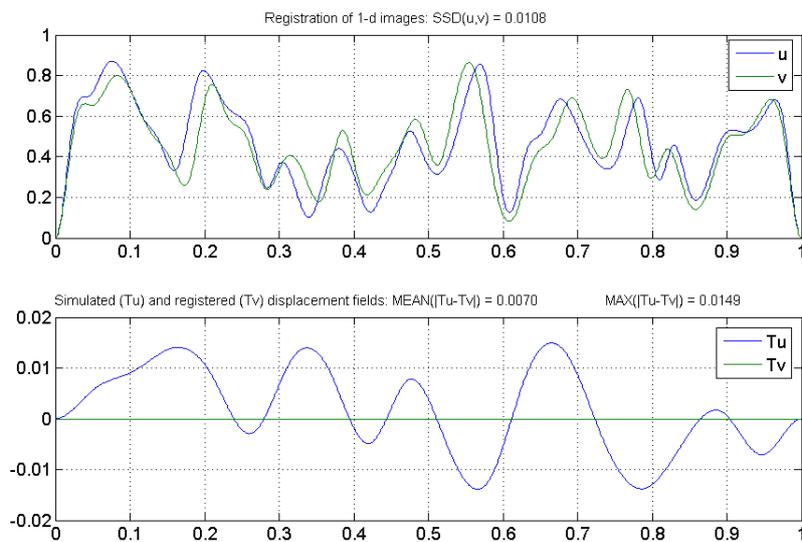


(b) registered with bias

Figure 2.31: Deformable registration with additive intensity distortion. The experiment was similar to that shown in Figure 2.29 with additive intensity distortion introduced. As a result reference image u is composition of additive intensity distortion $u_a(x) = v(x) + A(x)$ and spatial deformation $u(x) = u_a(T_u(x))$. T_v is a displacement field estimated during registration. This displacement field was in the beginning initialised to constant function. After registration $T_u \neq T_v$ because of intensity distortion A .



(a) registered vs. initial deformation



(b) deformation which still can be registered

Figure 2.32: Mean difference between simulated and estimated displacement fields versus maximum value of simulated displacement field. Values of control points of deformation were randomly generated and then multiplied by a gradually increasing value. When simulated deformation exceeds limit value 0.04 registration fails.

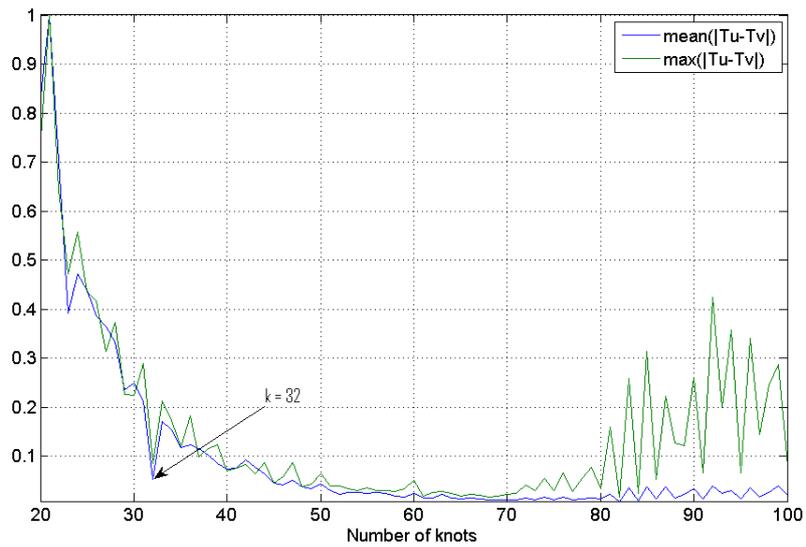


Figure 2.33: Mean and maximum discrepancy between simulated T_u and estimated T_v transformation versus number of knots used for registration. Both graphs were normalised to range $[0, 1]$ to show regularity rather than absolute values. Knots for registration were randomly deviated from uniform positions so that they do not match knots used for simulation. Number of knots in simulated transformation was equal to 32 and for this value graph has narrow minimum. Choosing number of knots from the range 50-70 gives good registration results.

2.6 Summary

- Sum of squared differences SSD is a suitable cost function for parametric affine and B-Spline image registration. Minimisation of SSD by Levenberg-Marquardt optimisation gives an efficient registration algorithm.
- If the displacement field model used to simulate the deformation is similar to the model applied to estimated the displacement field then SSD based registration is able to recover the simulated displacement field.
- In case of affine registration the algorithm is able to recover much bigger deformations than those encountered in medical practice. Since affine registration is fast and robust, it can be applied as a preprocessing step before deformable registration.
- In case of B-Spline registration to make displacement estimation feasible the frequency component of the image and the displacement field and value of displacement field should be bounded and fulfil specific conditions (Section 2.5.2).
- In presence of intensity distortion SSD based image registration often fails so the registration algorithm should be supplemented by intensity correction.

Chapter 3

Investigation of intensity correction

Usually images are captured at different times or by different devices which introduce intensity distortions. Variation of intensity in two images representing the same object can have different sources. Imaging devices or capture condition can introduce global intensity distortion. Some medical imaging devices introduce intensity inhomogeneity, whereby certain objects or tissues can appear with different intensities.

3.1 Fitting non-linear function

It is possible to assume that intensity distortion is coordinate independent mapping. For CBCT images it is a reasonable assumption. Given RTPCT and CBCT images of the same object, it is known that RTPCT has higher resolution, is less noisy and has more consistent intensity mapping. As a result RTPCT comparing to CBCT images can be considered as intensity distortion free. Figure 3.1 shows an example of estimation of linear mapping between RTPCT and CBCT images. The Figure 3.1c,d shows image of differences between intensities of corresponding pixels of RTPCT and CBCT images before and after correction. The correction has made image of differences more uniform. In other words image in Figure 3.1d has smaller contrast than this in Figure 3.1c.

Fitting non-linear function assumes that intensity distortion is a global (coordinate independent - homogeneous) non-linear function. To estimate this function pairs of corresponding pixels can be taken from both images to form an input-output sequence for approximation [36]. Approximation is consistent if intensity dependence of most of the image is independent of coordinates.

Intensities of images u and v might be considered as instances of random variables U and V . Joint intensity histogram $h(u, v)$ is used to calculate

probability distributions (Section 2.3.6). Having probabilities $p(u)$, $p(v)$ and $p(u, v)$ conditional probability is calculated as follows

$$p(v|u) = \frac{p(u, v)}{p(u)} \quad (3.1)$$

then conditional expectancy

$$f(u) = E(V|U = u) = \int v \cdot p(v|u)dv \quad (3.2)$$

and variance

$$Var(V|U = u) = \int (v - f(u))^2 \cdot p(v|u)dv \quad (3.3)$$

Conditional expectancy $f(u) = E(V|U = u)$ is a good estimator of correction mapping [11], [36]. Variance $Var(V|U)$ measures predictability of the mapping. Figure 3.2 shows intensity dependence of images shown in Figure 3.1. Intensity mapping $f(u)$ and variance $Var(V|U)$ are overlapped on $h(u, v)$. If relation between intensity of images can be modelled as homogeneous function $f(u)$, then $h(u, v)$ is distributed along the plot of this function. In Figure 3.1 joint intensity histogram $h(u, v)$ (black points in the background) is scattered along the plot of function $f(u)$ (blue curve) with variance $Var(V|U = u)$ (red curve). Variance demonstrates level of predictability of the mapping.

Figure 3.3 shows variance overlapped on the image. The bigger is the variance, the broader are scattered joint intensity histogram points around the curver representing the intensity mapping. The variance was calculated according to equation (3.3) for intensity values of each pixel. The variance values are overlapped as green component on image u . A significant part of variance comes from image edges. The edges are slightly misaligned which make $f(u)$ ambiguous. Such points for which mapping is unpredictable are outliers. The outliers may come from the image noise, image misalignment or components which exist only on one image. Figure 3.3 shows that small misalignment of images introduces small fraction of outliers. If misalignment is small then taking pairs of corresponding pixels to estimate intensity mapping can give good results. However even small number of outliers can reduce accuracy of approximation. The authors of [38] applied robust regression to solve this problem. They used a parametric polynomial representation of $f(u|\theta)$ where θ is a vector of coefficients:

$$f(u, \theta) = \theta_0 + \theta_1 \cdot u + \theta_2 \cdot u^2 + \dots + \theta_k \cdot u^k \quad (3.4)$$

Figure 3.2 suggests some probabilistic models for mapping. With Gaussian noise η introduced to model intensity mapping, intensity mapping is given by

$$g(u, \theta, \sigma) = f(u, \theta) + \eta = N(f(u, \theta), \sigma) \quad (3.5)$$

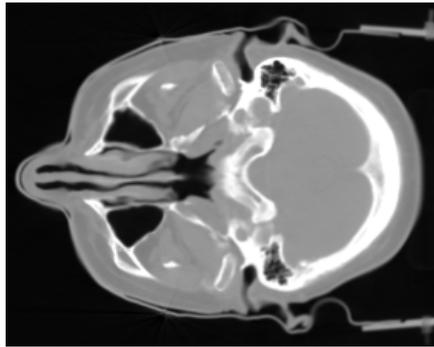
where $N(\cdot, \cdot)$ is Gaussian distribution, and θ, σ are mapping parameters. To estimate θ and σ mean square error is minimised:

$$\epsilon(\theta, \sigma) = \sum_{x,y} (g(u(x, y), \theta, \sigma) - v(x, y))^2 = \sum_{x,y} r^2(x, y) \quad (3.6)$$

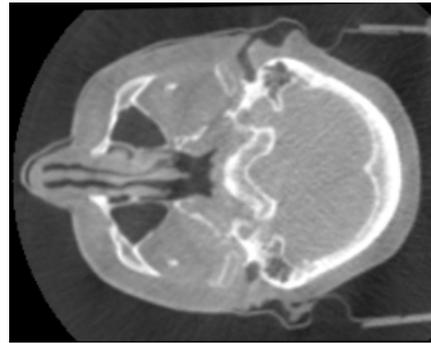
$\hat{\theta}$ and $\hat{\sigma}$ can be obtained by minimising $\epsilon(\theta, \sigma)$

$$(\hat{\theta}, \hat{\sigma}) = \arg \min_{\theta, \sigma} \epsilon(\theta, \sigma) \quad (3.7)$$

To deal with outliers the authors of [38] applied least trimmed squares (LTS) and binary reweighted least squares (RLS) estimation [39]. Pixels taken to approximation were sorted, and certain fraction of pixels with highest values of r was rejected in the next iteration. Variance was calculated from remaining pixels but taking into account correction on bias [38], [39]. They obtained iterative polynomial regression which was stable and repeatable for single-modal images.



(a) u (RTPCT)



(b) v (CBCT)



(c) $u - v$



(d) $f(u) - v$

Figure 3.1: (a) and (b) show two aligned medical images u and v whose intensity range is $[0, 1]$. Difference between images u and v is shown in (c), and difference between images $f(u)$ and v is shown in (d). f is linear regression function taking corresponding pixel values of u and v as input data. Difference images are displayed in the range of $[-0.5, 0.5]$. After linear correction of u the difference image is more uniform than the difference image without correction.

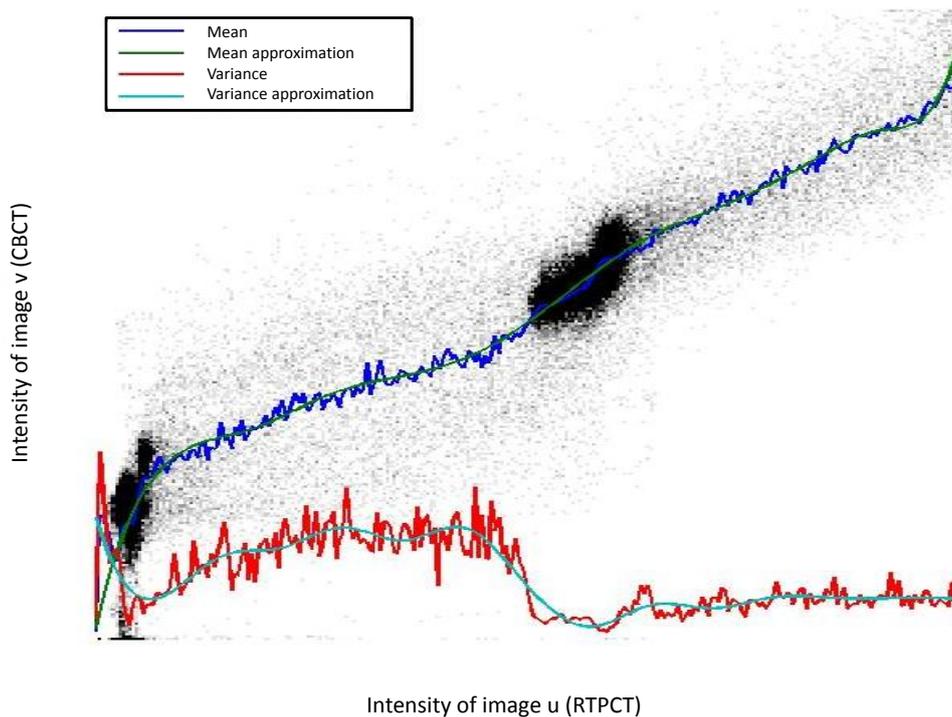


Figure 3.2: Non-linear mapping between intensities of images u and v shown in Figure 3.1. Joint intensity histogram $h(u, v)$ is a background of the image (black colour represent high value of histogram), blue line - $E(V|U)$, green line - B-Spline approximation of $E(V|U)$, red line - variance $Var(V|U)$, light blue line - variance approximation. Dark regions in $h(u, v)$ represent big regions of images with approximately constant intensity.

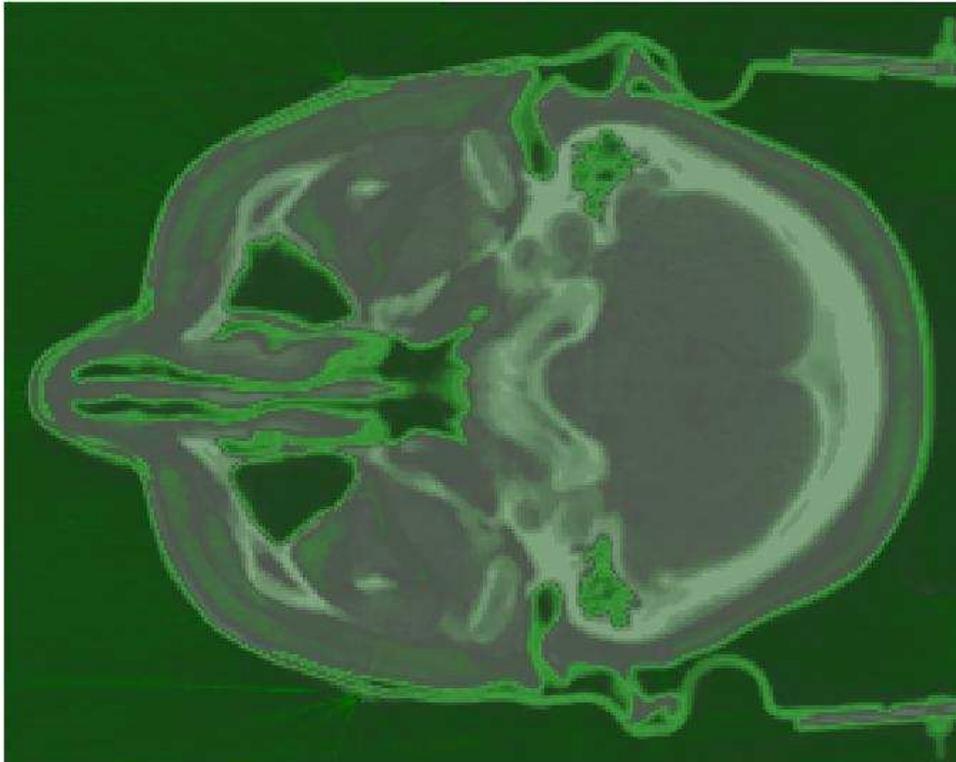


Figure 3.3: Variance $Var(V|U)$ is overlapped on image u as green component. The variance $Var(V|U)$ is a function of intensity. The value of variance was calculated for each coordinate (x, y) by mapping intensity of this pixel $u(x, y)$ and then displayed as green component. It is visible that regions of high variance are spread along edges where both images u and v are misaligned.

3.2 Histogram matching

If images are significantly misaligned, then robust regression can produce faulty mapping. Better results are achieved by histogram matching [37].

Histogram matching is finding such mapping f which after applying to reference image u gives image $f(u)$ for which intensity histogram is similar to intensity histogram of source image v . Given u or v , intensity histogram h_u or h_v can be calculated for each of them and the cumulative histograms are obtained as follows:

$$H_u(x) = \int_0^x h_u(u)du, \quad H_v(x) = \int_0^x h_v(v)dv \quad (3.8)$$

For each intensity value u_i of the reference image the best matching value $f(u_i)$ of the source image is found as follows:

$$f(u_i) = \arg \min_{v_j} |H_u(u_i) - H_v(v_j)| \quad (3.9)$$

Obtained function f maps intensity of the reference image so that it has a histogram similar to that of the source image.

The difference between approximation and histogram matching is presented in the following experiment. Source and reference images were taken from Figure 3.1. Additionally the source image was spatially deformed to form the third image. As a result reference and deformed source images are misaligned. Intensity mapping between reference and source images was compared to intensity mapping between reference and deformed source images. Intensity mapping was estimated by approximation and histogram matching. The results are shown in Figure 3.4 and Figure 3.5. Approximated intensity mapping significantly differs from other mappings in the case of misaligned images. Remaining mappings are close to each other. It is common that mapping obtained by histogram matching is close to mapping approximated from the registered images. However, histogram matching can be ambiguous for certain images.

3.3 Bi-functional mapping

For multi-modal images intensity correspondence is not a unique function. For two identical images, as shown in Figure 3.6a, the mapping is an identity function and joint intensity histogram is a deterministic identity line. For single-modal images, as shown in Figure 3.6b, joint intensity histogram is scattered around curve. Big scatter which can be seen in Figure 3.6b can be modelled as Gaussian noise. For multi-modal images, as shown in Figure 3.6c, the mapping can not be approximated by one curve. In case of Figure 3.6c at least two curves should be used to render variability of data. For the

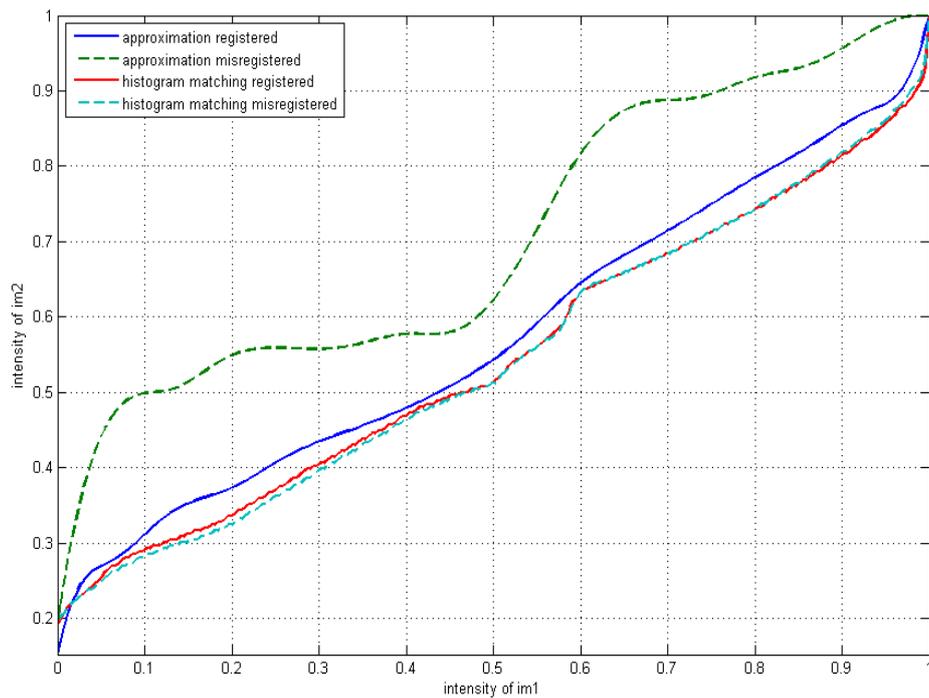
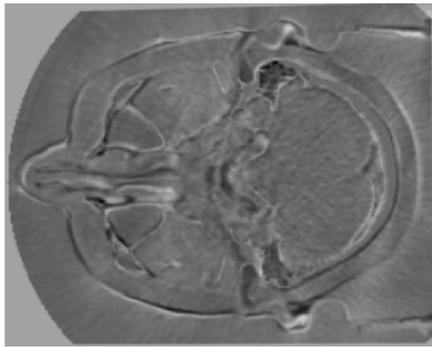
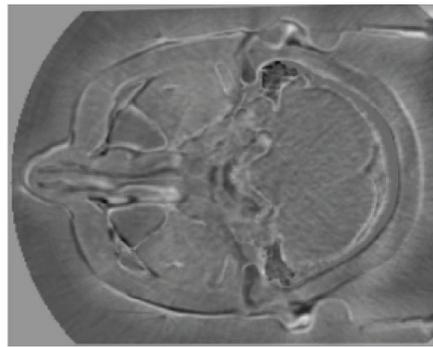


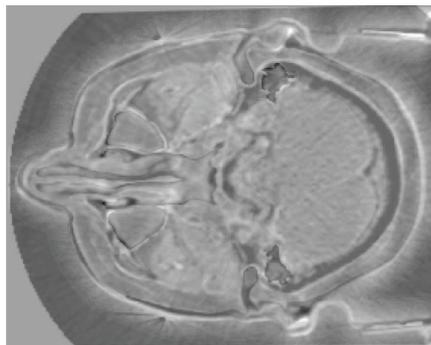
Figure 3.4: Non-linear mappings between corresponding pixel intensities of images u (shown in Figure 3.1a), v (shown in Figure 3.1b) and spatially deformed image v . Blue line - approximation of intensity mapping for pair of aligned images, green dashed line - approximation of intensity mapping for pair of misaligned images, red line - intensity mapping obtained by histogram matching for pair of aligned images, light blue dashed line - intensity mapping obtained by histogram matching for pair of misaligned images.



(a)

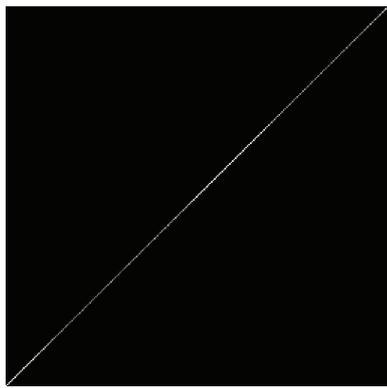


(b)

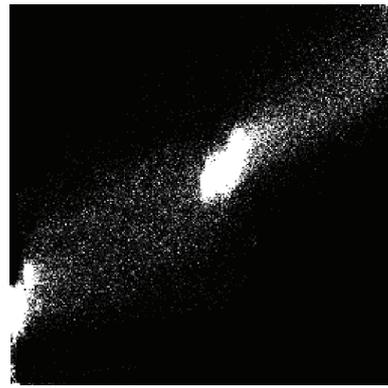


(c)

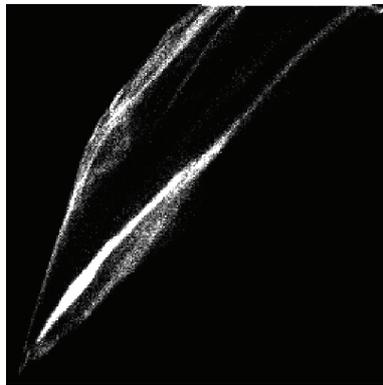
Figure 3.5: Non-linear correction of images shown in Figure 3.1a, Figure 3.1b and spatially deformed image shown in Figure 3.1b. The images were corrected using mapping shown in Figure 3.4. (a) - intensity correction approximated from pair of aligned images, (b) - intensity correction calculated by histogram matching of misaligned images, (c) - intensity correction approximated from pair of misaligned images



(a)



(b)



(c)

Figure 3.6: Three joint intensity histograms (a) same images, (b) single-modal images, (c) multi-modal images

first two categories intensity mapping can be approximated by a function, for the last category different methods should be used.

The authors of [38] applied bi-functional approximation for multi-modal images. They assumed that intensity mapping is determined by two polynomial functions:

$$f_1(u) = \theta_0 + \theta_1 \cdot u + \theta_2 \cdot u^2 + \dots + \theta_k \cdot u^k \quad (3.10)$$

$$f_2(u) = \psi_0 + \psi_1 \cdot u + \psi_2 \cdot u^2 + \dots + \psi_k \cdot u^k \quad (3.11)$$

and modeled at every point (x, y) as mixture of Gaussians:

$$P(v|u) = \pi_1(u) \cdot N(f_1(u), \sigma) + \pi_2(u) \cdot N(f_2(u), \sigma) \quad (3.12)$$

$$\pi_1(u) + \pi_2(u) = 1 \quad (3.13)$$

In (3.12) and (3.13) the following abbreviation is used with $u = u(x, y)$ and $v = v(x, y)$. When it is not ambiguous, $u(x, y)$ or $v(x, y)$ will be replaced by u or v . $N(\mu, \sigma)$ denotes normal distribution with mean μ and standard deviation σ .

The procedure can be divided into parameter estimation and intensity evaluation. To estimate θ they applied least trimmed squares (LTS) and binary reweighted least squares (RLS) estimation [38]. Outliers not used to estimate θ were used to estimate ψ . n_1 pixels taken to estimate θ were used to calculate σ_1 , and n_2 pixels taken to estimate ψ were used to calculate σ_2 . Calculation of σ_1 and σ_2 required correction of bias because they were not calculated from random sample of pixels [38], [39]. Then π_1, π_2, σ were calculated according to:

$$\sigma = \frac{n_1}{n_1 + n_2} \cdot \sigma_1 + \frac{n_2}{n_1 + n_2} \cdot \sigma_2, \quad (3.14)$$

$$\pi_\epsilon(u) = \frac{n_\epsilon}{n_1 + n_2}, \quad \epsilon = 1, 2 \quad (3.15)$$

Evaluation of the mapping function was implemented as weighted sum:

$$f(u, v) = P(\epsilon = 1|u, v) \cdot f_1(u) + P(\epsilon = 2|u, v) \cdot f_2(u) \quad (3.16)$$

where weights are calculated as follows:

$$P(\epsilon|u, v) = \frac{\pi_\epsilon(u) \cdot G_\sigma(v - f_\epsilon(u))}{\pi_1(u) \cdot G_\sigma(v - f_1(u)) + \pi_2(u) \cdot G_\sigma(v - f_2(u))} \quad (3.17)$$

and $G_\sigma(\cdot)$ denotes Gaussian function.

As a result effective mapping was mixture of two functions f_1 and f_2 with weights defined by pair of corresponding pixels taken from both images. The method has been successfully applied to intensity correction of CT and MR images. Registration algorithms with bi-functional intensity correction outperformed NMI registration on CT and MR images [38].

3.4 Intensity inhomogeneity correction

Intensity inhomogeneity is a common problem of magnetic resonance (MR) images and cone beam computed tomography (CBCT) images. There are methods to normalise intensity of MRI [32]. However, they try to estimate unknown intensity distortion having only one image.

In image registration there are two misaligned images which makes problem slightly different. The task is not to detect unknown inhomogeneity but to make intensity of the images to be registered consistent. The problem is that images are misaligned so corresponding pixels have different coordinates. Before registration coordinates of these pixels are unknown.

The authors of [2] applied intensity correction to CBCT and RTPCT images. They used histogram matching with linear correction as the first step. Then they applied rigid registration as the second step. It is assumed that after rigid registration problem of correspondence between pixels is approximately solved. In the third step intensity correction was estimated as pixelwise quotient of the images and then smoothed. Then CBCT image was pixelwise divided by the correction matrix. Before estimation of the mapping bony structure was segmented and removed. Remaining empty regions were filled by interpolation. Similar procedure was applied to air bubbles. The correction enabled the use of CBCT images for dose calculation in radiotherapy.

Another approach is to apply spatial registration which is independent of smooth inhomogeneous intensity distortion [33]. Finite set of reference points is scattered in the reference image. The reference points are surrounded by small rectangular frames of image. Each frame of reference image is rigidly registered to local part of source image. The reference point which is the centre of the frame is displaced together with the frame. In [33] normalized cross correlation coefficient (NCC) was used as a similarity measure. NCC is independent of linear intensity changes so every frame can be registered with different linear intensity distortion. Spatial deformation is estimated in finite set of reference points and then approximated using *spring mass system* [33]. This method is not influenced by smooth changes of intensity mapping under condition that the mapping is so smooth that can be approximated as linear inside single frame. On the other hand the bigger is the frame the better is accuracy of estimated displacement for the frame centre. Therefore choosing proper frame size is matter of compromise.

The authors of [34] demonstrated that estimation of smooth intensity correction and registration can be included in one optimization step. When the algorithm is progressing the displacement field is getting smaller and estimated intensity correction more accurate. In [35] more sophisticated pre-processing of intensity correction was applied. It was demonstrated [35] that SSD based registration with intensity correction can outperform registration based on normalized mutual information (NMI) or correlation ratio (CR).

More detailed description of the algorithm from [34] is presented below as an example.

Local affine registration with correction of brightness and contrast

The authors of [34] applied intensity correction together with spatial registration in the same optimisation algorithm. The displacement field and intensity correction were parameterised as local affine transformation.

They used the following cost function which was derived from brightness constancy assumption [24]:

$$f_b(\mathbf{p}(x, y)) = \sum_{(x, y) \in L} (p_7(x, y) \cdot u(x, y) + p_8(x, y) - v(x', y'))^2 \quad (3.18)$$

where

$$x' = x + p_1(x, y) \cdot x + p_2(x, y) \cdot y + p_3(x, y) \quad (3.19)$$

$$y' = y + p_4(x, y) \cdot x + p_5(x, y) \cdot y + p_6(x, y) \quad (3.20)$$

Image transformation is determined by 8 parameters for each pixel of the image. To make displacement field smooth authors of [34] introduced the following regularisation term:

$$f_s(\mathbf{p}(x, y)) = \sum_{(x, y) \in L} \sum_{i=1}^8 \lambda_i \left[\left(\frac{\partial p_i(x, y)}{\partial x} \right)^2 + \left(\frac{\partial p_i(x, y)}{\partial y} \right)^2 \right] \quad (3.21)$$

The regularization term penalizes big variation of \mathbf{p} . Coefficients λ_i control variation for each component of \mathbf{p} separately. To avoid pure intensity modulation of registered images coefficients $\lambda_7(x, y)$ and $\lambda_8(x, y)$ should be carefully chosen with respect to $\lambda_1(x, y), \dots, \lambda_6(x, y)$.

The authors of [34] optimised the following cost function

$$f(\mathbf{p}(x, y)) = f_b(\mathbf{p}(x, y)) + f_s(\mathbf{p}(x, y)) \quad (3.22)$$

where f_b are forces defined by brightness difference and f_s is regularization term. (3.22) was linearised and solved by explicit discretisation.

They applied the Expectation Maximization (EM) step [36] to deal with outliers. The EM step allows images to be registered to have missing components in one image.

The algorithm was tested on natural and artificial images and was able to recover spatial deformations and intensity distortion in range encountered in typical CT and MR images.

3.5 Summary

The investigation was carried out in intensity correction methods based on global non-linear mapping or smooth intensity correction. As far as medical images are concerned there are much less publications about intensity correction for image registration than about inhomogeneity correction of MR and CT images which is a slightly different problem.

The problem considered here is to make intensity of two images more consistent in order to improve accuracy of spatial registration. The most important obstacle is that images to be corrected are misregistered. To overcome this difficulty the most representative methods use global intensity correction or intensity correction which is smoother than estimated displacement field (e.g. [34]). Very few methods take into account discontinuities (e.g. bi-functional mapping [38], [2]). Application of intensity correction for image registration is justified because it was reported that registration with intensity correction can outperform registration with application of common multimodality similarity measures [38], [35].

Chapter 4

Image registration with intensity correction

4.1 Intensity mapping and image registration

Most registration algorithms impose regularisation constraints on estimated deformation [4], [20] and intensity correction [34]. Regularisation of the displacement field is justified by the nature of deformation of physical objects. In the case of intensity mapping there is also another justification. If two images represent the same object, then they are geometrically similar to each other. When they are better aligned, then intensity mapping between them is predictable and has low complexity. The complexity can be measured as description length of intensity mapping [41], [42].

If the intensity mapping is allowed to be any function it is possible to map any two images on each other. It is expected that intensity mapping estimated from two registered images which represent the same object is simple. Simple means restricted to simple models which can be specific to the class of images to be registered. Therefore if during registration intensity correction is estimated then a simple model should be used. For example smooth mapping might be appropriate for single-modal images and piecewise smooth for multi-modal images. Piecewise smooth means that intensity mapping consists of compact segments of smooth mapping. Such mapping allows discontinuity only along borders between segments.

Multi-modal images can be registered without explicit intensity correction. Normalized mutual information (NMI) [8], [9] and Correlation ratio (CR) [11] are used to register such images. However, they contain implicit estimation of intensity mapping. The way the intensity mapping is estimated is fixed. As a result the class of images which can be registered with similarity measure is also fixed. There is no perfect similarity measure and each similarity measure can be outperformed by others on some class of images (Section 2.3.6).

The author decided to avoid applying multi-modal similarity measures because they are inflexible. SSD with intensity correction is used instead. Intensity correction was restricted to simple models. Our registration algorithm is optimising cost function with respect to spatial deformation and intensity correction simultaneously. Selecting a proper class of intensity correction makes the algorithm specific to some class of images.

Here only RTPCT and CBCT images are considered. For this class the following intensity mappings were selected: global non-linear mapping and smooth local affine mapping. In the case of 2-d affine registration local affine intensity mapping was restricted to additive bias.

4.2 Affine 2-d registration with intensity correction

In the case of 2-d affine registration the following intensity correction models are considered:

- global linear mapping
- global non-linear mapping
- smooth additive bias

4.2.1 Linear intensity correction

Linear correction is included in the same optimisation algorithm as affine registration. Spatial registration was described in Section 2.4.1. This Section describes the part related to intensity correction. Now vector \mathbf{p} contains 2 additional components responsible for intensity correction. The model of intensity correction described here is a simplified version of the model from [34].

Cost function Additional parameters are contrast p_7 and bias p_8 which control intensity correction. They are applied to reference image u instead of v to simplify solution. As a result the cost function is similar to that defined in [34]:

$$\begin{aligned}
 f(\mathbf{p}) &= \sum_{(x,y) \in L} (p_7 \cdot u(x,y) + p_8 - v_{T_P}(x,y))^2 \\
 &= \sum_{(x,y) \in L} (p_7 \cdot u(x,y) + p_8 - v(x',y'))^2 \\
 &= \sum_{i=1}^m r_i^2(\mathbf{p}) = \|\mathbf{r}(\mathbf{p})\|^2
 \end{aligned} \tag{4.1}$$

where:

$$r_i(\mathbf{p}) = p_7 \cdot u(x_j, y_i) + p_8 - v(x_i + p_1 \cdot x_i + p_2 \cdot y_i + p_3, y_i + p_4 \cdot x_i + p_5 \cdot y_i + p_6) \quad (4.2)$$

Pixels were sorted then summation over $(x, y) \in L$ was replaced by summation over integer index $i = 1, 2, \dots, m$ where $m = |L|$ is number of pixels.

Jacobian The i -th row of Jacobian is equal to:

$$\begin{aligned} (\nabla_p r_i(\mathbf{p}))^T &= (\nabla_p (p_7 \cdot u(x_i, y_i) + p_8 - v(x'_i, y'_i)))^T \\ &= -[v_x \cdot x_i, v_x \cdot y_i, v_x, v_y \cdot x_i, v_y \cdot y_i, v_y, -u(x_i, y_i), -1] \end{aligned} \quad (4.3)$$

The remaining part of algorithm is similar to that described in Section 2.4.1.

4.2.2 Non-linear intensity correction

Intensity correction was separated from spatial registration. In our algorithm intensity correction procedure is alternated with affine spatial registration described in Section 2.4.1. The intensity correction is based on linear regression with respect to B-Spline basis functions.

The intensity correction step searches for such mapping g_P which makes image $g_P(v(x, y))$ as similar as possible to reference image $u(x, y)$. Since g_P is parameterised by $\mathbf{p} = [p_1, p_2, \dots, p_n]$ searching for mapping is equivalent to searching for vector $\hat{\mathbf{p}}$ which minimises the following cost function:

$$\begin{aligned} f(\mathbf{p}) &= \sum_{(x,y) \in L} (u(x, y) - g_P(v(x, y)))^2 \\ &= \sum_{i=1}^m (u(x_i, y_i) - g_P(v(x_i, y_i)))^2 \\ &= \sum_{i=1}^m (u_i - g_P(v_i))^2 \\ &= \sum_{i=1}^m (u_i - \mathbf{b}^T(v_i) \cdot \mathbf{p})^2 \end{aligned} \quad (4.4)$$

Equation (4.4) describes only intensity correction step which is alternated with affine registration step. g_P is parameterised according to a B-Spline model (2.14), in which \mathbf{b} is the vector of B-Spline basis functions and \mathbf{p} is the vector of control points (2.15). The images are normalised to range

[0, 1] so knots are distributed uniformly over this range. The number of knots are chosen to compromise between accuracy and smoothnes of the mapping function. Equation (4.4) is a linear regression equation [36] which is a least-square optimisation problem (Section 2.3.7).

Comparing to (4.1) equation (4.4) has non-linear intensity correction, which is applied to source image v instead of reference u and does not contain spatial deformation. Solution of (4.4) might be ambiguous and can lead to over fitting of estimated function [36]. To avoid this problems regularisation term $\lambda \cdot \mathbf{p}^T \cdot \mathbf{p}$ was added to (4.4):

$$f(\mathbf{p}) = \sum_{i=1}^m \left(u_i - \mathbf{b}^T(v_i) \cdot \mathbf{p} \right)^2 + \lambda \cdot \mathbf{p}^T \cdot \mathbf{p} \quad (4.5)$$

Setting $\nabla f(\mathbf{p}) = 0$ gives:

$$- \sum_{i=1}^m \mathbf{b}(v_i) \left(u_i - \mathbf{b}^T(v_i) \cdot \mathbf{p} \right) + \lambda \cdot \mathbf{p} = 0 \quad (4.6)$$

which has the following solution:

$$\mathbf{p} = \left(\sum_{i=1}^m \mathbf{b}(v_i) \cdot \mathbf{b}^T(v_i) + \lambda \cdot Id \right)^{-1} \left(\sum_{i=1}^m \mathbf{b}(v_i) \cdot u_i \right) \quad (4.7)$$

Regularisation term $\lambda \cdot \mathbf{p}^T \cdot \mathbf{p}$ shrinks vector \mathbf{p} towards zero [36]. Dumping parameter λ controls the importance of regularisation. The bigger the λ , the smoother is the estimated function.

Figure 4.1 demonstrates the efficiency of our algorithm. Affine deformation and global non-linear distortion were randomly simulated. Affine deformation was simulated by random generation of all 6 parameters with uniform distribution. In the example shown in Figure 4.1 maximum value of initial displacement field was 46 pixels. Despite intensity distortion the registration algorithm recovered the displacement field with accuracy less than 3 pixels. For the same example without intensity correction (described in Section 2.4) the displacement field was recovered with the error of 9 pixels. The source image registered with correction (Figure 4.1d) is similar to the reference image (Figure 4.1b). They are not only spatially aligned (Figure 4.1f) but also their intensity values agree.

Figure 4.2 shows the quality of spatial registration versus non-linear intensity distortion for different correction models. Non-linear intensity distortion was generated as a sum of identity function and integral of some smooth curve:

$$g(y) = y \cdot (1 - p) + \frac{\int_0^y h(y) dy}{\int_0^1 h(y) dy} \cdot p \quad (4.8)$$

where $h(y)$ is an arbitrary smooth function over range $[0, 1]$ which follows $h(0) = 0$ and $h(y) \geq 0$. In the experiment the function $h(y)$ was a B-Spline curve with randomly generated values of control points and uniformly distributed knots. Parameter p describes non-linearity and if $p = 0$ then $g(y) = y$ is an identity function and when $p = 1$ then $g(y)$ has biggest level of non-linearity.

The best results are achieved for global non-linear correction which agree with the distortion model used for simulation. From the results of registration with different correction models shown in Figure 4.2, registration with any form of corrections is seen to be better than no correction at all.

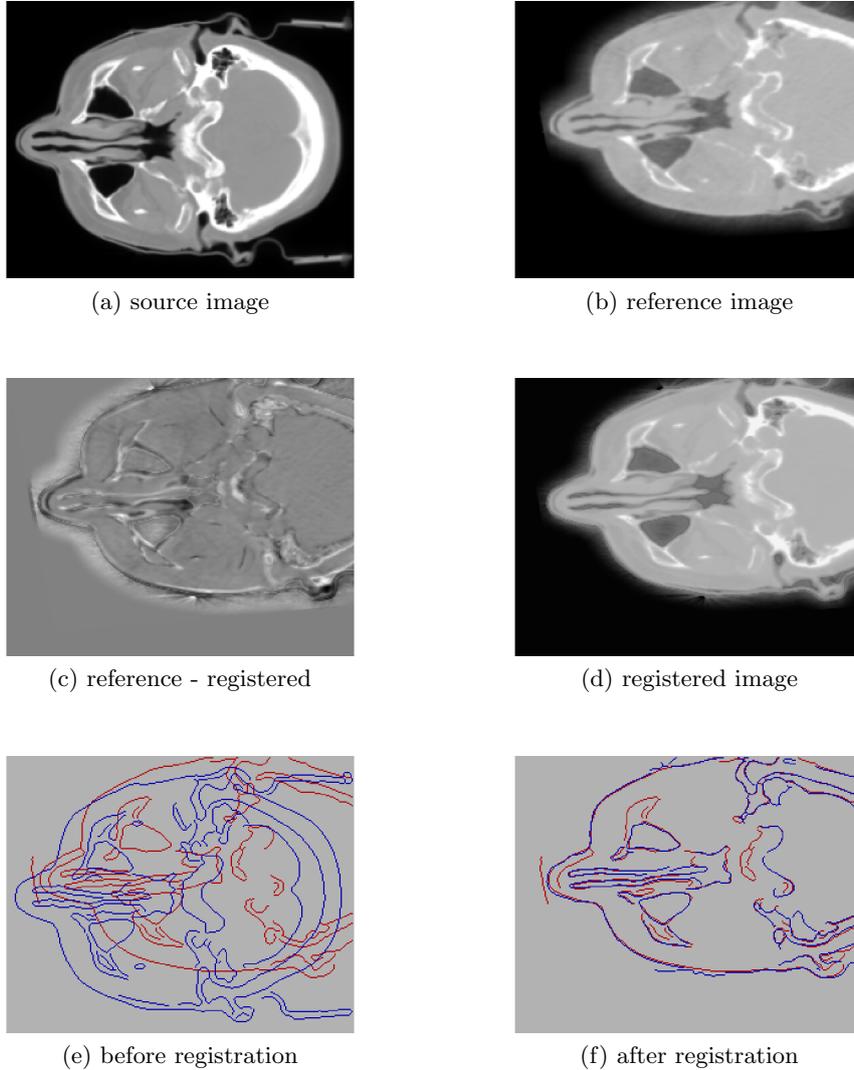


Figure 4.1: 2-d affine registration of RTPCT and CBCT slices of size 226x282. Reference image is an affine transformed version of CBCT slice. Apart from affine deformation global non-linear intensity distortion was introduced. Spatial deformation and intensity distortion were known before registration and maximum value of the displacement field of spatial deformation was 46 pixels. During registration non-linear intensity correction was applied. The registration was successful because maximum discrepancy between simulated and registered displacement fields was reduced to 2.8 pixels (which is 3 times better than in the same case but without intensity correction Figure 2.22).

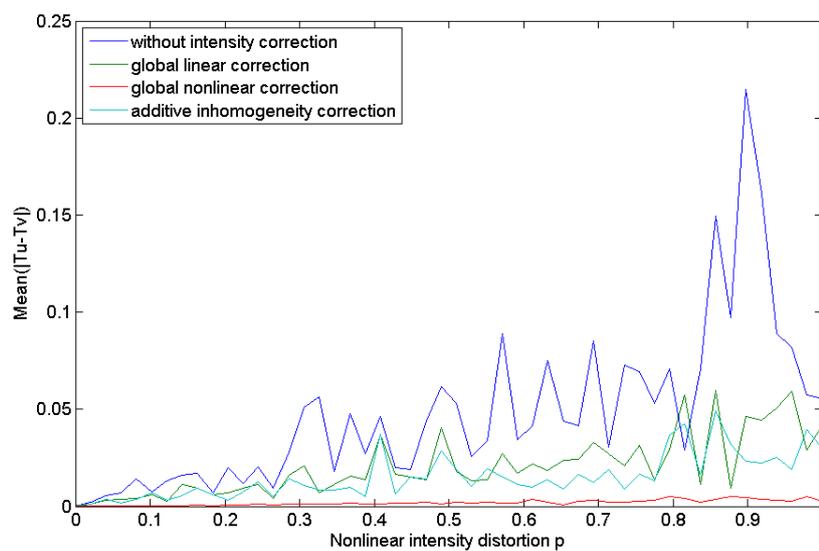


Figure 4.2: Mean difference between simulated and estimated displacement fields versus non-linearity of intensity mapping. RTPCT and CBCT slices normalised to $[0,1]$ range were used as input data. Both images were manually registered and had intensity normalised before experiment. Affine deformation was simulated together with intensity distortion. The best registration result is produced by the global non-linear intensity correction model which was the same as the model used in simulation.

4.2.3 Additive inhomogeneity correction

In the algorithm proposed in this Section inhomogeneity correction is separated from spatial registration. The optimisation procedure has two steps: intensity correction step and spatial registration step. The iterative algorithm is similar to non-linear correction described in Section 4.2.2 and alternates both of these steps.

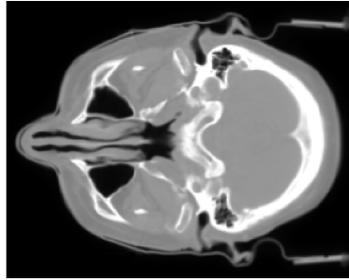
To make image v identical to image u it is enough to add difference $u - v$ to v . This way intensity correction can be calculated if images are aligned. If images are misaligned it is assumed that pairs of corresponding pixels from different images are close to each other. So it is reasonable to blur the difference by convolving with Gaussian kernel. The kernel parameters should be chosen according to the predicted maximum displacement field. Not enough smoothing causes spatial registration to fail and too much smoothing prevents correction from following intensity distortions (Figure 4.5).

For testing purposes smooth intensity distortion was added to reference images. Then the images were spatially deformed by affine transformation. The intensity distortion was modeled as 2-d B-Spline surface with randomly generated control points and uniformly distributed knots. Regularity of distortion was controlled by number of knots (Section 2.3.5).

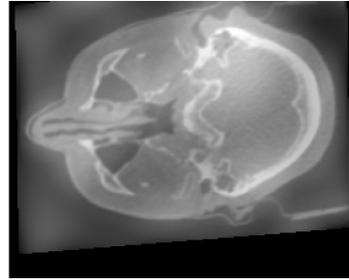
A result of registration is presented in Figure 4.3, where the source image registered with correction (Figure 4.3d) is seen to be similar to the reference image (Figure 4.3b). Not only they are spatially aligned (Figure 4.3f) but also their intensity values agree.

Despite inhomogeneous intensity distortion the registration algorithm recovered the displacement field with accuracy less than 1.25 pixel which was more than 46 pixels before registration. For the same example without intensity correction (described in Section 2.4) the displacement field after registration was even larger increasing error to 58 pixels.

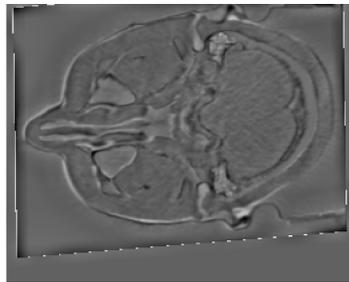
Figure 4.4 shows the quality of spatial registration versus additive intensity distortion. The best result is achieved for local additive correction which agrees with the distortion model used in simulation. From the results of registration with different correction models shown in Figure 4.4, registration with any form of corrections is seen to be better than no correction at all.



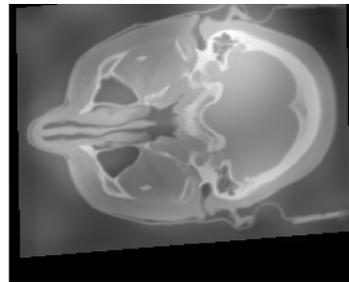
(a) source image



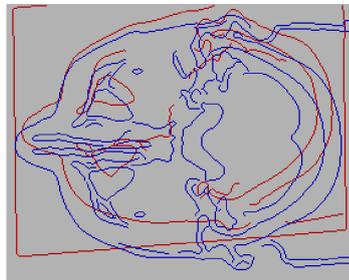
(b) reference image



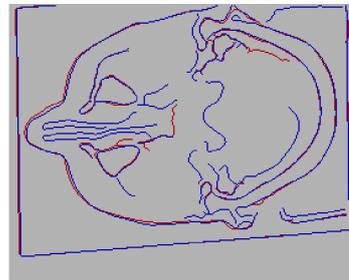
(c) reference - registered



(d) registered image



(e) before registration



(f) after registration

Figure 4.3: 2-d affine registration of RTPCT and CBCT slices of size 226x282. Reference image is an affine transformed version of the CBCT slice. Spatial deformation was known before registration and the maximum value of displacement field of this deformation was 46 pixels. Before spatial transformation smooth intensity distortion was added to reference image. During registration smooth additive intensity correction was applied. Registration was successful because maximum discrepancy between simulated and registered displacement fields was reduced to 1.25 pixels. The same registration algorithm without intensity correction only increased misalignment to 58 pixels (Figure 2.23).

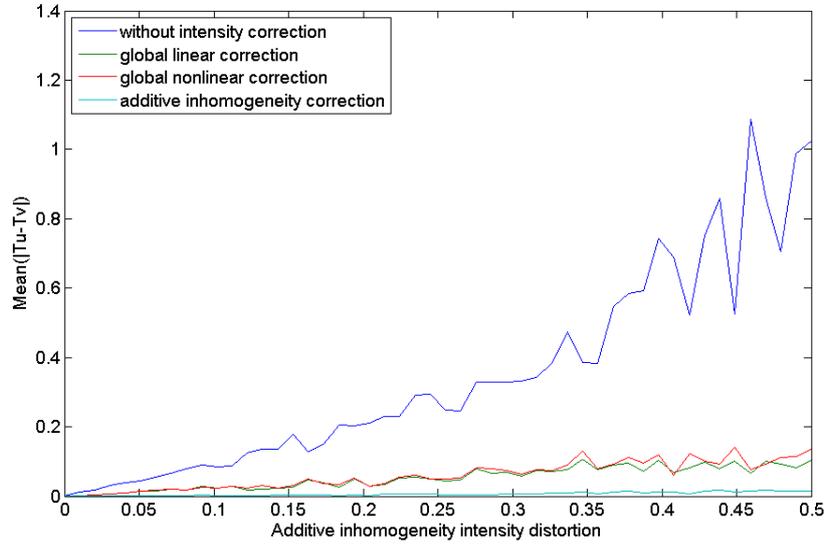


Figure 4.4: Mean difference between simulated and estimated displacement fields versus maximum value of additive intensity distortion.

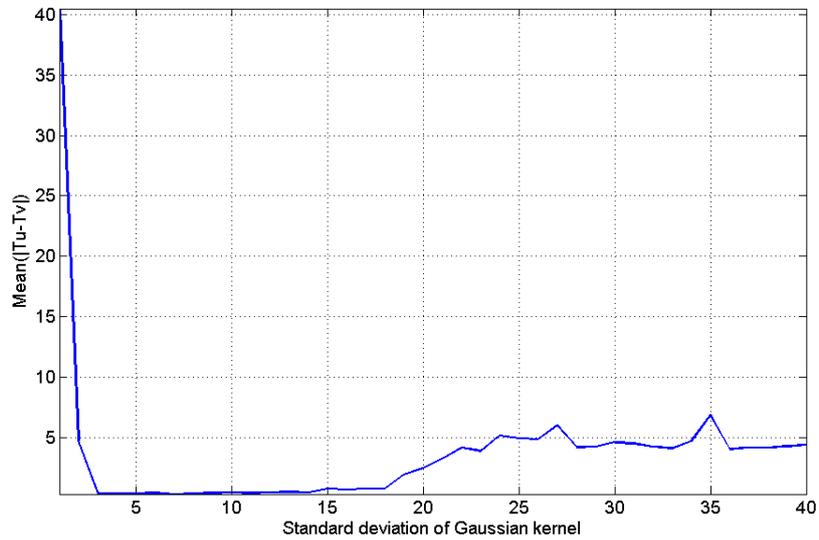


Figure 4.5: Mean difference between simulated and estimated displacement fields versus standard deviation of Gaussian kernel in additive intensity correction.

4.2.4 Concluding remarks

- Intensity correction significantly improves registration accuracy (Figure 4.4). The higher accordance of correction model with the simulation model the higher accuracy of spatial registration.

4.3 Nonrigid B-Spline 1-d registration

The algorithm for 1-d deformable registration with intensity correction is similar to the algorithm described in Section 2.5. The difference is that parameter vector \mathbf{p} is longer because, besides deformation part, it contains components responsible for intensity correction. Two intensity correction models are considered:

- global non-linear mapping
- smooth local affine

Both models are described in the following sections. The description is concentrated on elements which are missing in Section 2.5.

4.3.1 Global non-linear intensity correction

Non-linear correction consists of applying non-linear function F_P to all pixels of an image. Mapping F_P is independent of pixel coordinates. It is assumed that F_P is smooth and can be approximated as a B-Spline curve.

Transformation and non-linear correction Spatial transformation and non-linear correction are parameterised as B-Spline curves

$$\begin{aligned} T_{P_1}(x) &= x + \mathbf{b}_1^T(x) \cdot \mathbf{p}_1 = x' \\ F_{P_2}(y) &= \mathbf{b}_2^T(y) \cdot \mathbf{p}_2 = y' \end{aligned}$$

where $\mathbf{b}_1(x)$, $\mathbf{b}_2(y)$ are vectors of B-Spline basis functions, and \mathbf{p}_1 , \mathbf{p}_2 are vectors of control points with lengths n_1 and n_2 (Section 2.3.5, Section 2.5). In both cases knots are distributed uniformly. In the case of T_{P_1} domain spreads over image point positions and in the case of F_{P_2} domain spreads over intensity levels. Both vectors form joint parameter vector $\mathbf{p} = [\mathbf{p}_1, \mathbf{p}_2]^T$ with length $n = n_1 + n_2$.

Cost function

$$\begin{aligned}
f(\mathbf{p}) &= \sum_{x \in L} \left(F_{P_2}(u(x)) - v_{T_{P_1}}(x) \right)^2 \\
&= \sum_{x \in L} \left(\mathbf{b}_2^T(u(x)) \cdot \mathbf{p}_2 - v(x + \mathbf{b}_1^T(x) \cdot \mathbf{p}_1) \right)^2 \\
&= \sum_{i=1}^m r_i^2(\mathbf{p}) = \|\mathbf{r}(\mathbf{p})\|^2
\end{aligned} \tag{4.9}$$

where summation over x was exchanged by summation over integer index $i = 1, 2, \dots, m$ with $m = |L|$ being number of points. Comparing to (4.1) equaton (4.9) instead of linear contain non-linear spatial deformation and non-linear intensity correction.

Jacobian the i -th row of Jacobian is equal to:

$$(\nabla_p r_i(\mathbf{p}))^T = [-v_x \cdot \mathbf{b}_1^T(x_i), \mathbf{b}_2^T(u(x_i))] \tag{4.10}$$

Jacobian $J(\mathbf{p})$ and vector $\mathbf{r}(\mathbf{p})$ are necessary data for the Levenberg-Marquardt algorithm (Section 2.3.7).

4.3.2 Registration examples

Our experiments are similar to those described in Section 2.5. Image $u(x)$, spatial deformation T_u , and intensity distortion F_u are known before registration. They were randomly generated using B-Spline curves.

Frequency components of registered images and displacement fields were restricted to make recovery of simulated displacement field feasible. The main issue was to keep maximum frequency component of the image twice bigger than the maximum frequency component of the displacement field (similar to Section 2.5.2). Apart from this the maximum value of the displacement field was kept smaller than half period of the maximum frequency component of the image (Figure 2.3). The rules are given approximately in terms of knot spacing and are formalized as follows:

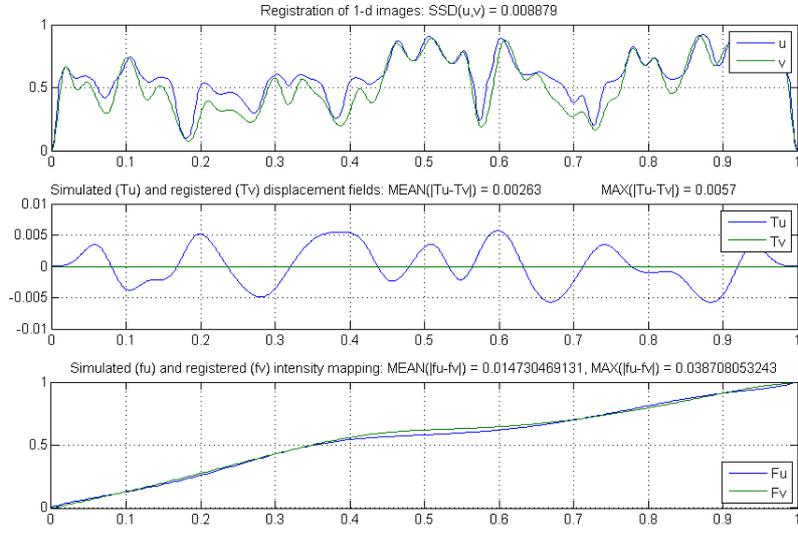
- Knots are uniformly distributed.
- The image has twice as many knots than the displacement field.
- The maximum value of the displacement field is less than the distance between knots in image.

Reference image u is formed from source image v according to $u(x) = F_u(v(T_u(x)))$. Non-linear intensity distortion F_u was generated according to (4.8). Input to the registration algorithm are $u(x)$ and $v(x)$ and the goal is

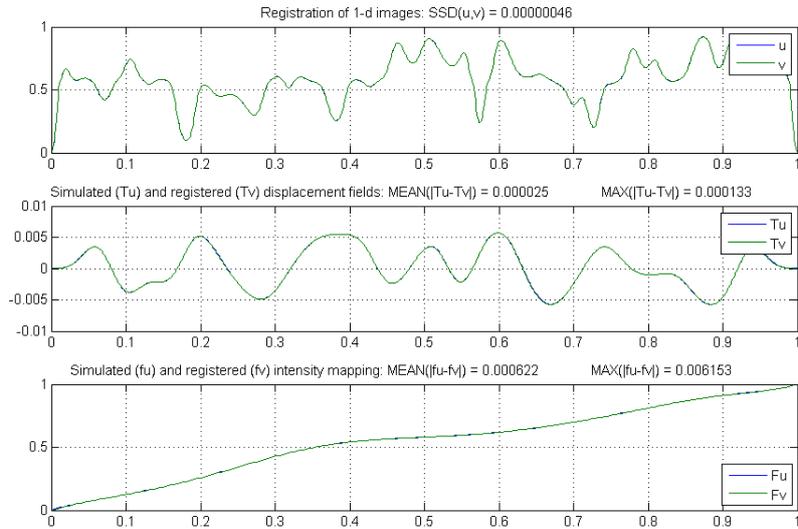
to recover T_u and F_u . F_v was initialised by histogram matching Section 3.2. Figure 4.6 is an example of successful registration with non-linear intensity correction. The goal was achieved with high accuracy.

Figure 4.7 demonstrates the significance of non-linear intensity correction for registration accuracy. In this example the model used for intensity correction was the same as the model for intensity distortion.

To demonstrate sensitivity of the registration algorithm to the deformation model, the number of knots used for estimation and their positions were varied in a way similar to that described in Section 2.5.2. Figure 4.8, shows how accuracy of registration depends on the number of knots in the registration model. There is a narrow local minimum when the number of knots is equal to 32 which match with the simulation model. Registration accuracy is acceptable when the number of knots is about twice bigger than the number of knots in the simulation model.

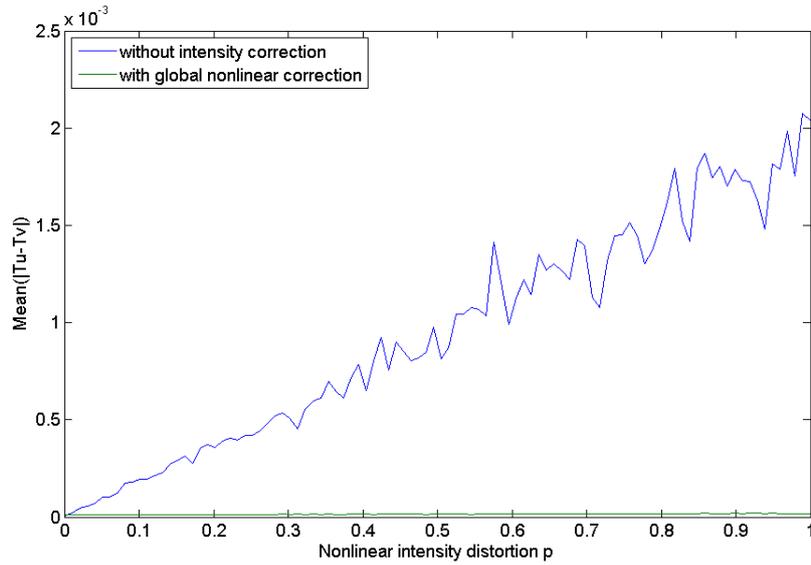


(a) before registration

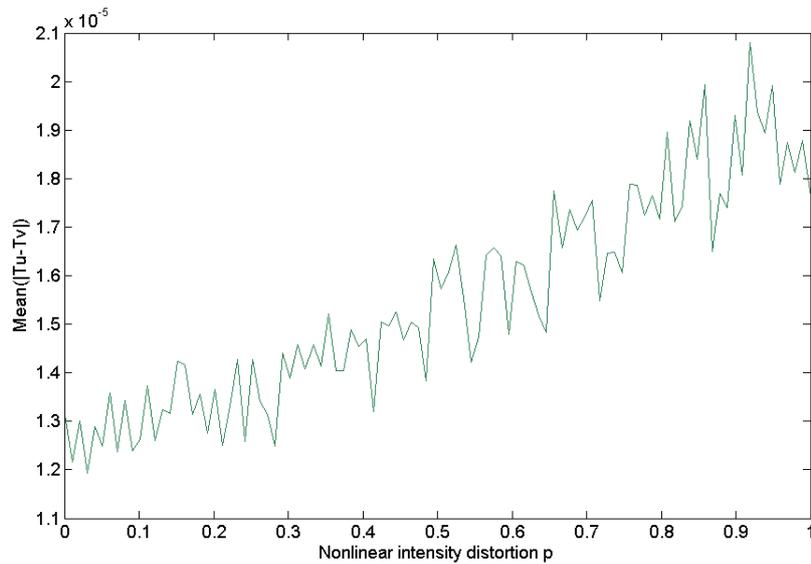


(b) after registration

Figure 4.6: 1-d deformable registration with non-linear intensity correction. Source image v is registered to reference image u . Reference image is a transformed version of source image $u(x) = F_u(v(T_u(x)))$. F_u is simulated intensity distortion and T_u is simulated spatial deformation. Displacement field T_v and correction function F_v are estimated during registration. After registration $T_u \approx T_v$ and $F_u \approx F_v$.



(a) graph with intensity correction and without



(b) graph with intensity correction

Figure 4.7: Mean difference between simulated T_u and estimated T_v displacement fields after registration versus non-linear intensity distortion. Non-linear intensity correction significantly improves accuracy of spatial registration. The model used for intensity correction was the same as the model used for simulation of intensity distortion.

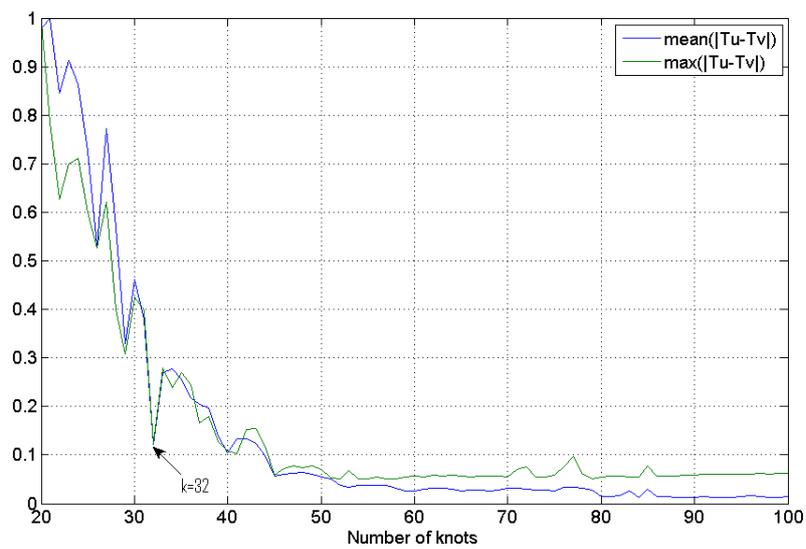


Figure 4.8: Mean and maximum discrepancy between simulated T_u and estimated T_v transformation versus number of knots used for registration. Both graphs were normalised to range $[0, 1]$ to show regularity rather than absolute values. Knots for registration were randomly deviated from uniform positions so that they do not match knots used for simulation. Number of knots in simulated transformation was equal to 32 and for this value the graph has a narrow minimum.

4.3.3 Affine inhomogeneity correction

As described in the previous section, intensity correction mapping is independent of pixel position. Here position dependent correction was introduced. Correction mapping consists of two components: contrast $A(x)$ and bias $B(x)$. Both components $A(x)$ and $B(x)$ are parametrized by B-Spline control points. Correcting image u consists of local affine mapping according to equation $A(x) \cdot u(x) + B(x)$. It is assumed that both functions $A(x)$ and $B(x)$ are smooth so they are approximated by B-Spline curves.

Transformation and non-linear correction Spatial transformation and affine correction are parameterised as B-Spline curves

$$\begin{aligned} T_{P_1}(x) &= x + \mathbf{b}_1^T(x) \cdot \mathbf{p}_1 = x' \\ A_{P_2}(x) &= \mathbf{b}_2^T(x) \cdot \mathbf{p}_2 \\ B_{P_3}(x) &= \mathbf{b}_3^T(x) \cdot \mathbf{p}_3 \end{aligned} \quad (4.11)$$

where $\mathbf{b}_1(x)$, $\mathbf{b}_2(x)$, $\mathbf{b}_3(x)$ are vectors of B-Spline basis functions, and \mathbf{p}_1 , \mathbf{p}_2 , \mathbf{p}_3 are vectors of control points with lengths n_1 , n_2 , and n_3 (Section 2.3.5, Section 2.5). In all cases knots are distributed uniformly. All three parameter vectors form joint parameter vector $\mathbf{p} = [\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3]^T$ with length $n = n_1 + n_2 + n_3$.

Cost function

$$\begin{aligned} f(\mathbf{p}) &= \sum_{x \in L} \left(A_{P_2}(x) \cdot u(x) + B_{P_3}(x) - v_{T_{P_1}}(x) \right)^2 \\ &= \sum_{x \in L} \left(\mathbf{b}_2^T(x) \cdot \mathbf{p}_2 \cdot u(x) + \mathbf{b}_3^T(x) \cdot \mathbf{p}_3 - v(x + \mathbf{b}_1^T(x) \cdot \mathbf{p}_1) \right)^2 \\ &= \sum_{i=1}^m r_i^2(\mathbf{p}) = \|\mathbf{r}(\mathbf{p})\|^2 \end{aligned} \quad (4.12)$$

In the above equation summation over x is replaced by summation over integer index $i = 1, 2, \dots, m$, where $m = |L|$ is number of points.

Jacobian the i -th row of Jacobian is equal to:

$$(\nabla_p r_i(\mathbf{p}))^T = [-v_x \cdot \mathbf{b}_1^T(x_i), \mathbf{b}_2^T(x_i) \cdot u(x_i), \mathbf{b}_3^T(x_i)] \quad (4.13)$$

4.3.4 Registration examples

These experiments are similar to those described in Section 2.5. Image $u(x)$, spatial deformation T_u , and affine intensity distortion A_u , B_u are known before experiments. They were randomly generated using B-Spline curves.

Similar to Section 4.3.2 frequency components of registered images, displacement fields and intensity distortion were restricted to make registration feasible. The maximum frequency component of the displacement field was kept twice smaller than the maximum frequency component of the image. Also the maximum frequency component of the intensity distortion was twice smaller than the maximum frequency component of the displacement field. Apart from this the maximum value of the displacement field was kept smaller than half period of the maximum frequency component of the image (Figure 2.3). The rules are given in terms of knot spacing and are formalized as follows:

- Knots are uniformly distributed.
- The image has twice as many knots than the displacement field.
- The maximum value of the displacement field is less than the distance between knots in image
- Displacement field T_u has twice as many knots than intensity distortion A_u and B_u .

Reference image u is formed from source image v according to equations: $u_a(x) = A_u(x) \cdot v(x) + B_u(x)$ and $u(x) = u_a(T_u(x))$. Input to the registration algorithm are $u(x)$ and $v(x)$ and the goal is to recover T_u , A_u and B_u . Figure 4.9 and Figure 4.10 shows examples of successful registration with local affine intensity correction.

Figure 4.11 demonstrates the significance of affine intensity correction for registration accuracy. In this example the model used for intensity correction was the same as the model used for intensity distortion.

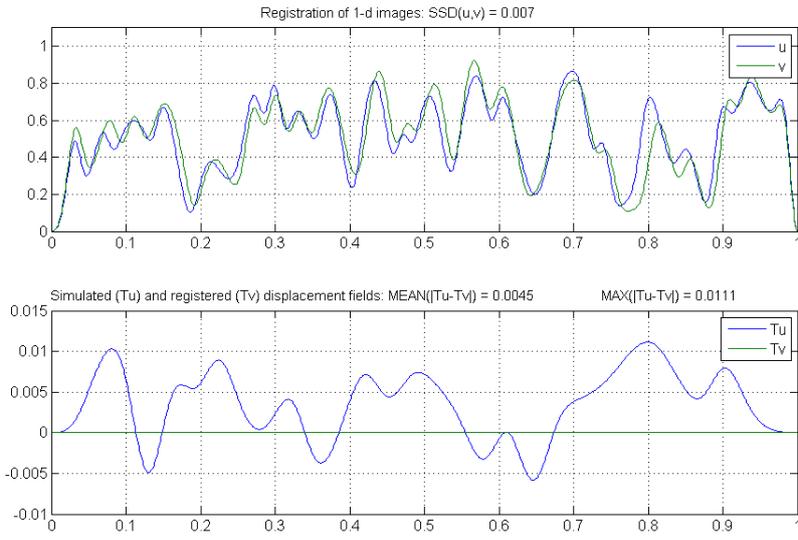
To demonstrate sensitivity of the registration algorithm to the deformation model, the number of knots used for estimation and their positions were varied in a way similar to that in Section 2.5.2. Figure 4.12 and Figure 4.13 show how accuracy of registration depends on the number of knots in the registration model. Figure 4.12 shows that there is a narrow local minimum when the number of knots in registration is equal to 32 which match the number of knots used in the simulation model. If the number of knots is approximately twice bigger than that for simulation, then registration accuracy is acceptable.

4.3.5 Concluding remarks

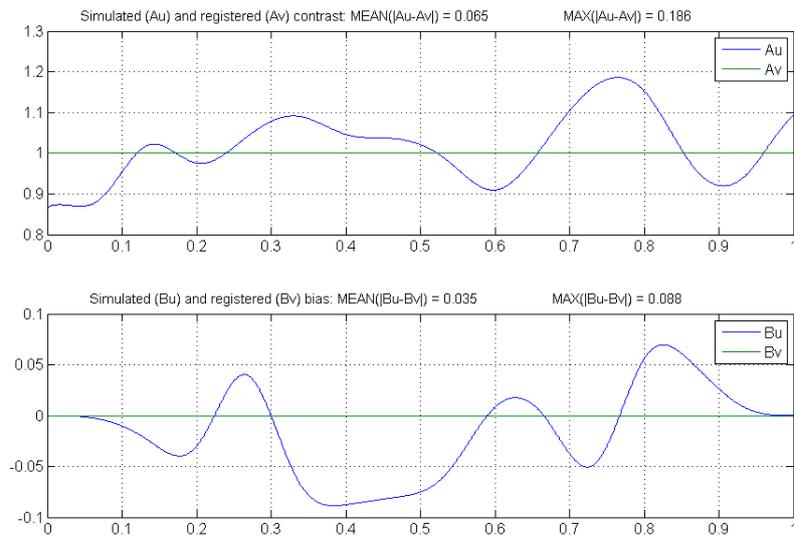
- Sum of squared differences SSD is suitable as a cost function for B-Spline deformable registration with intensity correction. Minimisation of SSD by applying the Levenberg-Marquardt optimisation gives an efficient registration algorithm.
- If images to be registered and the displacement field fulfil the following conditions
 - the maximum frequency component of the image is twice bigger than the maximum frequency component of the displacement field
 - the maximum value of the displacement field is smaller than half period of the maximum frequency component of the image (Figure 2.3)
 - the maximum frequency component of the displacement field is twice bigger than the maximum frequency component of the intensity distortion

then simulated deformation and intensity distortion can be recovered.

- If the exact value of the number of knots for best fitting of the displacement field and intensity distortion is unknown, then good results can be achieved for values which are approximately twice bigger than the expected one.

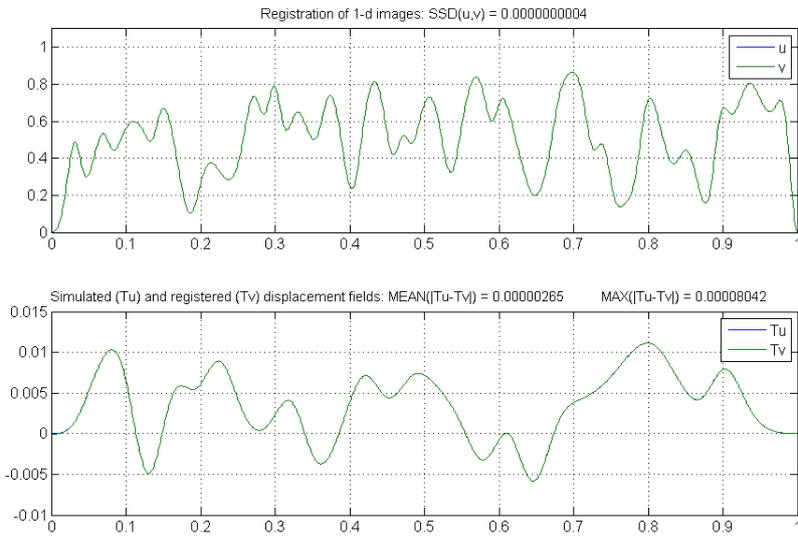


(a) images u and v , deformation T_u and displacement field T_v

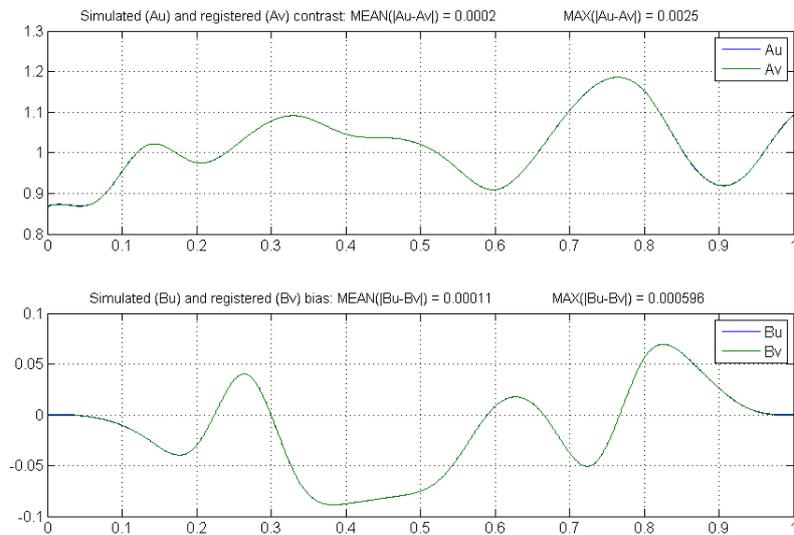


(b) intensity distortion A_u, B_u and correction A_v, B_v

Figure 4.9: Deformable registration with affine intensity correction - before registration. Source image v is registered to reference image u . Reference image is a transformed version of source image. u is composition of affine intensity distortion $u_a(x) = A_u(x) \cdot v(x) + B_u(x)$ and spatial deformation $u(x) = u_a(T_u(x))$. T_v , A_v and B_v were initialised as constant functions and estimated during registration.



(a) images u and v , deformation T_u and displacement field T_v



(b) intensity distortion A_u, B_u and correction A_v, B_v

Figure 4.10: Deformable registration with affine intensity correction - data after registration. Spatial deformation T_u and affine intensity distortion A_u and B_u were known before registration. After registration estimated displacement field is approximately equal to simulated $T_u \approx T_v$, $A_u \approx A_v$ and $B_u \approx B_v$.

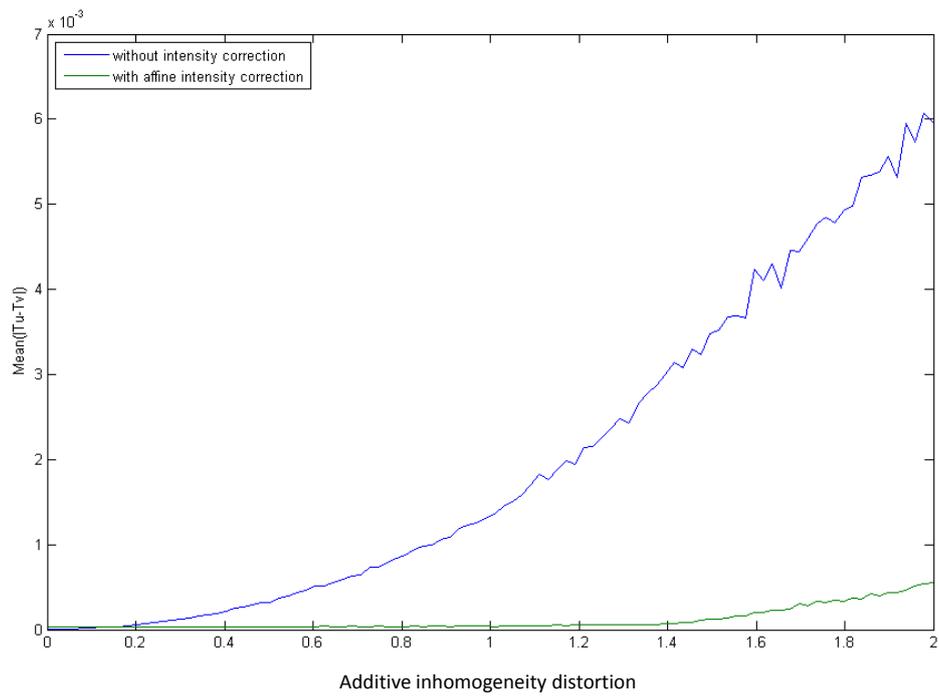


Figure 4.11: Mean difference between simulated and estimated displacement fields after registration versus additive inhomogeneity distortion. Affine intensity correction significantly improves accuracy of spatial registration. The model used for simulation of distortion was a special case of the model used for intensity correction.

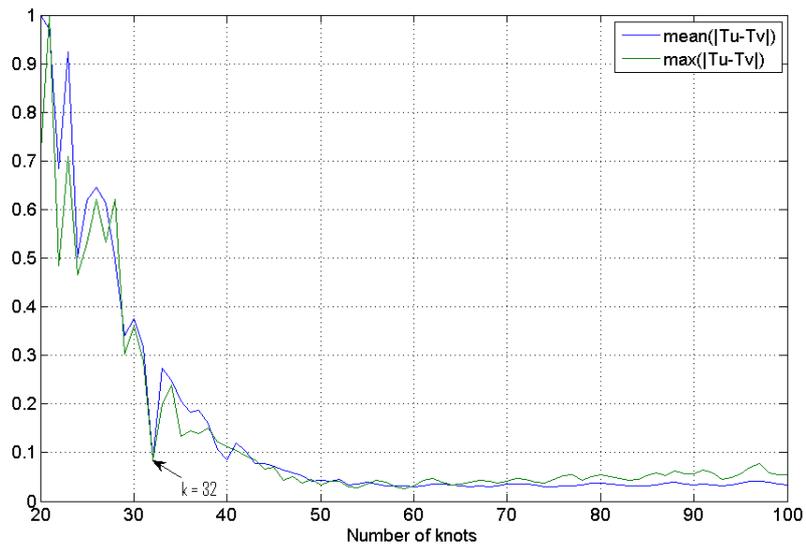


Figure 4.12: Mean and maximum value of difference between simulated T_u and estimated T_v transformation versus number of knots used for registration. Number of knots used for simulation and estimation of affine intensity correction was the same. Both graphs were normalised to range $[0, 1]$ to show regularity rather than absolute values. Knots for registration were randomly deviated from uniform positions so that they do not match knots used for simulation. Number of knots in simulated transformation was equal to 32 and for this value the graph has a narrow minimum.

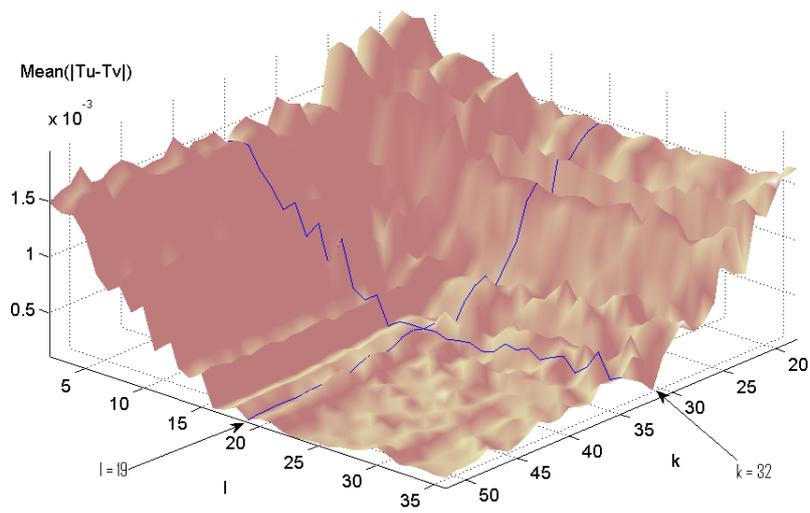


Figure 4.13: Mean difference between simulated T_u and estimated T_v transformation versus number of knots used for estimation of spatial deformation (k) and affine intensity correction (l). Knots for simulation were randomly deviated from uniform positions so that they do not match knots used for estimation. The deviation did not change knot order. Number of knots used for simulation are marked in blue ($l = 19$ and $k = 32$).

4.4 Summary

- Sum of squared differences SSD is suitable as a cost function for B-Spline deformable registration with intensity correction. Minimisation of SSD by applying the Levenberg-Marquardt optimisation gives an efficient registration algorithm.
- If the models used to simulate spatial deformation and intensity distortion are similar to that used to estimate them then SSD based registration is able to recover the simulated deformation and intensity distortion.
- To make recovery of deformation and intensity distortion feasible the frequency component of spatial deformation and intensity distortion should be bounded. Also maximum value of displacement field cannot exceed maximum value determined by frequency component of the image (Figure 2.3).
- Intensity correction applied together with spatial registration improves accuracy of SSD based image registration.

Chapter 5

Conclusions and future work

The research aimed to develop means to increase the accuracy of radiation therapy treatment. Radiation therapy is planned on RTPCT images and validated using CBCT images. CBCT image might contain inhomogeneous intensity distortion which can decrease accuracy of spatial registration.

Intensity based image registration has been applied because it works without human expert interaction. Such registration optimizes similarity measure between registered images. Choice of similarity measure is difficult in presence of intensity distortion. There are multi-modal similarity measures which deal with images with inconsistent intensity mapping. However, they can fail when intensity distortion is smooth inhomogeneous mapping [35]. To overcome this problem author included intensity correction in the registration algorithm. The algorithm optimises sum of squared differences (SSD) with respect to spatial transformation and intensity correction simultaneously. The problem was shifted from choice of similarity measure to choice of model of intensity correction. In fact multi-modal similarity measures like normalized mutual information (NMI) or correlation ratio (CR) have implicit estimation of intensity mapping. The model of the mapping is fixed and may fail to render specific intensity distortion.

2-d affine registration has been implemented with intensity correction. The algorithm was implemented as Levenberg-Marquardt optimisation of SSD. Registration was tested on limited set of medical images and simulated spatial deformation and intensity distortion. The intensity distortion was simulated to resemble typical CBCT image. Two intensity correction models were applied: global non-linear and smooth local additive. Simulations showed that both methods work properly for simulated data. Experiments demonstrated that intensity correction improves accuracy of spatial registration. Tested algorithm was able to recover much bigger spatial deformations than those encountered in medical practice.

Nonrigid B-Spline 1-d registration was implemented with intensity correction. The algorithm was implemented as Levenberg-Marquardt optimisa-

tion with SSD as a cost function. Registration was tested on simulated data. Two intensity correction models were implemented: global non-linear and local affine. Experiments demonstrated that if the simulation model agree with the registration model, then it is possible to recover spatial deformation and intensity distortion. The algorithm was able to recover deformation which did not exceed the theoretical limit on displacement field calculated for registered images. Experiments shows that intensity correction often improves accuracy of spatial registration even when models of intensity correction used for simulation and registration are different.

The idea of separating intensity correction mapping from similarity measure is shown to have advantages. Image registration can be considered as simultaneous estimation of spatial deformation and intensity distortion. Choosing proper models of spatial deformation and intensity distortion is essential. These models can make the registration algorithm specific to the image class of interest. Author expects that demand on increase accuracy of medical image registration will motivate development of spatial deformation and intensity distortion models.

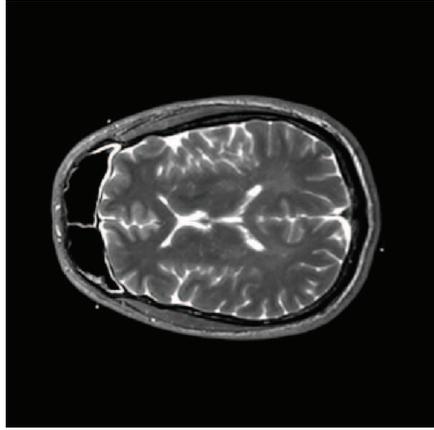
Future extensions to the algorithm

- Application of multi-resolution is required to increase the capture range of the B-Spline registration algorithm.
- Extension of 1-d deformable registration to 2-d is considered as a future step. Implementation could utilise the property of B-Spline basis functions being separable.
- B-Spline models of spatial deformation and intensity distortion might be replaced by more general model which can deal with discontinuities.
- Global non-linear intensity mapping and smooth local affine can be generalised to inhomogeneous non-linear intensity mapping.
- The algorithm should be extended to deal with noisy images and outliers which are generated by components which exist in one image and do not exist in another image.
- *Segmentation based intensity correction.* Most intensity correction methods impose smooth regularisation on correction mapping. The author plans to implement method which can deal with discontinuities. The method is motivated by the assumption that each physical objects or tissues is mapped by one intensity mapping. On the other hand objects or tissues usually form connected regions. As a result image can be divided into connected regions with one mapping function assigned to each region. This procedure is a segmentation, where similarity between pixels in segment is based on proximity to intensity

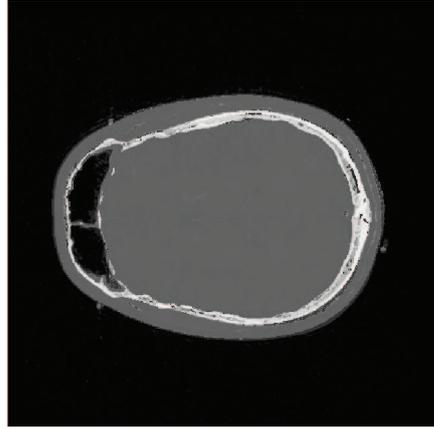
mapping assigned to this segment. Segmentation can be alternated with estimation of intensity mapping inside segments, which is similar to the method described in [45]. Good candidates of segmentation algorithms are graph cuts [43], [44] and level set methods [46].

The author implemented simplified version of this method as preliminary sample. CT and MR slices of the same object were taken for tests. Such images are inherently multimodal so difficult to correct intensity. The area of image was divided into 2 sets of pixels and for each set only one intensity mapping was used. The first set was obtained by applying least trimmed squares (Section 3.3, [38]). Apart from rejecting outliers, pixel sets were not allowed to be scattered. After each iteration of rejecting outliers, morphological closing was applied. Morphological closing forced pixel sets to form geometrically connected regions. Promising result of the experiment is shown in Figure 5.1.

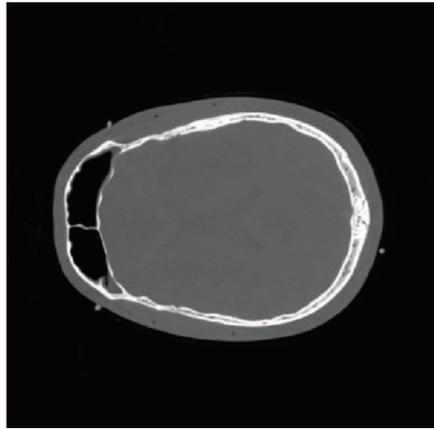
As a next step homogeneous mapping which is assigned to segments might be replaced by smooth inhomogeneous mapping. As a result the whole mapping would be smooth with discontinuities along borders of segments.



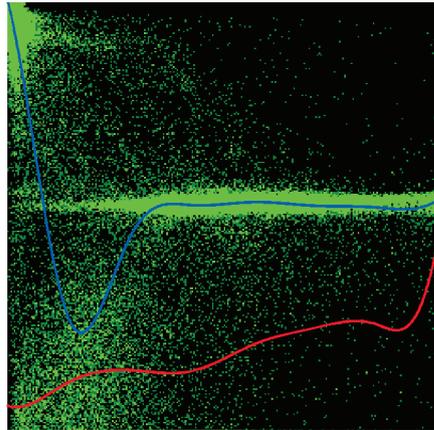
(a) u - MR image



(b) v - CT image



(c) $f_{\epsilon(u,v)}(u)$



(d) f_1, f_2

Figure 5.1: MR and CT images of the same object with segmentation based intensity correction. Pixel $u(x, y)$ is mapped by one function $f_1(u(x, y))$ or $f_2(u(x, y))$. Each mapping forms a pair $(u(x, y), f_\epsilon(u(x, y)))$ where $\epsilon = 1, 2$. The function f_ϵ for which the pair $(u(x, y), f_\epsilon(u(x, y)))$ is closer to the pair $(u(x, y), v(x, y))$ is used to map intensity at position (x, y) . MR image after correction shown in (c) is much more similar to CT image shown in (b) than MR without correction shown in (a).

Bibliography

- [1] J.R. Sykes, A. Amer, J. Czajka, C.J. Moore, *A feasibility study for image guided radiotherapy using low dose, high speed, cone beam X-ray volumetric imaging*, Radiotherapy and Oncology 77, 2005
- [2] T.E. Marchant, C.J. Moore, C.G. Rowbottom, R.I. MacKay, P.C. Williams, *Shading correction algorithm for improvement of cone-beam CT images in radiotherapy*, Physics in Medicine and Biology 53, 2008
- [3] I.N. Bankman, *Handbook of Medical Imaging, Processing and Analysis*, Academic Press, 2000
- [4] J. Modersitzki, *Numerical Methods for Image Registration*, Oxford University Press 2004
- [5] B. Zitova, J. Flusser, *Image registration methods: a survey*, Image and Vision Computing 21, 2003
- [6] J.P. Lewis, *Fast Normalized Cross-Correlation*, Vision Interface, 1995
- [7] J.P.W. Pluim, J. B.A. Maintz and M.A. Viergever, *Mutual information based registration of medical images: a survey*, IEEE Transactions on Medical Imaging, 2003
- [8] A. Collignon, F. Maes, D. Delaere, D. Vandermeulen, P. Suetens, G. Marchal, *Automated Multi-Modality Image Registration Based on Information Theory*, Information Processing in Medical Imaging, Y.J.C. Bizais, Ed., 1995
- [9] P. Viola, W.M. Wells III, *Alignment by Maximization of Mutual Information*, Proc. International Conference on Computer Vision, Boston, MA, USA, June, 1995
- [10] C. Studholme, D.L.G. Hill, and D.J. Hawkes, *An overlap invariant entropy measure of 3D medical image alignment*, Pattern Recognition, vol. 32, no. 1, pp. 7186, 1999

- [11] A. Roche, G. Malandain, N. Ayache, *The Correlation Ratio as a New Similarity Measure for Multimodal Image Registration*, LNCS vol. 1496, pp1115-1124, 1998
- [12] A. Roche, G. Malandain, X. Pennac, N. Ayache, *Multimodal image registration by maximization of the correlation ratio*, Technical Report 3378, INRIA, March 1998
- [13] R. Hartley, A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2003
- [14] P. Thevenaz, E. Ruttimann, M. Unser, *A Pyramid Approach to Sub-Pixel Registration Based on Intensity*, IEEE Transactions on Image Processing, vol. 7, no. 1, January 1998
- [15] P. Thevenaz, T. Blu, and M. Unser, *Interpolation Revisited*, IEEE Transaction on Medical Imaging, Vol. 19, No. 7, 2000
- [16] M. Unser, P. Thevenaz, *Stochastic Sampling for Computing the Mutual Information of Two Images*, 2003
- [17] S. Lee, G. Wolberg, S.Y. Shin, *Scattered Data Interpolation with Multilevel B-Splines*, IEEE Transactions on Visualization and Computer Graphics, Vol. 3, No. 3, 1997.
- [18] W. Tiller, L. Piegl, *The Nurbs Book*, Springer-Verlag, 1996.
- [19] M. Holden, *A Review of Geometric Transformations for Nonrigid Body Registration*, IEEE Transaction on Medical Imaging, Vol. 27, No. 1, 2008
- [20] D. Rueckert, L.I. Sonoda, C. Hayes, D.L.G. Hill, M.O. Leach, D.J. Hawkes, *Nonrigid Registration Using Free-Form Deformations: Application to Breast MR Images*, IEEE Transactions on Medical Imaging, Vol. 18, No. 8, August 1999
- [21] J.A. Schnaubel, D. Rueckert, M. Quist, J.M. Blackall, A.D. Castellano-Smith, T. Hartkens, G.P. Penney, W.A. Hall, H. Liu, C.L. Truwit, F.A. Gerritsen, D.L.G. Hill D.J. Hawkes, *A Generic Framework for Non-rigid Registration Based on Non-uniform Multi-level Free-Form Deformations*, LNCS2208, pp.573-581,2001
- [22] J. Kybic, M. Unser, *Fast Parametric Elastic Image Registration*, IEEE Transactions on Image Processing, Vol. 12, No. 11, 2003
- [23] G.E. Christensen, H.J. Johnson, *Consistent Image Registration*, IEEE Transactions on Medical Imaging, Vol. 20, No. 7, 2001

- [24] B. Horn and B. Schunck. *Determining optical flow*, Artificial Intelligence, 17:185-203, 1981
- [25] J. Weickert, B.M. Haar Romeny, M. Viergever, *Efficient and reliable schemes for nonlinear diffusion filtering*, IEEE Transactions on Image Processing, Vol. 7, No. 3, 1998
- [26] J. Weickert, C. Schnorr, *A Theoretical Framework for Convex Regularizers in PDE-Based Computation of Image Motion* Source International Journal of Computer Vision archive Vol. 45, 2001
- [27] T. Brox, A. Bruhn, N. Papenberg, J. Weickert, *High Accuracy Optical Flow Estimation Based on a Theory for Warping*, In Proc. 8th European Conference on Computer Vision, Springer LNCS 3024, T. Pajdla and J. Matas (Eds.), vol. 4, pp. 25-36, May 2004
- [28] G.D. Smith, *Numerical Solution of Partial Differential Equations: Finite Difference Method*, Oxford University Press, 1985
- [29] J. Nocedal, S.J. Wright, *Numerical Optimization*, Springer-Verlag, 1999
- [30] K. Madsen, H.B. Nielsen, O. Tingleff, *Methods for non-linear least squares problem*, Informatics and Mathematical Modelling, Technical University of Denmark, 2nd Edition, April 2004
- [31] S. Kabus, T. Netsch, B. Fischer, J. Modersitzki, *B-spline registration of 3D images with Levenberg-Marquardt optimization*, In J.M. Fitzpatrick and M. Sonka, editors, Medical Imaging: Image Processing, volume 5370, pages 304-313. SPIE, 2004
- [32] U. Vovk, F. Pernus, B. Likar, *A Review of Methods for Correction of Intensity Inhomogeneity in MRI*, IEEE Transactions on Medical Imaging, Vol. 26, No. 3, 2007
- [33] J.-K. Shen, B.J. Matuszewski, L.-K. Shark, *Deformable Image Registration*, Proceedings of International Conference on Image Processing, Vol. 3, 2005
- [34] S. Periaswamy, H. Farid, *Medical Image Registration with Partial Data*, Medical Image Analysis, Vol. 10, 2006
- [35] A. Myronenko, X. Song, *Image Registration by Minimization of Residual Complexity*, 2009.
- [36] M.C. Bishop, *Pattern Recognition and Mashine Learning*, Springer, 2006
- [37] R.C. Gonzalez, R.E. Woods, *Digital Image Processing*, Prentice Hall, 2007

- [38] A. Guimond, A. Roche, N. Ayache, J. Meunier, *Three-dimensional multimodal brain warping using the Demons algorithm and adaptive intensity corrections*, IEEE Transaction on Medical Imaging, vol.20, issue 1, 2001
- [39] P.J. Rousseeuw, A.M. Leroy, *Robust regression and outlier detection*, Wiley series in probability and mathematical statistics, John Wiley & Sons, 1987
- [40] T.M. Cover, J.A. Thomas, *Elements of information theory*, Wiley-Interscience, 2006
- [41] P. Grunwald, *The minimum description length principle*, MIT Press, 2007
- [42] J. Rissanen, *Modeling by shortest data description*, Automatica Vol. 14 No. 15, 1978
- [43] Y. Boykov M-P. Jolly *Interactive Graph Cuts for Optimal Boundary and Region Segmentation of Objects in N-D Images*, Proceedings of International Conference on Computer Vision, Vancouver, Canada, vol.I, p.105, 2001
- [44] R. Zabih, V. Kolmogorov, *Spatially Coherent Clustering Using Graph Cuts*, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Washington, DC, June 2004
- [45] C. Rother, V. Kolmogorov, *'GrabCut' Interactive Foreground Extraction using Iterated Graph Cuts*, Proc. ACM Siggraph, 2004
- [46] S. Osher, R. Fedkiw, *Level set methods and dynamic implicit surfaces*, Springer-Verlag, 2003