Fast optimization methods for image registration in adaptive radiation therapy

Yuchuan Qiao

Colophon

About the covers:

This is a pathway to find the summit (optimal value).

Fast optimization methods for image registration in adaptive radiation therapy
Yuchuan Qiao

ISBN: 978-94-6299-732-5
Thesis layout & cover designs by Yuchuan Qiao
Printed by Ridderprint BV

© 2017 Yuchuan Qiao, Leiden, the Netherlands
All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without permission in writing from the copyright owner.

Fast optimization methods for image registration in adaptive radiation therapy

Proefschrift

ter verkrijging van
de graad van Doctor aan de Universiteit Leiden,
op gezag van Rector Magnificus prof.mr. C.J.J.M. Stolker,
volgens besluit van het College voor Promoties
te verdedigen op woensdag, 1 november, 2017
klokke 10:00 uur

door

Yuchuan Qiao
geboren te Hubei, China

Promotor: Prof. dr. ir. B. P. F. Lelieveldt

Co-promotor: Dr. ir. M. Staring

Leden promotiecommissie: Prof. dr. Andrew Webb
Dr M. Hoogeman
Erasmus MC, Rotterdam
Prof J.P.W. Pluim
Technische Universiteit Eindhoven, Eindhoven

Advanced School for Computing and Imaging

This work was carried out in the ASCI graduate school.
ASCI dissertation series number: 377

Financial support for the publication of this thesis was kindly provided by:

- China Scholorship Council (CSC),
- Bontius Stichting,
- Library of the University of Leiden.

Contents

Contents **i**

1 Introduction **1**
- 1.1 Medical image registration . . . 1
- 1.2 The image registration framework and acceleration approaches . . . 3
- 1.3 Outline of the thesis . . . 5

2 Fast Automatic Step Size Estimation for Gradient Descent Optimization of Image Registration **7**
- 2.1 Introduction . . . 9
- 2.2 Method . . . 11
  - 2.2.1 Maximum voxel displacement . . . 12
  - 2.2.2 Maximum step size for deterministic gradient descent . . . 13
  - 2.2.3 Noise compensation for stochastic gradient descent . . . 13
  - 2.2.4 Summary and implementation details . . . 13
  - 2.2.5 Performance of proposed method . . . 14
- 2.3 Data sets . . . 15
  - 2.3.1 RIRE brain data – multi-modality rigid registration . . . 15
  - 2.3.2 SPREAD lung data – intra-subject nonrigid registration . . . 15
  - 2.3.3 Hammers brain data – inter-subject nonrigid registration . . . 17
  - 2.3.4 Ultrasound data – 4D nonrigid registration . . . 17
- 2.4 Experiment setup . . . 17
  - 2.4.1 Experimental setup . . . 18
  - 2.4.2 Evaluation measures . . . 19
- 2.5 Results . . . 20
  - 2.5.1 Accuracy results . . . 20
  - 2.5.2 Runtime results . . . 23
  - 2.5.3 Convergence . . . 25
- 2.6 Discussion . . . 26
- 2.7 Conclusion . . . 29

3 A Stochastic Quasi-Newton Method for Non-rigid Image Registration **33**
- 3.1 Introduction . . . 35
- 3.2 Methods . . . 35
  - 3.2.1 Deterministic quasi-Newton . . . 36
  - 3.2.2 Stochastic quasi-Newton . . . 36

3.3 Experiment . . . 37
3.4 Results . . . 38
3.5 Conclusion . . . 40

4 An efficient preconditioner for stochastic gradient descent optimization of image registration 43
4.1 Introduction . . . 45
4.2 Background . . . 46
  4.2.1 Preconditioned stochastic gradient descent . . . 46
  4.2.2 Related work . . . 47
4.3 Method . . . 47
  4.3.1 Preliminaries . . . 47
  4.3.2 Diagonal preconditioner estimation . . . 48
  4.3.3 Regularization . . . 49
  4.3.4 Condition number . . . 50
4.4 Data sets . . . 51
  4.4.1 Mono-modal lung data: SPREAD . . . 51
  4.4.2 Multi-modal brain data: RIRE and BrainWeb . . . 51
4.5 Experiments . . . 53
  4.5.1 Experimental setup . . . 53
  4.5.2 Convergence and runtime performance . . . 54
  4.5.3 Mono-modal image registration: SPREAD . . . 54
  4.5.4 Multi-modal image registration: RIRE and BrainWeb . . . 54
4.6 Results . . . 55
  4.6.1 Parameter sensitivity analysis . . . 55
  4.6.2 Results of mono-modal image registration . . . 60
  4.6.3 Results of multi-modal image registration . . . 62
4.7 Discussion . . . 65
4.8 Conclusion . . . 67

5 Evaluation of an open source registration package for automatic contour propagation in online adaptive intensity-modulated proton therapy of prostate cancer 69
5.1 Introduction . . . 71
5.2 Materials and Methods . . . 72
  5.2.1 Patients and imaging . . . 72
  5.2.2 Image registration . . . 72
  5.2.3 Evaluation measures . . . 73
5.3 Results . . . 74
  5.3.1 Image registration performance . . . 74
  5.3.2 Dosimetric validation . . . 75
5.4 Discussion . . . 75
5.5 Conclusion . . . 80

6 Discussion and conclusion 81
6.1 Summary . . . 81
6.2 Discussion . . . 83

6.3 Conclusion 85

Samenvatting **87**

Bibliography **93**

Appendix **103**

Publications **105**

Acknowledgements **107**

Curriculum Vitae **109**

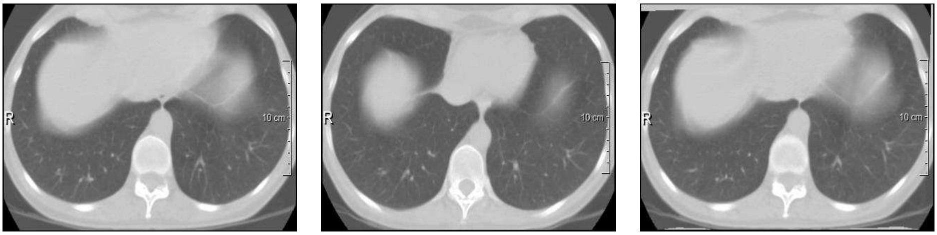
1

Introduction

1.1 Medical image registration

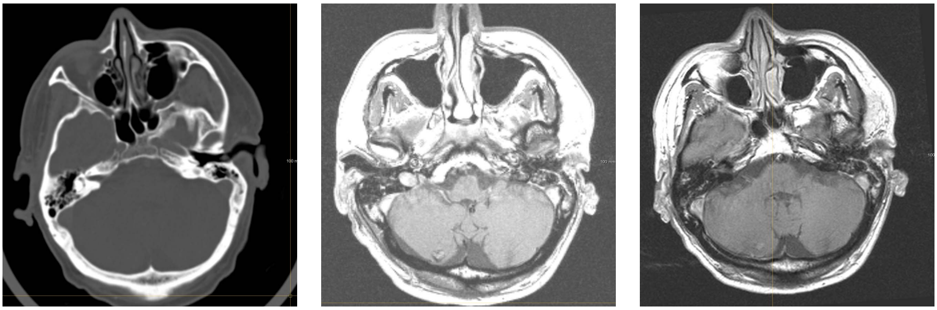
Medical imaging has become an indispensable tool in health care for diagnosis, treatment planning and therapy monitoring. In many cases, medical images are acquired at different stages of the diagnosis and treatment chain. However, medical imaging data is often very heterogenous, in that it can be acquired at different time points (to monitor disease course), or at different imaging devices (providing complementary information). In many cases, the anatomical structures in the images may move or deform due to internal movement (e.g. breathing, bladder filling or cardiac motion) or external function differences between imaging modalities. Also, in studies across multiple subjects, the anatomical structures of different subjects may also differ a lot due to inter-individual differences. The main goal of medical image registration is to find the spatial connection between heterogeneous images or populations.

With the increasing use of medical imaging in routine clinical care, medical image registration is an important driver for the development of innovative image analysis technologies. Application examples are CT screening for lung cancer, atlas-based segmentation and image-guided interventions [1, 2, 3]. For instance, in CT screening for lung cancer, follow-up CT scans of the same subject are compared against a



(a) Fixed image (b) Moving image (c) Registered moving image

Figure 1.1: Example of deformable image registration on lung CT images.



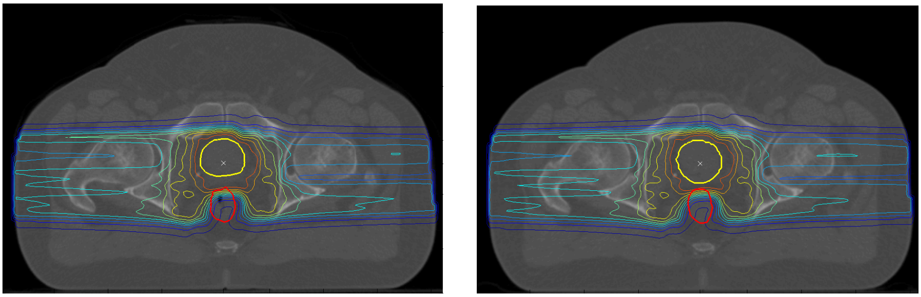
(a) CT image (b) MRI image (c) Registered image

Figure 1.2: Example of CT-MRI registration for target-volume delineation of brain tumors.

baseline CT scan, and a comparison is performed to assess the tumor changes. Even though lung CT scans are acquired at more or less standardized respiration stages, the deformations of the lung can be large. It is essential to register the CT scans to investigate the tumor development in the lung with respect to normal tissues. Figure 1.1 shows an example of deformable image registration to register a follow-up lung CT scan to a baseline CT scan. Besides mono-modal image registration, multi-modal image registration is also used frequently. For example, it can be used to delineate the target volume of brain tumors for the same patient. An example is shown in Figure 1.2. As CT imaging and MR imaging have different resolution and different tissue contrast properties, image registration could integrate these sources of information and provide a better observation of the tumor size change.

In image-guided interventions, for instance image-guided radiation therapy, a planning CT scan is acquired, based on which a treatment plan is generated. The total dose in the treatment plan is usually delivered in daily fractions. The treatment, in particular proton therapy, is sensitive to daily changes in patient setup, the location and shape of the tumor and target volume, and changes in tissue density along the proton beam path. These changes can be captured with the acquisition of a daily CT scan as shown in Figure 1.3. The induced uncertainties by these changes could dramatically distort the dose distribution compared to the planned dose distribution [4, 5, 6, 7, 8]. To achieve highest possible accuracy, the planned dose distribution need therefore be adjusted for the deformations of the tumors over the course of the treatment, which can be computed by image registration.

The procedure of online adaptive image-guided radiation therapy requires a fast, online image registration to automatically and efficiently re-contour the target and organs-at-risk (OARs) of repeat CT scans by establishing the spatial correspondence with the planning CT scan. Image registration then enables the use of small margins and high robustness without losing dose coverage. It is of high practical importance that image registration can be performed on the fly, so that treatment adaptation can be applied before new intra-fraction motions occur in the patient [9]. Nowadays, the registration computing time is quite long (usually several minutes), and it is difficult for image registration to obtain the optimal solution within a few seconds due to the



(a) Planning CT image

(b) Daily CT image

Figure 1.3: Example of organ motion in the planned dose distribution for IMPT of prostate cancer. The prostate and rectum are delineated and represented as a yellow and red solid line, respectively. The shape change of the prostate can be observed.

complicated cost functions, transformation models and optimization methods.

It would therefore be highly desirable to accelerate the procedure of image registration, to enable its use in real-time interventions.

1.2 The image registration framework and acceleration approaches

Many approaches can be applied for image registration, such as feature-based image registration and intensity-based image registration. As intensity-based image registration is widely used and most algorithms are developed based on it, we focus on this type of problem in this thesis.

The procedure of intensity-based image registration can be formulated as a parametric optimization problem to minimize the dissimilarity between a d-dimensional fixed image I_F and moving image I_M:

$$\widehat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} \mathscr{C}(I_F, I_M \circ \boldsymbol{T}_{\boldsymbol{\mu}}), \tag{1.1}$$

in which $\boldsymbol{T}_{\boldsymbol{\mu}}(\boldsymbol{x})$ is a coordinate transformation parameterized by $\boldsymbol{\mu}$. Often used dissimilarity measures $\mathscr{C}$ for intensity-based image registration include mutual information (MI), normalized correlation (NC) and the mean squared intensity difference (MSD) [1, 2, 3]. To account for rotations, translation, global scaling, shrinking and local deformations that occur in medical images, different transformation models are adopted including the translation transform, affine transform and B-spline transform. In particular, complex local deformation models require more degrees of freedom for the transformation models, and are thus more computationally expensive. Multi-resolution strategies on both the image data and the transformation model, allow for a fast and robust image registration [10].

To solve this registration optimization problem, the following iterative scheme is commonly used:

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k - \gamma_k \boldsymbol{d}_k, \tag{1.2}$$

where k is the iteration number, γ_k is the step size at iteration k, and $\boldsymbol{d}_k$ is the search direction in the parameter space. For fast registration methods the search direction $\boldsymbol{d}_k$ as well as the estimation of the step size γ_k need to be performed with high efficiency.

Gradient descent directions are widely used for the search direction $\boldsymbol{d}_k$. Gradient-type search directions include steep gradient descent, conjugate gradient descent, Newton gradient descent and their ***stochastic*** variations. Because of the exponential growth of data and parameter spaces in the past twenty years, the computational burden eventually became a bottleneck to find the optimal solution. Stochastic variations of these methods are therefore commonly used with its commendable properties: efficient implementation, little computational burden per iteration and overall less computation cost. These type of methods approximate the deterministic gradient by subsampling the fixed image. However, the inherent drawbacks of stochastic gradient methods are its slow convergence rate and unstable oscillations even if sufficient iterations are provided.

To improve the convergence rate, there are two common approaches. One can use the second order gradient to capture the curvature information of the cost function. A different option is the use of a preconditioning scheme to transform an ill-conditioned cost function to a well-conditioned one at the very beginning of the optimization. Both are well-known for deterministic gradient type methods. For ***stochastic*** type gradient methods, the noise in the curvature calculation or the preconditioner estimation may amplify the errors, resulting in a slow convergence rate or a failed registration. Besides this, the calculation of the Hessian or the preconditioner should also be fast, otherwise the gain in the convergence will be lost. New schemes of fast calculation of the Hessian or the preconditioner for ***stochastic*** type gradient methods are therefore needed.

Besides the acceleration schemes in the calculation of the search direction $\boldsymbol{d}_k$, the selection of the step size γ_k is also important. There are two classes of methods to determine the step size γ_k: exact and inexact methods. An exact way could be the conjugate gradient method to determine the step size. An example of an inexact approximation uses for instance a line search method to find the step size that satisfies the Wolfe conditions [11]. However, both schemes are developed for deterministic optimization methods and could not guarantee the convergence of stochastic type methods. For stochastic methods such as stochastic gradient descent, the step size is also an important condition to ensure the convergence, which should meet the following constraints [12, 13, 14, 15],

$$\sum_{k=1}^{\infty} \gamma_k = \infty, \qquad \sum_{k=1}^{\infty} \gamma_k^2 < \infty. \tag{1.3}$$

A common choice for the step size that satisfies this constraint is a monotonically non-increasing sequence. Consider the following decay function for the stochastic methods:

$$\gamma_k = \frac{a}{(A+k)^{\alpha}}, \tag{1.4}$$

with $a > 0,\ A \geq 1,\ 0 < \alpha \leq 1$, where $\alpha = 1$ gives a theoretically optimal rate of convergence [16].

As we can see, in Equation (1.4) the selection of a is important. In medical image registration, this selection is case-specific for different transformation models

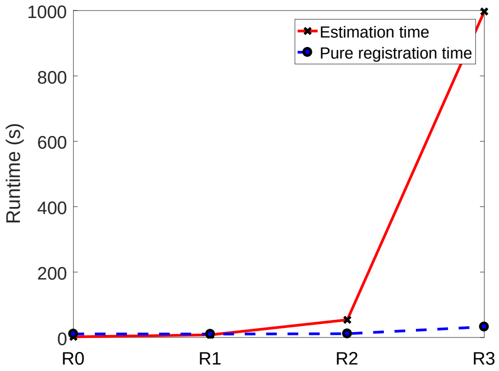


Figure 1.4: An example of runtime in seconds of ASGD for the mutual information measure and a B-spline transformation model. The blue line is the pure registration time and the red line the estimation time of the step size. R0 until R3 are the four resolution levels. The number of transformation parameters for these four resolutions is around 10^3, 10^4, 10^5 and 10^6, respectively. It can be seen that the estimation time for the step size becomes even larger than the registration time.

and different similarity measures. To select a good step size, the magnitude of a should be not too large, otherwise the estimated optimal value of cost function will be "bouncing", and not too small, otherwise the convergence will be slow [17, 18]. Choosing a suitable step size therefore is difficult to perform manually. Klein *et al.* [15] proposed a method to automatically estimate the step size for adaptive stochastic gradient descent (ASGD) by considering the distribution of transformations. This method works for few parameters within a reasonable time, but for a large number of transformation parameters, i.e. in the order of 10^5 or higher, the runtime is unacceptable and the time used in estimating the step size will dominate the optimization procedure. An example to illustrate this limitation is given in Figure 1.4. This limitation disqualifies ASGD for real-time image registration tasks. A fast alternative is therefore needed for real-time registration problems.

1.3 Outline of the thesis

The aim of this thesis is to develop novel optimization strategies for fast image registration. In particular, we address the following specific aims: 1) to investigate strategies to determine the step-size and search direction to accelerate image registration; 2) to develop new stochastic schemes for second order gradient optimization methods; 3) to investigate a new time-efficient preconditioner for preconditioned gradient descent optimization; 4) to validate these novel fast image registration techniques in the context of online adaptive image-guided radiation therapy. The thesis is further structured as follows:

Chapter 2 The Adaptive Stochastic Gradient Descent (ASGD) method has been proposed to automatically choose the optimization step size, but it comes at a high computational cost, depending on the number of transformation parameters. In Chapter 2, we propose a new computationally efficient method (fast

ASGD) to automatically determine the step size for gradient descent methods, by considering the observed distribution of the voxel displacements between iterations. A relation between the step size and the expectation and variance of the observed distribution is derived. While ASGD has quadratic complexity with respect to the transformation parameters, the fast ASGD method only has linear complexity. Extensive validation has been performed on different datasets with different modalities, inter/intra subjects, different similarity measures and transformation models. To perform a large scale experiment on 3D MR brain data, we have developed efficient and reusable tools to exploit an international high performance computing facility. This method is already integrated in an open source deformable image registration package `elastix`.

Chapter 3 ASGD not only outperforms deterministic gradient descent methods but also quasi-Newton method in terms of runtime. ASGD, however, only exploits first-order information of the cost function. In this chapter, we explore a stochastic quasi-Newton method (s-LBFGS) for non-rigid image registration. It uses the classical limited memory BFGS method in combination with noisy estimates of the gradient. Curvature information of the cost function is estimated once every L iterations and then used for the next L iterations in combination with a stochastic gradient. The method is validated on follow-up data of 3D chest CT scans (19 patients), using a B-spline transformation model and a mutual information metric.

Chapter 4 In case of ill-conditioned problems, ASGD only exhibits sublinear convergence properties. In Chapter 4, we propose an efficient preconditioner estimation method to improve the convergence rate of ASGD. Based on the observed distribution of voxel displacements in the registration, we estimate the diagonal entries of a preconditioning matrix, thus rescaling the optimization cost function. This makes the preconditioner suitable for stochastic as well as for deterministic optimization. It is efficient to compute and can be used for mono-modal as well as multi-modal cost functions, in combination with different transformation models like the rigid, affine and B-spline models.

Chapter 5 In Chapter 5, we have investigated the performance of the method developed in Chapter 2, for fast and robust contour propagation in the context of online-adaptive IMPT for prostate cancer. The planning CT scan and 7-10 repeat CT scans of 18 prostate cancer patients were used in this study. Automatic contour propagation of repeat CT scans was performed and compared with manual delineations in terms of geometric accuracy and runtime. Dosimetric accuracy was quantified by generating IMPT plans using the propagated contours expanded with a 2-mm (prostate) and 3.5-mm margin (seminal vesicles and lymph nodes) and calculating coverage based on the manual delineation. A coverage of $V_{95\%} \geq 98\%$ was considered clinically acceptable.

Chapter 6 In Chapter 6, the overall achievements of this thesis are summarized and discussed.

2

Fast Automatic Step Size Estimation for Gradient Descent Optimization of Image Registration

This chapter was adapted from:

Y. Qiao, B. van Lew, B.P.F. Lelieveldt and M. Staring. **Fast Automatic Step Size Estimation for Gradient Descent Optimization of Image Registration**, *IEEE Transactions on Medical Imaging*, Volume 35, Issue 2, Pages 539–549, 2016.

Abstract

Fast automatic image registration is an important prerequisite for image guided clinical procedures. However, due to the large number of voxels in an image and the complexity of registration algorithms, this process is often very slow. Among many classical optimization strategies, stochastic gradient descent is a powerful method to iteratively solve the registration problem. This procedure relies on a proper selection of the optimization step size, which is important for the optimization procedure to converge. This step size selection is difficult to perform manually, since it depends on the input data, similarity measure and transformation model. The Adaptive Stochastic Gradient Descent (ASGD) method has been proposed to automatically choose the step size, but it comes at a high computational cost, dependent on the number of transformation parameters.

In this chapter, we propose a new computationally efficient method (fast ASGD) to automatically determine the step size for gradient descent methods, by considering the observed distribution of the voxel displacements between iterations. A relation between the step size and the expectation and variance of the observed distribution is derived. While ASGD has quadratic complexity with respect to the transformation parameters, the fast ASGD method only has linear complexity. Extensive validation has been performed on different datasets with different modalities, inter/intra subjects, different similarity measures and transformation models. To perform a large scale experiment on 3D MR brain data, we have developed efficient and reusable tools to exploit an international high performance computing facility. For all experiments, we obtained similar accuracy as ASGD. Moreover, the estimation time of the fast ASGD method is reduced to a very small value, from 40 seconds to less than 1 second when the number of parameters is 10^5, almost 40 times faster. Depending on the registration settings, the total registration time is reduced by a factor of 2.5-7x for the experiments in this chapter.

2.1 Introduction

Image registration aims to align two or more images and is an important technique in the field of medical image analysis. It has been used in clinical procedures including radiotherapy and image-guide surgery, and other general image analysis tasks, such as automatic segmentation [19, 2, 3, 20]. However, due to the large number of image voxels, the large amount of transformation parameters and general algorithm complexity, this process is often very slow [13]. This renders the technique impractical in time-critical clinical situations, such as intra-operative procedures.

To accelerate image registration, multiple methods have been developed targeting the transformation model, the interpolation scheme or the optimizer. Several studies investigate the use of state-of-the-art processing techniques exploiting multi-threading on the CPU or also the GPU [21, 22]. Others focus on the optimization scheme that is used for solving image registration problems [23, 24, 25]. Methods include gradient descent [26, 27], Levenberg-Marquardt [28, 29], quasi-Newton [30, 31], conjugate gradient descent [25], evolution strategies [32], particle swarm methods [33, 34], and stochastic gradient descent methods [35, 15]. Among these schemes, the stochastic gradient descent method is a powerful method for large scale optimization problems and has a superb performance in terms of computation time, with similar accuracy as deterministic first order methods [25]. Deterministic second order methods gave slightly better accuracy in that study, but at heavily increased computational cost. It may therefore be considered for cases where a high level of accuracy is required, in a setting where real-time performance is not needed.

In this study, we build on the stochastic gradient descent technique to solve the optimization problem of image registration [27]:

$$\widehat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} \mathscr{C}(I_F, I_M \circ \boldsymbol{T}_{\boldsymbol{\mu}}), \tag{2.1}$$

in which $I_F(\boldsymbol{x})$ is the d-dimensional fixed image, $I_M(\boldsymbol{x})$ is the d-dimensional moving image, $\boldsymbol{T}(\boldsymbol{x}, \boldsymbol{\mu})$ is a parameterized coordinate transformation, and $\mathscr{C}$ the cost function to measure the dissimilarity between the fixed and moving image. To solve this problem, the stochastic gradient descent method adopts iterative updates to obtain the optimal parameters using the following form:

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k - \gamma_k \tilde{\boldsymbol{g}}_k, \tag{2.2}$$

where k is the iteration number, γ_k the step size at iteration k, $\tilde{\boldsymbol{g}}_k = \boldsymbol{g}_k + \boldsymbol{\epsilon}_k$ the stochastic gradient of the cost function, with the true gradient $\boldsymbol{g}_k = \partial\mathscr{C}/\partial\boldsymbol{\mu}_k$ and the approximation error $\boldsymbol{\epsilon}_k$. The stochastic gradient can be efficiently calculated using a subset of voxels from the fixed image [15] or using simultaneous perturbation approximation [36]. As shown previously [25], stochastic gradient descent has superior performance in terms of computation time compared to deterministic gradient descent and deterministic second order methods such as quasi-Newton, although the latter frequently obtains somewhat lower objective values. Similar to second order methods, stochastic gradient descent is less prone to get stuck in small local minima compared to deterministic gradient descent [37, 38]. Almost-sure convergence of the stochastic gradient descent method is guaranteed (meaning that it will converge to the local minimum "with probability 1"), provided that the step size sequence is a non-increasing and non-zero sequence with $\sum_{k=1}^{\infty} \gamma_k = \infty$ and $\sum_{k=1}^{\infty} \gamma_k^2 < \infty$ [12]. A suitable step size sequence is very important, because a poorly chosen step size will cause problems of estimated value "bouncing" if this step size is too large, or slow convergence if it is too small [17, 18]. Therefore, an exact and automatically estimated step size, independent of problem settings, is essential for the gradient-based optimization of image registration. Note that for deterministic quasi-Newton methods the step size is commonly chosen using an (in)exact line search.

Methods that aim to solve the problem of step size estimation can be categorized in three groups: manual, semi-automatic, and automatic methods. In 1952, Robbins and Monro [12] proposed to manually select a suitable step size sequence. Several methods were proposed afterwards to improve the convergence of the Robbins-Monro method, which focused on the construction of the step size sequence, but still required manual selection of the initial step size. Examples include Kesten's rule [39], Gaivoronski's rule [40], and the adaptive two-point step size gradient method [41]. An overview of these methods can be found here [42, 43]. These manual selection methods, however, are difficult to use in the practice, because different applications require different settings. Especially for image registration, different fixed or moving images, different similarity measures or transformation models require a different step size. For example, it has been reported that the step size can differ several orders of magnitude between cost functions [15]. Moreover, manual selection is time-consuming.

Spall [36] used a step size following a rule-of-thumb that the step size times the magnitude of the gradient is approximately equal to the smallest desired change of $\boldsymbol{\mu}$ in the early iterations. The estimation is based on a preliminary registration, after which the step size is manually estimated and used in subsequent registrations. This manual procedure is not adaptive to the specific images, depends on the parameterization $\boldsymbol{\mu}$, and requires setting an nonintuitive 'desired change' in $\boldsymbol{\mu}$.

For the semi-automatic selection, Suri [17] and Brennan [18] proposed to use a step size with the same scale as the magnitude of $\boldsymbol{\mu}$ observed in the first few iterations of a preliminary simulation experiment, in which a latent difference of the step size between the preliminary experiment and the current one is inevitable. Bhagalia also used a training method to estimate the step size of stochastic gradient descent optimization for image registration [44]. First, a pseudo ground truth was obtained using deterministic gradient descent. Then, after several attempts, the optimal step size was chosen to find the optimal warp estimates which had the smallest error values compared with the pseudo ground truth warp obtained in the first step. This method is complex and time-consuming as it requires training data, and moreover generalizes training results to new cases.

The Adaptive Stochastic Gradient Descent method (ASGD) [15] proposed by Klein *et al.* automatically estimates the step size. ASGD estimates the distribution of the gradients and the distribution of voxel displacements, and finally calculates the initial step size based on the voxel displacements. This method works for few parameters within reasonable time, but for a large number of transformation parameters, i.e. in the order of 10^5 or higher, the run time is unacceptable and the time used in estimating the step size will dominate the optimization [45]. This disqualifies ASGD for real-time image registration tasks.

In this chapter, we propose a new computationally efficient method, fast ASGD (hereafter FASGD), to automatically select the optimization step size for gradient descent optimization, by deriving a relation with the observed voxel displacement. This chapter extends a conference chapter [45] with detailed methodology and extensive validation, using many different datasets of different modality and anatomical structure. Furthermore, we have developed tools to perform extensive validation of our method by interfacing with a large international computing facility. In Section 2.2, the method to calculate the step size is introduced. The dataset description is given in Section 2.3. The experimental setup to evaluate the performance of the new method is presented in Section 2.4. In Section 2.5, the experimental results are given. Finally, Section 2.6 and 2.7 conclude the chapter.

2.2 Method

A commonly used choice for the step size estimation in gradient descent is to use a monotonically non-increasing sequence. In this chapter we use the following decaying function, which can adaptively tune the step size according to the direction and magnitude of consecutive gradients, and has been used frequently in the stochastic optimization literature [12, 16, 13, 35, 43, 40, 14, 42, 15]:

$$\gamma_k = \frac{a}{(A + t_k)^{\alpha}}, \tag{2.3}$$

with $a > 0$, $A \geq 1$, $0 < \alpha \leq 1$, where $\alpha = 1$ gives a theoretically optimal rate of convergence [16], and is used throughout this chapter. The iteration number is denoted by k, and $t_k = \max(0, t_{k-1} + f(-\tilde{\boldsymbol{g}}_{k-1}^T \tilde{\boldsymbol{g}}_{k-2}))$. The function f is a sigmoid function with $f(0) = 0$:

$$f(x) = \frac{f_{\max} - f_{\min}}{1 - (f_{\max}/f_{\min})e^{-x/\omega}} + f_{\min}, \tag{2.4}$$

in which $f_{\max}$ determines the maximum gain at each iteration, $f_{\min}$ determines the maximal step backward in time, and ω affects the shape of the sigmoid function [15]. A reasonable choice for the maximum of the sigmoid function is $f_{\max} = 1$, which implies that the maximum step forward in time equals that of the Robbins-Monro method [15]. It has been proven that convergence is guaranteed as long as $t_k \geq 0$ [14, 15]. Specifically, from Assumption A4 [14] and Assumption B5 [15], asymptotic normality and convergence can be assured when $f_{\max} > -f_{\min}$ and $\omega > 0$. In [15] (Equation (59)) $\omega = \zeta\sqrt{Var(\boldsymbol{\varepsilon}_k^T \boldsymbol{\varepsilon}_{k-1})}$ was used, which requires the estimation of the distribution of the approximation error for the gradients, which is time consuming. Moreover, a parameter ζ is introduced which was empirically set to 10%. Setting $\omega = 10^{-8}$ avoids a costly computation, and still guarantees the conditions required for convergence. For the minimum of the sigmoid function we choose $f_{\min} = -0.8$ in this chapter, fulfilling the convergence criteria.

In the step size sequence $\{\gamma_k\}$, all parameters need to be selected before the optimization procedure. The parameter α controls the decay rate; the theoretically optimal value is 1 [10, 15]. The parameter A provides a starting point, which has most influence at the beginning of the optimization. From experience [10, 15], $A = 20$ provides a reasonable value for most situations. The parameter a in the numerator determines the overall scale of the step size sequence, which is important but difficult to select, since it is dependent on I_F, I_M, $\mathscr{C}$ and $\boldsymbol{T_\mu}$. The step size can differ substantially between resolutions (Figure 4 [15]) and for different cost functions (Table 2 [15]). This means that the problem of estimating the step size sequence is mainly determined by a. In this work, we therefore focus on automatically selecting the parameter a in a less time-consuming manner.

2.2.1 Maximum voxel displacement

The intuition of the proposed step size selection method is that the voxel displacements should start with a reasonable value and gradually diminish to zero. The incremental displacement of a voxel $\boldsymbol{x}_j$ in a fixed image domain Ω_F between iteration k and $k+1$ for an iterative optimization scheme is defined as

$$\boldsymbol{d}_k(\boldsymbol{x}_j) = \boldsymbol{T}\left(\boldsymbol{x}_j, \boldsymbol{\mu}_{k+1}\right) - \boldsymbol{T}\left(\boldsymbol{x}_j, \boldsymbol{\mu}_k\right), \forall \boldsymbol{x}_j \in \Omega_F. \tag{2.5}$$

To ensure that the incremental displacement between each iteration is neither too big nor too small, we need to constrain the voxel's incremental displacement $\boldsymbol{d}_k$ into a reasonable range. We assume that the magnitude of the voxel's incremental displacement $\boldsymbol{d}_k$ follows some distribution, which has expectation $E||\boldsymbol{d}_k||$ and variance $Var||\boldsymbol{d}_k||$, in which $\|\cdot\|$ is the ℓ^2 norm. For a translation transform, the voxel displacements are all equal, so the variance is zero; for non-rigid registration, the voxel displacements vary spatially, so the variance is larger than zero. To calculate the magnitude of the incremental displacement $||\boldsymbol{d}_k||$, we use the first-order Taylor expansion to make an approximation of $\boldsymbol{d}_k$ around $\boldsymbol{\mu}_k$:

$$\boldsymbol{d}_k \approx \frac{\partial \boldsymbol{T}}{\partial \boldsymbol{\mu}}\left(\boldsymbol{x}_j, \boldsymbol{\mu}_k\right) \cdot \left(\boldsymbol{\mu}_{k+1} - \boldsymbol{\mu}_k\right) = \boldsymbol{J}_j\left(\boldsymbol{\mu}_{k+1} - \boldsymbol{\mu}_k\right), \tag{2.6}$$

in which $\boldsymbol{J}_j = \frac{\partial \boldsymbol{T}}{\partial \boldsymbol{\mu}}\left(\boldsymbol{x}_j, \boldsymbol{\mu}_k\right)$ is the Jacobian matrix of size $d \times |\boldsymbol{\mu}|$. Defining $\boldsymbol{M}_k(\boldsymbol{x}_j) = \boldsymbol{J}(\boldsymbol{x}_j)\boldsymbol{g}_k$ and combining with the update rule $\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k - \gamma_k \boldsymbol{g}_k$, $\boldsymbol{d}_k$ can be rewritten as:

$$\boldsymbol{d}_k(\boldsymbol{x}_j) \approx -\gamma_k \boldsymbol{J}(\boldsymbol{x}_j)\boldsymbol{g}_k = -\gamma_k \boldsymbol{M}_k(\boldsymbol{x}_j). \tag{2.7}$$

For a maximum allowed voxel displacement, Klein [15] introduced a user-defined parameter δ, which has a physical meaning with the same unit as the image dimensions, usually in mm. This implies that the maximum voxel displacement for each voxel between two iterations should be not larger than δ: i.e $\|\boldsymbol{d}_k(\boldsymbol{x}_j)\| \le \delta, \forall \boldsymbol{x}_j \in \Omega_F$. We can use a weakened form for this assumption:

$$P(\|\boldsymbol{d}_k(\boldsymbol{x}_j)\| > \delta) < \rho, \tag{2.8}$$

where ρ is a small probability value often 0.05. According to the Vysochanskij Petunin inequality [46], for a random variable X with unimodal distribution, mean μ and finite, non-zero variance σ^2, if $\lambda > \sqrt{(8/3)}$, the following theorem holds:

$$P(|X - \mu| \ge \lambda\sigma) \le \frac{4}{9\lambda^2}. \tag{2.9}$$

This can be rewritten as:

$$P(\mu - 2\sigma \le x \le \mu + 2\sigma) \approx 0.95. \tag{2.10}$$

Based on this boundary, we can approximate Equation (2.8) with the following expression:

$$E\left\|\boldsymbol{d}_k(\boldsymbol{x}_j)\right\| + 2\sqrt{Var\left\|\boldsymbol{d}_k(\boldsymbol{x}_j)\right\|} \le \delta. \tag{2.11}$$

This is slightly different from the squares used in Equation (42) in [15], which avoids taking square roots for performance reasons. In this chapter we are interested in the incremental displacements, not its square. Combining with Equation (2.7), we obtain the relationship between step size and maximum voxel displacement as follows:

$$\gamma_k\left(E\left\|\boldsymbol{M}_k(\boldsymbol{x}_j)\right\| + 2\sqrt{Var\left\|\boldsymbol{M}_k(\boldsymbol{x}_j)\right\|}\right) \le \delta. \tag{2.12}$$

2.2.2 Maximum step size for deterministic gradient descent

From the step size function $\gamma(k) = a/(k+A)^\alpha$, it is easy to find the maximum step size $\gamma_{\max} = \gamma(0) = a/A^\alpha$, and the maximum value of a, $a_{\max} = \gamma_{\max}A^\alpha$. This means that the largest step size is taken at the beginning of the optimization procedure for each resolution. Using Equation (2.12), we obtain the following equation of $a_{\max}$:

$$a_{\max} = \frac{\delta A^\alpha}{E\|\boldsymbol{M}_0(\boldsymbol{x}_j)\| + 2\sqrt{Var\|\boldsymbol{M}_0(\boldsymbol{x}_j)\|}}. \tag{2.13}$$

For a given δ, the value of a can be estimated from the initial distribution of $\boldsymbol{M}_0$ at the beginning of each resolution.

2.2.3 Noise compensation for stochastic gradient descent

The stochastic gradient descent method combines fast convergence with a reasonable accuracy [25]. Fast estimates of the gradient are obtained using a small subset of the fixed image voxels, randomly chosen in each iteration. This procedure introduces noise to the gradient estimate, thereby influencing the convergence rate. This in turn means that the optimal step size for *stochastic* gradient descent will be different compared to *deterministic* gradient descent. When the approximation error $\boldsymbol{\epsilon} = \boldsymbol{g} - \tilde{\boldsymbol{g}}$ increases, the search direction $\tilde{\boldsymbol{g}}$ is more unpredictable, thus a smaller and more careful step size is required. Similar to [15] we assume that $\boldsymbol{\epsilon}$ is a zero mean Gaussian variable with small variance, and we adopt the ratio between the expectation of the exact and approximated gradient to modify the step size $a_{\max}$ as follows:

$$\eta = \frac{E\|\boldsymbol{g}\|^2}{E\|\tilde{\boldsymbol{g}}\|^2} = \frac{E\|\boldsymbol{g}\|^2}{E\|\boldsymbol{g}\|^2 + E\|\boldsymbol{\epsilon}\|^2}. \tag{2.14}$$

2.2.4 Summary and implementation details

2.2.4.1 The calculation of $a_{\max}$ for exact gradient descent

The cost function used in voxel-based image registration usually takes the following form:

$$\boldsymbol{C}(\boldsymbol{\mu}) = \frac{1}{|\Omega_F|}\sum_{\boldsymbol{x}_j\in\Omega_F}\Psi\left(I_F(\boldsymbol{x}_j), I_M(\boldsymbol{T}(\boldsymbol{x}_j,\boldsymbol{\mu}))\right), \tag{2.15}$$

in which Ψ is a similarity measure, Ω_F is a discrete set of voxel coordinates from the fixed image and $|\Omega_F|$ is the cardinality of this set. The gradient $\boldsymbol{g}$ of this cost function is:

$$\boldsymbol{g} = \frac{\partial \boldsymbol{C}}{\partial \boldsymbol{\mu}} = \frac{1}{|\Omega_F|} \sum_{\boldsymbol{x}_j \in \Omega_F} \frac{\partial \boldsymbol{T}'}{\partial \boldsymbol{\mu}} \frac{\partial I_M}{\partial \boldsymbol{x}} \frac{\partial \Psi}{\partial I_M}. \tag{2.16}$$

The reliable estimate of $a_{\max}$ relies on the calculation of the exact gradient. We obtain a trade-off between the accuracy of computing $\boldsymbol{g}$ with its computation time, by randomly selecting a sufficiently large number of samples from the fixed image. Specifically, to compute (2.16) we use a subset $\Omega_F^1 \subset \Omega_F$ of size N_1 equal to the number of transformation parameters $P = |\boldsymbol{\mu}|$.

Then, $\boldsymbol{J}_j = \frac{\partial \boldsymbol{T}}{\partial \boldsymbol{\mu}} \left(\boldsymbol{x}_j, \boldsymbol{\mu}_k\right)$ is computed at each voxel coordinate $\boldsymbol{x}_j \in \Omega_F^1$. The expectation and variance of $\|\boldsymbol{M}_0(\boldsymbol{x}_j)\|$ can be calculated using the following expressions:

$$E\|\boldsymbol{M}_0(\boldsymbol{x}_j)\| = \frac{1}{N_1} \sum_{\boldsymbol{x}_j \in \Omega_F^1} \|\boldsymbol{M}_0(\boldsymbol{x}_j)\|, \tag{2.17}$$

$$Var\|\boldsymbol{M}_0(\boldsymbol{x}_j)\| = \tfrac{1}{N_1-1} \sum_{\boldsymbol{x}_j \in \Omega_F^1} \left(\|\boldsymbol{M}_0(\boldsymbol{x}_j)\| - E\|\boldsymbol{M}_0(\boldsymbol{x}_j)\|\right)^2. \tag{2.18}$$

2.2.4.2 The calculation of η

The above analysis reveals that the noise compensation factor η also influences the initial step size. This factor requires computation of the exact gradient $\boldsymbol{g}$ and the approximate gradient $\tilde{\boldsymbol{g}}$. Because the computation of the exact gradient using all voxels is too slow, uniform sampling is used, where the number of samples is determined empirically as $N_2 = \min(100000, |\Omega_F|)$. To obtain the stochastic gradient $\tilde{\boldsymbol{g}}$, we perturb $\boldsymbol{\mu}$ by adding Gaussian noise and recompute the gradient, as detailed in [15].

2.2.4.3 The final formula

The noise compensated step size is obtained using the following formula:

$$a = \eta \frac{\delta A^\alpha}{E\|\boldsymbol{M}_0(\boldsymbol{x}_j)\| + 2\sqrt{Var\|\boldsymbol{M}_0(\boldsymbol{x}_j)\|}}. \tag{2.19}$$

In summary, the gradient $\boldsymbol{g}$ is first calculated using Equation (2.16), and then the magnitude $\boldsymbol{M}_0(\boldsymbol{x}_j)$ is computed at each voxel $\boldsymbol{x}_j$, finally $a_{\max}$ is obtained. In step 2, the noise compensation η is calculated through the perturbation process. Finally, a is obtained through Equation (2.19).

2.2.5 Performance of proposed method

In this section, we compare the time complexity of the fast ASGD method with the ASGD method. Here we only give the final formula of the ASGD method, for more details see reference [15]. The ASGD method uses the following equation:

$$a_{\max} = \frac{\delta A^\alpha}{\sigma} \min_{\boldsymbol{x}_j \in \Omega_F^1} \left[Tr(\boldsymbol{J}_j \boldsymbol{C} \boldsymbol{J}_j') + 2\sqrt{2} \|\boldsymbol{J}_j \boldsymbol{C} \boldsymbol{J}_j'\|_F \right]^{-\frac{1}{2}}, \tag{2.20}$$

where σ is a scalar constant related to the distribution of the exact gradient $\boldsymbol{g}$ [15], $\boldsymbol{C} = \frac{1}{|\Omega_F^1|^2} \sum_j \boldsymbol{J}_j' \boldsymbol{J}_j$ is the covariance of the Jacobian, and $\|\cdot\|_F$ denotes the Frobenius norm.

From Equation (2.13), the time complexity of FASGD is dominated by three terms: the Jacobian $\boldsymbol{J}(\boldsymbol{x}_j)$ with size $d \times P$, the gradient $\boldsymbol{g}$ of size P, and the number of voxels N_1 from which the expectation and variance of $\boldsymbol{M}_0$ are calculated. The matrix computation $\boldsymbol{M}_0(\boldsymbol{x}_j) = \boldsymbol{J}(\boldsymbol{x}_j)\boldsymbol{g}$ requires $d \times P$ multiplications and additions for each of the N_1 voxels $\boldsymbol{x}_j$, and therefore the time complexity of the proposed method is $\mathcal{O}(dN_1P)$. The dominant terms in Equation (2.20) are the Jacobian (size $d \times P$) and its covariance matrix $\boldsymbol{C}$ (size $P \times P$). Calculating $\boldsymbol{J}_j \boldsymbol{C} \boldsymbol{J}_j'$ from right to left requires $d \times P^2$ multiplications and additions for $\boldsymbol{C}\boldsymbol{J}_j'$ and an additional $d^2 \times P$ operations for the multiplication with the left-most matrix $\boldsymbol{J}_j$. Taking into account the number of voxels N_1, the time complexity of the original ASGD method is therefore $\mathcal{O}(N_1 \times (d \times P^2 + d^2 \times P)) = \mathcal{O}(dN_1P^2)$, as $P \gg d$. This means that FASGD has a linear time complexity with respect to the dimension of $\boldsymbol{\mu}$, while ASGD is quadratic in P.

For the B-spline transformation model, the size of the non-zero part of the Jacobian is much smaller than the full Jacobian, i.e. only $d \times P_2$, where P_2 is determined by the B-spline order used in this model. For a cubic B-spline transformation model, each voxel is influenced by 4^d control points, so $P_2 = 4^2 = 16$ in 2D and $P_2 = 4^3 = 64$ in 3D. For the fast ASGD method the time complexity reduces to $\mathcal{O}(dN_1P_2)$ for the cubic B-spline model. However, as the total number of operations for the calculation of $\boldsymbol{J}_j \boldsymbol{C} \boldsymbol{J}_j'$ is still $d \times P_2 \times P$, the time complexity of ASGD is $\mathcal{O}(dN_1P_2P)$. Since $P \gg N_1 \geq P_2 > d$, the dominant term of FASGD becomes the number of samples N_1, while for ASGD it is still a potentially very large number P.

2.3 Data sets

In this section we describe the data sets that were used to evaluate the proposed method. Data sets were chosen to represent a broad category of use cases, i.e. mono-modal and multi-modal, intra-patient as well as inter-patient, from different anatomical sites, and having rigid as well as nonrigid underlying deformations. The overview of all data sets is presented in Table 2.1.

2.3.1 RIRE brain data – multi-modality rigid registration

The Retrospective Image Registration Evaluation (RIRE) project provides multi-modality brain scans with a ground truth for rigid registration evaluation [47]. These brain scans were obtained from 9 patients, where we selected CT scans and MR T1 scans. Fiducial markers were implanted in each patient, and served as a ground truth. These markers were manually erased from the images and replaced with a simulated background pattern.

In our experiments, we registered the T1 MR image (moving image) to the CT image (fixed image) using rigid registration. At the website of RIRE, eight corner points of both CT and MR T1 images are provided to evaluate the registration accuracy.

2.3.2 SPREAD lung data – intra-subject nonrigid registration

During the SPREAD study [48], 3D lung CT images of 19 patients were scanned without contrast media using a Toshiba Aquilion 4 scanner with scan parameters: 135 kVp; 20 mAs per rotation; rotation time 0.5 s; collimation: 4 × 5 mm. Images were

Table 2.1: Overview of data sets and experiments

	RIRE	SPREAD	Hammers	Abdomen
Anatomy	Brain	Lung	Brain	Abdomen
Modality	CT and 1.5T MR T1	CT	MR	Ultrasound
Dimensions	CT: $512 \times 512 \times 50$ MR: $256 \times 256 \times 50$	3D: $450 \times 300 \times 130$	3D: $180 \times 200 \times 170$	4D: $227 \times 229 \times 227 \times 96$
Voxel size (mm)	CT: $0.45 \times 0.45 \times 3$ MR: $0.85 \times 0.85 \times 3$	$\sim 0.7 \times 0.7 \times 2.5$	$0.94 \times 0.94 \times 0.94$	$0.7 \times 0.7 \times 0.7 \times 1$
Number of patients	9	21	30	3 volunteers × 3 positions
Registration	Multi-modality Intra subject	Single modality Intra subject	Single modality Inter subject	Single modality Intra subject
Similarity measure	MI	MSD, NC, MI, NMI	MI	MI
Transformation	Rigid	Affine + B-spline	Similarity + B-spline	B-spline
B-spline control point grid spacing (mm)	-	$10 \times 10 \times 10$	$5 \times 5 \times 5$	$15 \times 15 \times 15 \times 1$
Number of parameters (last resolution)	6	$\sim 90k$	$\sim 150k$	$\sim 870k$
Ground truth	8 corner points	100 corresponding points	83 labelled regions	22 landmarks
Evaluation measure	Euclidean distance	Euclidean distance	Dice overlap	Euclidean distance
Number of registrations per setting	9×3	19×3	$30 \times 29 \times 3$	9×3
Settings	1	1	252	1
Total number of registrations	27	228	657,720	27

reconstructed with a standardized protocol optimized for lung densitometry, including a soft FC12 kernel, using a slice thickness of 5 mm and an increment of 2.5 mm, with an inplane resolution of around 0.7 × 0.7 mm. The patient group, aging from 49 to 78 with 36%-87% predicted FEV_1 had moderate to severe COPD at GOLD stage II and III, without $\alpha 1$ antitrypsin deficiency.

One hundred anatomical corresponding points from each lung CT image were semi-automatically extracted as a ground truth using Murphy's method [49]. The algorithm automatically finds 100 evenly distributed points in the baseline, only at characteristic locations. Subsequently, corresponding points in the follow-up scan are predicted by the algorithm and shown in a graphical user interface for inspection and possible correction. More details can be found in [50].

2.3.3 Hammers brain data – inter-subject nonrigid registration

We use the brain data set developed by Hammers *et al.* [51], which contains MR images of 30 healthy adult subjects. The median age of all subjects was 31 years (range 20 ~ 54), and 25 of the 30 subjects were strongly right handed as determined by routine pre-scanning screening. MRI scans were obtained on a 1.5 Tesla GE Sigma Echospeed scanner. A coronal T1 weighted 3D volume was acquired using an inversion recovery prepared fast spoiled gradient recall sequence (GE), TE/TR/NEX 4.2 msec (fat and water in phase)/15.5 msec/1, time of inversion (TI) 450 msec, flip angle 20Âř, to obtain 124 slices of 1.5 mm thickness with a field of view of 18×24 cm with a 192×256 matrix [52]. This covers the whole brain with voxel sizes of $0.94 \times 0.94 \times 1.5$ mm^3. Images were resliced to create isotropic voxels of $0.94 \times 0.94 \times 0.94$ mm^3, using windowed sinc interpolation.

Each image is manually segmented into 83 regions of interest, which serve as a ground truth. All structures were delineated by one investigator on each MRI in turn before the next structure was commenced, then a separate neuroanatomically trained operator evaluated each structure to ensure that consensus was reached for the difficult cases. In our experiment, we performed inter-subject registration between all patients. Each MR image was treated as a fixed image as well as a moving image, so the total number of registrations for 30 patients was 870 for each particular parameter setting.

2.3.4 Ultrasound data – 4D nonrigid registration

We used the 4D abdominal ultrasound dataset provided by Vijayan *et al.* [53], which contains 9 scans from three healthy volunteers at three different positions and angles. Each scan was taken over several breathing cycles (12 seconds per cycle). These scans were performed on a GE Healthcare vivid E9 scanner by a skilled physician using an active matrix 4D volume phased array probe.

The ground truth is 22 well-defined anatomical landmarks, first indicated in the first time frame by the physician who acquired the data, and then manually annotated in all 96 time frames by engineers using VV software [54].

2.4 Experiment setup

In this section, the general experimental setup and the evaluation measurements are presented and more details about the experimental environment are given.

2.4.1 Experimental setup

The experiments focus on the properties of the fast ASGD method in terms of registration accuracy, registration runtime and convergence of the algorithm. We will compare the proposed method with two variants of the original ASGD method. While for FASGD $f_{\min}$ and ω are fixed, the ASGD method automatically estimates them. For a fair comparison, a variant of the ASGD method is included in the comparison, that sets these parameters to the same value as FASGD: $f_{\min} = -0.8$ and $\omega = 10^{-8}$. In summary, three methods are compared in all the experiments: the original ASGD method that automatically estimates all parameters (ASGD), the ASGD method with default settings only estimating a (ASGD′) and the fast ASGD method (FASGD). The fast ASGD method has been implemented using the C++ language in the open source image registration toolbox `elastix` [10], where the ASGD method is already integrated.

To thoroughly evaluate FASGD, a variety of imaging problems including different modalities and different similarity measures are considered in the experiments. Specifically, the experiments were performed using four different datasets, rigid and nonrigid transformation models, inter/intra subjects, four different dissimilarity measures and three imaging modalities. The experiments are grouped by the experimental aim: registration accuracy in Section 2.5.1, registration time in Section 2.5.2 and algorithm convergence in Section 2.5.3. The RIRE brain data is used for the evaluation of rigid registration. The SPREAD lung CT data is especially used to verify the performance of FASGD on four different dissimilarity measures, including the mean squared intensity difference (MSD) [2], normalized correlation (NC) [2], mutual information (MI) [27] and normalized mutual information (NMI) [55]. The Hammers brain data is intended to verify inter-subject registration performance. The ultrasound data is specific for 4-dimensional medical image registration, which is more complex. An overview of the experimental settings is given in Table 2.1.

For the evaluation of the registration accuracy, the experiments on the RIRE brain data, the SPREAD lung CT data and the ultrasound abdominal data, were performed on a local workstation with 24 GB memory, Linux Ubuntu 12.04.2 LTS 64 bit operation system and an Intel Xeon E5620 CPU with 8 cores running at 2.4 GHz. To see the influence of the parameters A and δ on the registration accuracy, we perform an extremely large scale experiment on the Hammers brain data using the Life Science Grid (`lsgrid`) [56], which is a High Performance Computing (HPC) facility. We tested all combinations of the following settings: $A \in \{1.25, 2.5, \ldots, 160, 320\}$, $\delta \in \{0.03125, 0.0625, \ldots, 128, 256\}$ (in mm) and $k \in \{250, 2000\}$. This amounts to 252 combinations of registration settings and a total of 657,720 registrations, see Table 2.1. Each registration requires about 15 minutes of computation time, which totals about 164,000 core hours of computation, i.e ~19 years, making the use of an HPC resource essential. With the `lsgrid` the run time of the Hammers experiment is reduced to 2-3 days. More details about the `lsgrid` are given in the Appendix.

For a fair comparison, all timing experiments were carried out on the local workstation. Timings are reported for all the registrations, except for the Hammers data set, where we only report timings from a subset. From Equation (2.19), we know that the runtime is independent of the parameters A and δ. Therefore, for the Hammers data, we used $A = 20$ and δ equal to the voxel size. We randomly selected

100 out of the 870 registrations, as a sufficiently accurate approximation.

The convergence of the algorithms is evaluated in terms of the step size, the Euclidean distance error and the cost function value, as a function of the iteration number.

All experiments were done using the following routine: (1) Perform a linear registration between fixed and moving image to get a coarse transformation $\boldsymbol{T}_0$, using a rigid transformation for the RIRE brain data, an affine transformation for the SPREAD lung CT data, a similarity transformation rigid plus isotropic scaling for the Hammers brain data, and no initial transformation for the 4D ultrasound data; (2) Perform a non-linear cubic B-spline based registration [57] for all datasets except the RIRE data to get the transformation $\boldsymbol{T}_1$. For the ultrasound data, the B-spline transformation model proposed by Metz *et al.* [58] is used, which registers all 3D image sequences in a group-wise strategy to find the optimal transformation that is both spatially and temporally smooth. A more detailed explanation of the registration methodology is in [53]; (3) Transform the landmarks or moving image segmentations using $\boldsymbol{T}_1 \circ \boldsymbol{T}_0$; (4) Evaluate the results using the evaluation measures defined in Section 2.4.2.

For each experiment, a three level multi-resolution strategy was used. The Gaussian smoothing filter had a standard deviation of 2, 1 and 0.5 mm for each resolution. For the B-spline transformation model, the grid size of the B-spline control point mesh is halved in each resolution to increase the transformation accuracy [57]. We used $K = 500$ iterations and 5000 samples, except for the ultrasound experiment where we used 2000 iterations and 2000 samples according to Vijayan [53]. We set $A = 20$ and δ equal to the voxel size (the mean length of the voxel edges).

2.4.2 Evaluation measures

Two evaluation measures were used to verify the registration accuracy: the Euclidean distance and the mean overlap. The Euclidean distance measure is given by:

$$\mathrm{ED} = \frac{1}{n} \sum_{i=1}^{n} \| \boldsymbol{T}(\boldsymbol{p}_F^i) - \boldsymbol{p}_M^i \|, \tag{2.21}$$

in which $\boldsymbol{p}_F^i$ and $\boldsymbol{p}_M^i$ are coordinates from the fixed and moving image, respectively. For the RIRE brain data, 8 corner points and for the SPREAD data 100 corresponding points are used to evaluate the performance. For the 4D ultrasound image, we adopt the following measure from [53]:

$$\mathrm{ED} = \left(\frac{1}{\tau - 1} \sum_t \| \boldsymbol{p}_t - \boldsymbol{T}_t(\boldsymbol{q}) \|^2 \right)^{\frac{1}{2}}, \tag{2.22}$$

in which $\boldsymbol{p}_t = 1/J \sum_j \boldsymbol{p}_{tj}$ and $\boldsymbol{p}_{tj}$ is a landmark at time t placed by observer j, $\boldsymbol{q} = 1/\tau \sum_t \boldsymbol{S}_t(\boldsymbol{p}_t)$ is the mean of landmarks after inverse transformation.

The mean overlap of two segmentations from the images is calculated by the Dice Similarity Coefficient (DSC) [13]:

$$\mathrm{DSC} = \frac{1}{R} \sum_r \frac{2|\boldsymbol{M}_r \cap \boldsymbol{F}_r|}{|\boldsymbol{M}_r| + |\boldsymbol{F}_r|}, \tag{2.23}$$

in which r is a labelled region and $R = 83$ the total number of regions for the Hammers data.

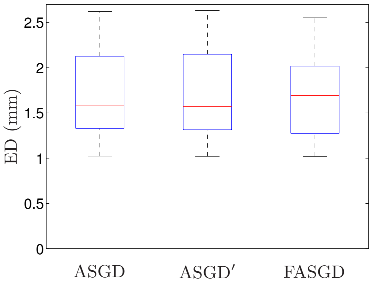


Figure 2.1: Euclidean distance error in mm for the RIRE brain data performed using MI.

To assess the registration accuracy, a Wilcoxon signed rank test ($p = 0.05$) for the registration results was performed. For the SPREAD data, we first obtained the mean distance error of 100 points for each patient and then performed the Wilcoxon signed rank test to these mean errors.

Registration smoothness is assessed for the SPREAD experiment by measuring the determinant of the spatial Jacobian of the transformation, $J = |\partial \boldsymbol{T}/\partial \boldsymbol{x}|$ [59]. Because the fluctuation of J should be relatively small for smooth transformations, we use the standard deviation of J to represent smoothness.

The computation time is determined by the number of parameters and the number of voxels sampled from the fixed image. For a small number of parameters the estimation time can be ignored, and therefore we only provide the comparison for the B-spline transformation. Both the parameter estimation time and pure registration time were measured, for each resolution.

2.5 Results

2.5.1 Accuracy results

In this section, we compare the registration accuracy between ASGD, ASGD′ and FASGD.

2.5.1.1 RIRE brain data

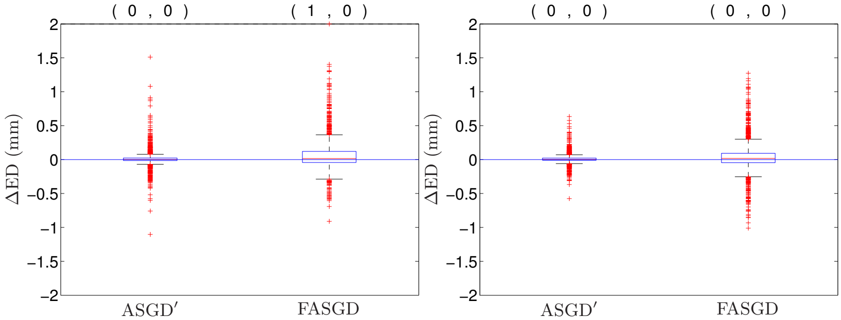
The results shown in Figure 2.1 present the Euclidean distance error of the eight corner points from the brain images. The median Euclidean distance before registration is 21.7 mm. The result of the FASGD method is very similar to the ASGD method: median accuracy is 1.6, 1.6 and 1.7 mm for ASGD, ASGD′ and FASGD, respectively. The p value of the Wilcoxon signed rank test of FASGD compared with ASGD and ASGD′ is 0.36 and 0.30, respectively, indicating no statistical difference.

2.5.1.2 SPREAD lung CT data

Table 2.2 shows the median of the mean Euclidean distance error of the 100 corresponding points of 19 patients for four different similarity measures. Compared with ASGD, FASGD has a significant difference for MSD, MI and NMI, but the median error difference is smaller than 0.03 mm.

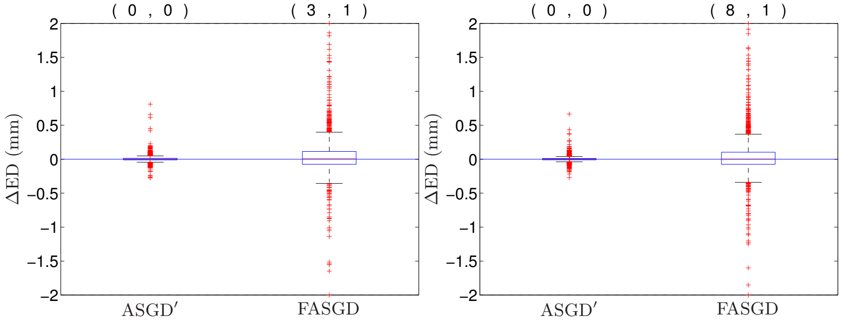
	Initial	ASGD	ASGD$'$	FASGD
MSD	3.62	1.09	1.10 ×	1.12 † ‡
NC	3.56	1.50	1.51 †	1.55 × ×
MI	3.17	1.65	1.65 †	1.66 † ‡
NMI	3.17	1.66	1.65 ×	1.68 † ‡

Table 2.2: The median Euclidean distance error (mm) for the SPREAD lung CT data. The symbols † and ‡ indicate a statistically significant difference with ASGD and ASGD$'$, respectively. × denotes no significant difference.



(a) MSD

(b) NC

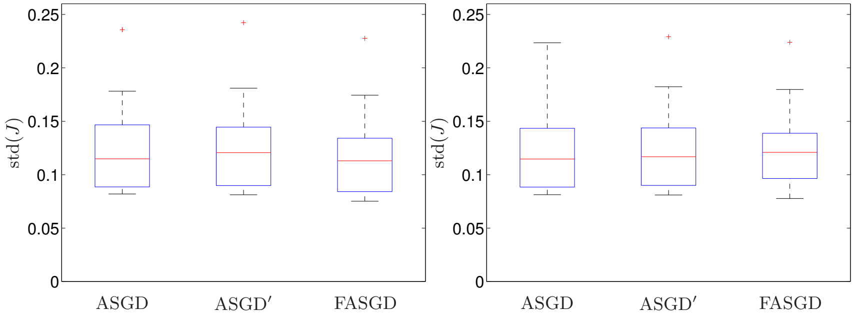


(c) MI

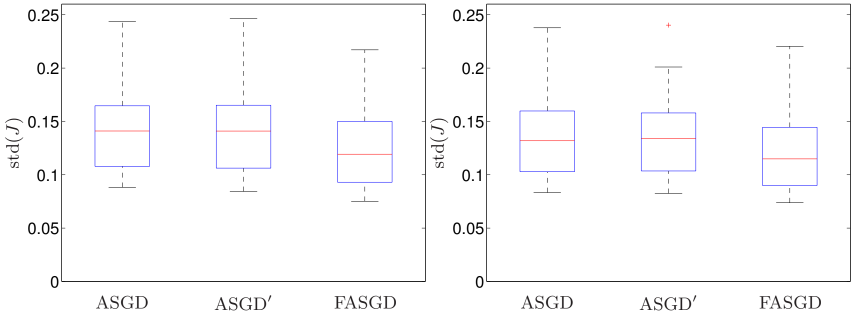
(d) NMI

Figure 2.2: The difference of Euclidean distance error in mm compared to ASGD for the SPREAD lung CT data. The two numbers on the top of each box denote the number of the landmark errors larger (left) and smaller (right) than 2 and -2 mm, respectively. All those landmarks, except one for NMI, belong to the same patient.

To compare FASGD and ASGD$'$ with ASGD we define the Euclidean landmark error difference as $\Delta\text{ED}_i = \text{ED}_i^{\text{FASGD}} - \text{ED}_i^{\text{ASGD}}$, for each landmark i, and similarly for ASGD$'$. This difference is shown as a box plot in Figure 2.2. Negative numbers mean that FASGD is better than ASGD, and vice versa. It can be seen that both ASGD$'$ and FASGD provide results similar to ASGD, for all tested cost functions. The spread of the ΔED box plot for ASGD$'$ is smaller than that of FASGD, as this method is almost identical to ASGD.



(a) MSD (b) NC



(c) MI (d) NMI

Figure 2.3: Box plots of the standard deviation of the Jacobian determinant J for the four similarity measures.

Smoothness of the resulting transformations is given in Figure 2.3 for all similarity measures. FASGD generates somewhat smoother transformations over ASGD and ASGD′ for the MSD, MI and NMI measures.

2.5.1.3 Hammers brain data

In this experiment, FASGD is compared with ASGD and ASGD′ in a large scale intersubject experiments on brain MR data, for a range of values of A, δ and the number of iterations K.

Figure 2.4 shows the overlap results of the 83 brain regions. Each square represents the median DSC result of 870 brain image registration pairs for a certain parameter combination of A, δ and K. These results show that the original ASGD method has a slightly higher DSC than FASGD with the same parameter setting, but the median DSC difference is smaller than 0.01. Note that the dark black color indicates DSC values between 0 and 0.5, i.e. anything between registration failure and low performance. The ASGD and ASGD′ methods fail for $\delta \geq 32$ mm, while FASGD fails for $\delta \geq 256$ mm.

2.5.1.4 Ultrasound Abdomen data

The results shown in Figure 2.5 present the Euclidean distance of 22 landmarks from ultrasound images after nonrigid registration. The median Euclidean distance before registration is 3.6 mm. The result of FASGD is very similar to the original method. The

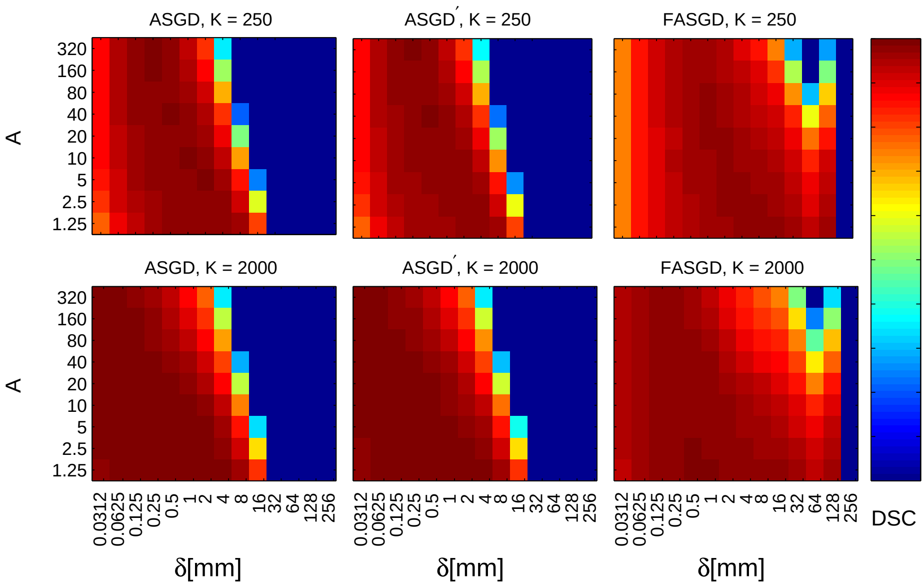


Figure 2.4: Median dice overlap after registration of the Hammers brain data, as a function of A and δ. A high DSC indicates better registration accuracy. Note that in this large scale experiment, each square represents 870 registrations, requiring about 870 × 15 minutes of computation, i.e. almost 200 core hours.

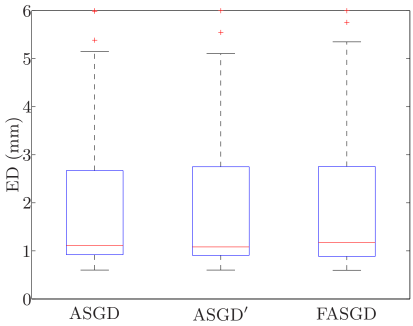
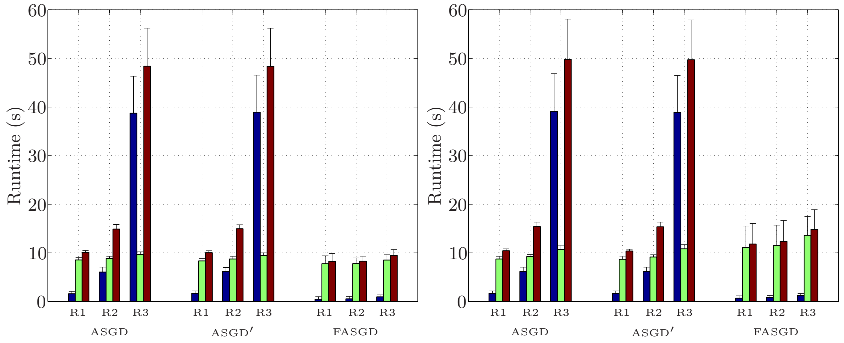


Figure 2.5: Euclidean distance in mm of the registration results for Ultrasound data performed using MI.

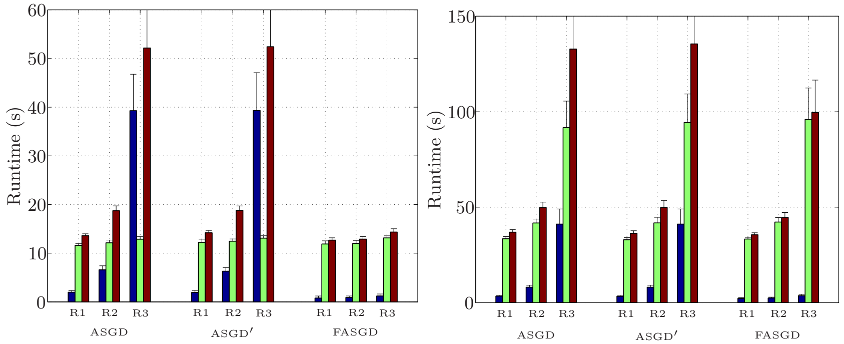
p value of the Wilcoxon signed rank test of FASGD compared with ASGD and ASGD′ is 0.485 and 0.465, respectively, indicating no statistical difference.

2.5.2 Runtime results

In this section the runtime of the three methods, ASGD, ASGD′ and FASGD is compared.



(a) MSD (b) NC



(c) MI (d) NMI

Figure 2.6: Runtime of SPREAD lung CT data in seconds. The black, green and red bar indicate estimation time, pure registration time and total time elapsed in each resolution, respectively. R1, R2, R3 indicate a three level multi-resolution strategy from low resolution to high resolution.

2.5.2.1 SPREAD lung CT data

The runtime on SPREAD lung CT data is shown in Figure 2.6, in which the time used in the estimations of the original method takes a large part of the total runtime per resolution, while FASGD consumes only a small fraction of the total runtime. From resolution 1 (R1) to resolution 3 (R3), the number of transformation parameters P increases from 4×10^3 to 9×10^4. For both ASGD and ASGD′ the estimation time increases from 3 seconds in R1 to 40 seconds in R3. However, FASGD maintains a constant estimation time of no more than 1 second.

2.5.2.2 Hammers brain data

The runtime result of the Hammers brain data is shown in Figure 2.7. For this dataset, $P \approx 1.5 \times 10^5$ in R3, i.e. larger than for the SPREAD data, resulting in larger estimation times. For ASGD and ASGD′ the estimation time in the third resolution is almost 95 seconds, while for FASGD it is almost 2 orders of magnitude smaller (≤ 1s).

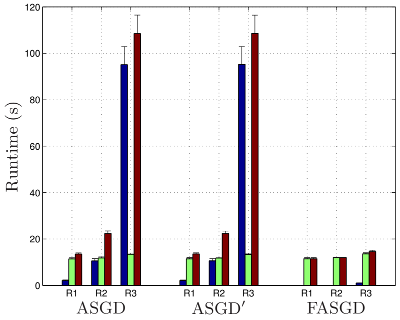


Figure 2.7: Runtime of Hammers brain data experiment in seconds. The black, green and red bar indicate estimation time, pure registration time and total time elapsed in each resolution, respectively. R1, R2, R3 indicate a three level multi-resolution strategy from low resolution to high resolution.

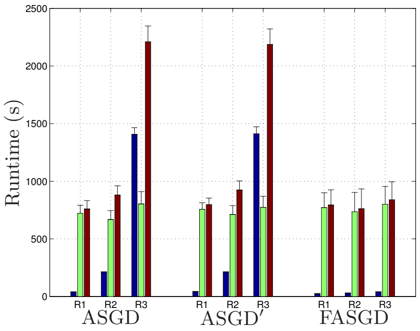


Figure 2.8: Runtime of Ultrasound data experiment in seconds. The black, green and red bar indicate estimation time, pure registration time and total time elapsed in each resolution, respectively. R1, R2, R3 indicate a three level multi-resolution strategy from low resolution to high resolution.

2.5.2.3 4D ultrasound data

The grid spacing of B-spline control points used in the 4D ultrasound data experiment is $15 \times 15 \times 15 \times 1$ and the image size is $227 \times 229 \times 227 \times 96$, so the total number of B-spline parameters for the third resolution R3 is around 8.7×10^5. From the timing results in Figure 2.8, the original method takes almost 1400 seconds, i.e. around 23 minutes, while FASGD only takes 40 seconds.

Figure 2.9 presents the runtime of estimating a_{max} and η for the ultrasound data. The estimation of η takes a constant time during each resolution, so for small P the estimation of η dominates the total estimation time.

2.5.3 Convergence

From each of the four experiments, we randomly selected one patient and analyzed the step size sequence $\{\gamma_k\}$. The results are presented in Figure 2.10 and show that

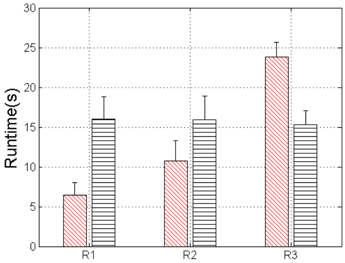


Figure 2.9: Runtime in seconds of FASGD for ultrasound experiment. The left bar indicate estimation time of $a_{\max}$ and the right bar is the estimation time of η. R1, R2, R3 indicate a three level multi-resolution strategy from low resolution to high resolution.

FASGD takes a larger step size than ASGD and ASGD′ for rigid registration and a smaller step size for nonrigid registration, when using the same δ. In addition, the original ASGD and ASGD′ take a very similar step size in all experiments even when ASGD′ uses the default settings for $f_{\min}$ and ω.

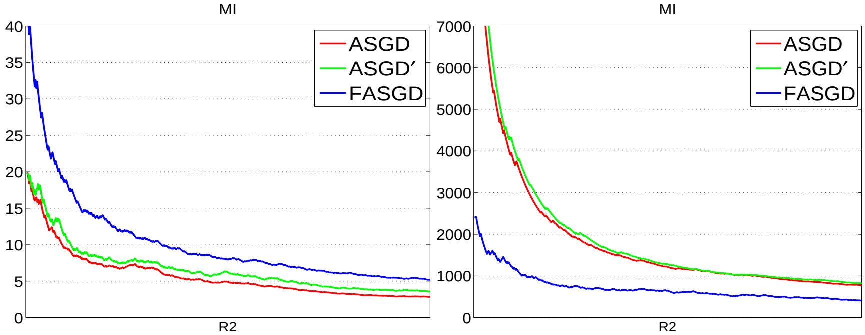
Convergence results of the three methods are presented in Figure 2.11 for several patients. Figure 2.11a and 2.11b present the Euclidean distance (mm) at each iteration for three resolutions with respect to the iteration number. The cost function values are shown in Figure 2.11c and 2.11d. The three methods behave similarly.

2.6 Discussion

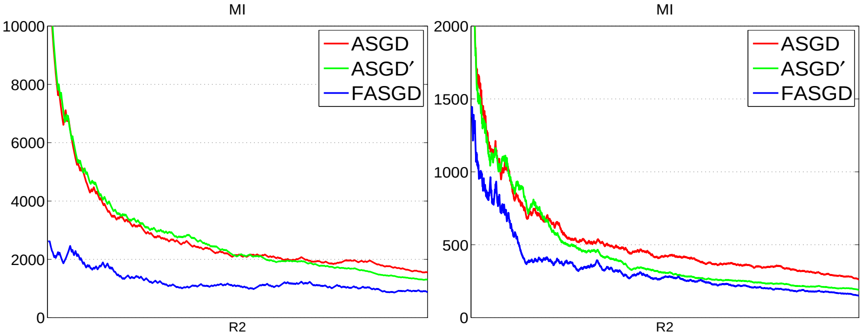
All experiments in this chapter show that the fast ASGD method works well both in rigid and nonrigid image registration, showing that the method can deal with differently parameterized transformations. The method was thoroughly evaluated on a variety of imaging problems, including different modalities such as CT, MRI and ultrasound, intra and inter subject registration, and different anatomical sites such as the brain, lung and abdomen. Various image registration settings were tested, including four popular similarity measures. A very large scale experiment investigated the sensitivity of the methods to the parameters A and δ.

All experiments show that FASGD has similar accuracy as the ASGD method. For the rigid registration on the RIRE data and the nonrigid 4D ultrasound experiment there was no significant statistical difference. For the nonrigid SPREAD lung CT experiment and the Hammers brain data we observed statistically significant differences, however, these differences were very small: on average less than 0.03 mm on the SPREAD data (less than 5% of the voxel size), and less than 0.01 Dice overlap on brain data. We conclude that FASGD obtains a very similar registration accuracy as the original ASGD method.

All results indicate that there is little difference between ASGD and ASGD′. Especially from Figure 2.10 it can be observed that both methods take very similar step size during the optimization, as well as similar cost function value and Euclidean distance error (Figure 2.11). This suggests that the default values of the parameters $f_{\min}$ and ω are sufficiently accurate, and that indeed the parameter a is the most important



(a) RIRE (b) SPREAD



(c) Hammers (d) Ultrasound

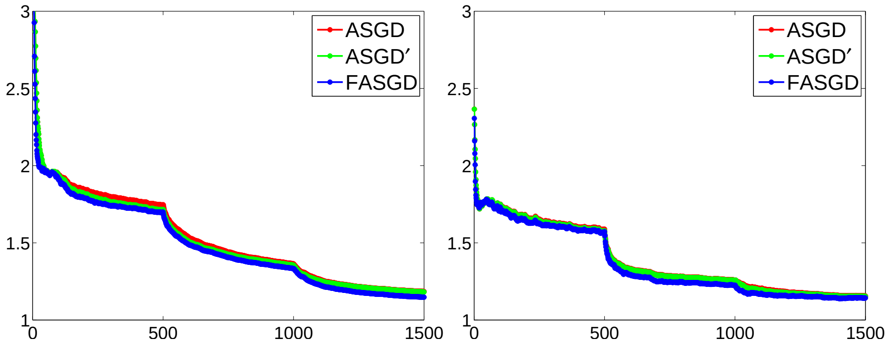
Figure 2.10: An example of the step size decay using 500 iterations except Ultrasound image data (2000 iterations) in last resolution from four experiments. The red line is the original ASGD, the black line is ASGD′ and the green line is FASGD.

parameter to estimate.

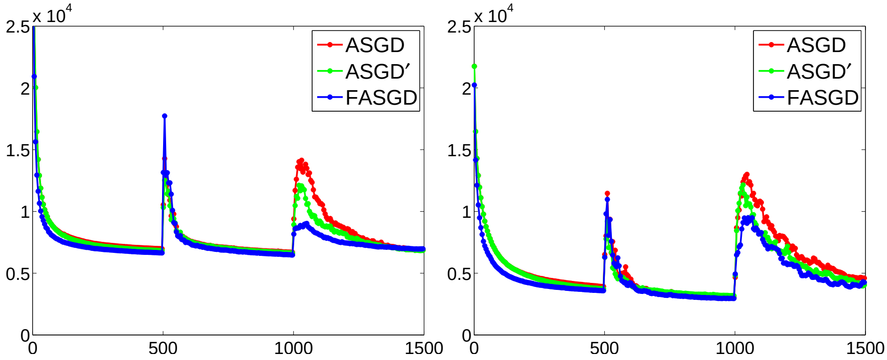
From Figure 2.10 it can be observed that FASGD typically estimates smaller step sizes than ASGD, for identical δ. This was also observed for the other patients. Figure 2.4 confirms this observation, as the accuracy plot for FASGD is somewhat shifted to the right compared to the other two methods. This suggests that more similar step sizes may be obtained when choosing δ about twice as large as for ASGD, i.e. to increase the default from one voxel size to two.

The accuracy results for the Hammers experiment shown in Figure 2.4 present an apparent accuracy increase when $\delta = 128$ for FASGD. Remember that δ represents the maximum allowed voxel displacement per iteration in mm, and that for the medical data used in this chapter larger δ are unrealistic. Note that for ASGD the registrations start failing when $\delta \geq 32$, and for FASGD when $\delta > 128$. The temporary increase in accuracy at $\delta = 128$ for FASGD is due to an undesired decrease in $\eta \times \delta$. Note that ASGD uses the exact same term, see Equation (2.20), but this does not result in increased accuracy, since ASGD is already failing for $\delta = 128$.

The time performance of the proposed method shown in Section 2.5.2 implies that



(a) ED, patient 1 (b) ED, patient 2



(c) MSD, patient 3 (d) MSD, patient 4

Figure 2.11: Convergence plots for four different patients. Top row shows the Euclidean distance error (mm) as a function of the iteration number. Bottom row shows the cost function value (MSD). Each plot shows three resolutions.

FASGD has a large reduction in time consumption of the step size estimation. For the SPREAD experiment the estimation time in the last resolution is reduced from 40 seconds to 1 second. This improvement is crucial for near real-time registration in high dimensional image registration.

From Figure 2.9 it is observed that a new bottle neck in the step size estimation is the estimation of the noise compensation parameter η. This is because in this work the calculation of the gradient $\boldsymbol{g}$ is performed with a relatively high number of voxels from the fixed image. Future work will include the investigation of accelerated methods to estimate η and so further reduce the step size estimation time, especially for 4D registration problems. A direct acceleration possibility is the use of parallelization, for example by a GPU implementation, as the gradient computation consists of an independent loop over the voxels.

The FASGD method provides a solution for step size selection for gradient descent optimizers. For Newton-like optimizers this is typically solved by a line search strategy. Note that such a strategy can not readily be adopted for stochastic optimization due to the stochastic approximation of the cost function [60]. Strengths of quasi-Newton optimizers are their adaptability to problems where the parameters are scaled with respect to each other, and the availability of stopping conditions. For FASGD as well as other stochastic gradient descent optimization routines typically the number of iterations is used to terminate the optimization. More sophisticated stopping conditions from deterministic gradient descent methods cannot be readily adopted. For example, due to the estimation noise, stopping conditions based on cost function values or cost function gradients cannot be trusted. The alternative to compute exact objective values every (few) iteration(s), is also not attractive due to the required computation time. In the `elastix` implementation a stochastic gradient computation is in the order of 50 ms, while exact metric value computation is at least in the order of seconds. A feasible possibility would be to create a stopping condition based on a moving average of the noisy objective values or gradients.

The use of the `lsgrid` for the Hammers data experiment was essential, and reduced computation time from 19 years to about 2-3 days. It however did require a one-time investment of time to develop the software supporting the registration jobs on the grid. Typical issues we encountered was attempting to store the results from hundreds of simultaneous executions, which proved incompatible with maximum transaction rate supported by the `lsgrid` Storage Resource Management services. We were able to solve this by pooling multiple results into a single storage operation. The infrastructure we built therefore screens the software under execution from the complexities that are encountered when running on the `lsgrid`. At the same time it is generic enough to provide a configurable set of execution environments to support other experiments not just the `elastix` workflow used in this work, and can therefore be re-used.

2.7 Conclusion

In this chapter, a new automatic method (FASGD) for estimating the optimization step size parameter a, needed for gradient descent optimization methods, has been presented for image registration. The parameter a is automatically estimated from the magnitude of voxel displacements, randomly sampled from the fixed image. A relation between the step size and the expectation and variance of the observed voxels displacement is derived. The proposed method has a free parameter δ, defining the maximally allowed incremental displacement between iterations. Unlike a, it can be interpreted in terms of the voxel size (mm). In addition, it is mostly independent of the application domain, i.e. setting it equal to the voxel size provided good results for all applications evaluated in this chapter. Compared to the original ASGD method, the time complexity of the FASGD method is reduced from quadratic to linear with respect to the dimension of the transformation parameters P. For the B-spline transformation, due to its compact support, the time complexity is further reduced, making the proposed method independent of P. The FASGD method is publicly available via the open source image registration toolbox `elastix` [10].

The FASGD method was evaluated on a large number of registration scenario's and shows a similar accuracy as the original ASGD method. It however improves the time complexity of the step size estimation from 40 seconds to no more than 1 second, when the number of parameters is $\sim 10^5$: almost 40 times faster. Depending on the registration settings, the total registration time is reduced by a factor of 2.5-7x for the experiments in this chapter.

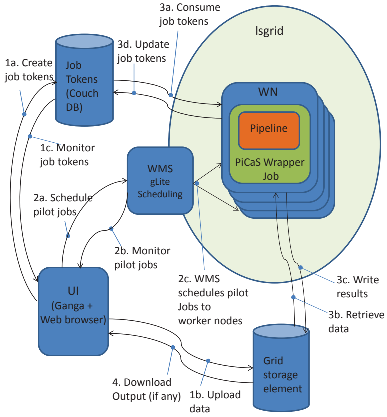


Figure 2.12: Running the Hammers pipeline in the pilot job architecture used on the `lsgrid`. Arrows represent the flow of information.

Appendix

The `lsgrid` infrastructure comprises distributed computing and storage resources along with a central grid facility. In total there is potential for approximately 10000 job slots. Job scheduling is performed using gLite grid middleware [61] via the gLite Workload Management System (WMS) [62], which was developed for the European Grid Infrastructure [63].

While it is possible to use this directly to schedule registration pipeline jobs, in practice these relatively short jobs are a poor fit to the standard queue lengths in `lsgrid`. In addition, unforseen delays in the push scheduling mechanism result in a considerable overhead [64]. These issues can be addressed by layering a pull scheduling system based on pilot jobs onto the grid software infrastructure. Matching jobs to Workload Nodes occurs once at pilot job startup after which job tokens are pulled into the pilot job environment. The concept of Pilot Jobs was first pioneered in the EGI grid within DIRAC [65], but we employed a light weight pilot job system developed by SURFsara called PiCaS [66], [67].

The pilot job architecture shown in Figure. 2.12 was used to execute the Hammers pipeline. PiCaS was extended with a wrapper job to perform standard elements of the pipeline such as environment setup and data retrieval. The wrapper job and the Hammers pipeline are coded using Python [68]. The job tokens contain the registration parameters to be used and the storage locations for the fixed and moving images. Ganga [69] is used to schedule and monitor pilot jobs which pull and execute the job-tokens from the PiCaS database. The overall progress of the execution can be checked by monitoring the status of the job tokens using the web browser to access job-token views defined in database.

Execution of the Hammers pipeline using PiCaS on the `lsgrid` follows these steps:

1. Initialize the Hammers jobs tokens. (a) Create the job tokens for each Hammers pipeline run. Job tokens contain job parameters and the grid location of the input data. (b) Upload the input data needed to specific locations in grid storage. (c) Monitor execution progress by checking job token consumption in a browser.

2. Schedule the pilot jobs to commence grid execution. (a) Schedule pilot jobs with the necessary job requirements using gLite WMS from inside Ganga. Additional information is passed to the pilot job concerning the runtime environment needed. (b) Monitor the progress of the pilot jobs using Ganga job monitoring. (c) gLite WMS identifies clusters matching the job requirements and schedules pilot jobs. Once the pilot is started the PiCaS Wrapper Job sets up the runtime environment on the worker node.

3. Job tokens are consumed and executed by the running pilot jobs. (a) Retrieve a job token from the PiCaS job tokens database and mark it as locked. (b) The necessary data identified in the job token for each Hammers job is downloaded by the PiCaS wrapper from grid storage and the Hammers pipeline is executed. (c) Any results are uploaded to the grid storage location as specified in the job token. (d) The job token is updated with the result: success or failure. In failure cases log-files are appended to assist in debugging.

4. Job results can be immediately downloaded while the run is in progress.

All tools that were created are reusable for other large scale image processing with the `lsgrid`.

Acknowledgment

This research was supported by the Netherlands Organization for Scientific Research (NWO VENI 639.021.919), by the Dutch national e-infrastructure with the support of the SURF Foundation (e-infra140085), and by the China Scholarship Council (No. 201206130066). The RIRE project is acknowledged for providing a platform for rigid registration evaluation. We are grateful to dr. A. Hammers *et al.* for the adult maximum probability brain atlas. Dr. M.E. Bakker and J. Stolk are acknowledged for providing a ground truth for the SPREAD study data used in this chapter. The 4D ultrasound data was made available by SINTEF Dept. Medical Technology and the Norwegian University of Science and Technology in the context of the IIIOS project (Marie Curie ITN no 238802).

3

A Stochastic Quasi-Newton Method for Non-rigid Image Registration

This chapter was adapted from:

Y. Qiao, Z. Sun, B.P.F. Lelieveldt and M. Staring. **A Stochastic Quasi-Newton Method for Non-rigid Image Registration**, *Medical Image Computing and Computer-Assisted Intervention, Lecture Notes in Computer Science*, Volume 9350, Pages 297–304, 2015.

Abstract

Image registration is often very slow because of the high dimensionality of the images and complexity of the algorithms. Adaptive stochastic gradient descent (ASGD) outperforms deterministic gradient descent and even quasi-Newton in terms of speed. This method, however, only exploits first-order information of the cost function.

In this chapter, we explore a stochastic quasi-Newton method (s-LBFGS) for non-rigid image registration. It uses the classical limited memory BFGS method in combination with noisy estimates of the gradient. Curvature information of the cost function is estimated once every L iterations and then used for the next L iterations in combination with a stochastic gradient. The method is validated on follow-up data of 3D chest CT scans (19 patients), using a B-spline transformation model and a mutual information metric. The experiments show that the proposed method is robust, efficient and fast. s-LBFGS obtains a similar accuracy as ASGD and deterministic LBFGS. Compared to ASGD the proposed method uses about 5 times fewer iterations to reach the same metric value, resulting in an overall reduction in run time of a factor of two. Compared to deterministic LBFGS, s-LBFGS is almost 500 times faster.

3.1 Introduction

Image registration is important in the field of medical image analysis. However, this process is often very slow because of the large number of voxels in the images and the complexity of the registration algorithms [20, 25]. A powerful optimization method is needed to shorten the time consumption during the registration process, which would benefit time-critical intra-operative procedures relying on image guidance.

The stochastic gradient descent method is often used to iteratively find the optimum [15]. This method is easy to implement and fast because at each iteration only a subset of voxels from the fixed image is evaluated to obtain gradients. Although it obtains a good accuracy, its convergence rate is poor since only first order derivatives are used. A preconditioning matrix can be used to improve the convergence rate of (stochastic) gradient descent, but this was only proposed in a mono-modal setting [70]. The quasi-Newton method also has a better convergence rate than deterministic gradient descent, but comes at a higher cost in computation time and large memory consumption. Limited memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) takes an advantage in the storage of only a few previous Hessian approximations, however, the computation time is still very long as all voxels are needed for new Hessian approximations [25].

Some approaches to create a stochastic version of the quasi-Newton method are proposed in a mathematical setting, such as online LBFGS [71], careful quasi-Newton stochastic gradient descent [72], regularized stochastic BFGS [73] and stochastic LBFGS [74]. However, there is no application in the image registration field, and applying the stochastic quasi-Newton method to non-rigid image registration is still a challenge. All of the previous methods either used a manually selected constant step size or a fixed decaying step size, which are not flexible when switching problem settings or applications. Moreover, the uncertainty of gradient estimation introduced by the stochastic gradient for Hessian approximation is still a problem. Although Byrd [74] used the exact Hessian to compute curvature updates, which is still difficult to calculate for high dimensional problems. For careful QN-SGD [72], the average scheme may be useless in case of an extremely large or small scaling value for H_0. Mokhtari [73] used a regularized term like Schraudolph [71] did to compensate the gradient difference y from the parameter difference s and introduced a new variable δ, which is not only complex, but also needs to store all previous curvature pairs.

In this chapter, we propose a stochastic quasi-Newton method specifically for non-rigid image registration inspired by Byrd *et al.* [74]. Different from Byrd's method, the proposed method employs only gradients and avoids computing second order derivatives of the cost function to capture the curvature. Secondly, we employ an automatic and adaptive scheme for optimization step size estimation instead of a fixed manual scheme. Finally, we propose a restarting mechanism where the optimal step size is recomputed when a new Hessian approximation becomes available, i.e. every L iterations. The proposed method and some variations are validated using 3D lung CT follow-up data using manually annotated corresponding points for evaluation.

3.2 Methods

Non-rigid image registration aims to align images following a continuous deformation strategy. The optimal transformation parameters are the solution that minimizes the dissimilarity between fixed I_F and moving image I_M:

$$\widehat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} \mathscr{C}(I_F, I_M \circ \boldsymbol{T}_{\boldsymbol{\mu}}), \tag{3.1}$$

in which $\boldsymbol{T}_{\boldsymbol{\mu}}(\boldsymbol{x})$ is a coordinate transformation parameterized by $\boldsymbol{\mu}$.

3.2.1 Deterministic quasi-Newton

The deterministic quasi-Newton method employs the following iterative form:

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k - \gamma_k \boldsymbol{B}_k^{-1} \boldsymbol{g}_k, \tag{3.2}$$

where $\boldsymbol{B}_k$ is a symmetric positive definite approximation of the Hessian matrix $\nabla^2 \mathscr{C}(\boldsymbol{\mu}_k)$. Quasi-Newton methods update the inverse matrix $\boldsymbol{H}_k = \boldsymbol{B}_k^{-1}$ directly using only first order derivatives, and have a super-linear rate of convergence. Among many methods to construct the series $\{\boldsymbol{H}_k\}$, Broyden-Fletcher-Goldfarb-Shanno (BFGS) tends to be efficient and robust in many applications. It uses the following update rule for $\boldsymbol{H}_k$:

$$\boldsymbol{H}_{k+1} = \boldsymbol{V}_k^T \boldsymbol{H}_k \boldsymbol{V}_k + \rho_k \boldsymbol{s}_k \boldsymbol{s}_k^T, \tag{3.3}$$

in which

$$\rho_k = \frac{1}{\boldsymbol{y}_k^T \boldsymbol{s}_k}, \quad \boldsymbol{V}_k = \boldsymbol{I} - \rho_k \boldsymbol{y}_k \boldsymbol{s}_k^T, \quad \boldsymbol{s}_k = \boldsymbol{\mu}_{k+1} - \boldsymbol{\mu}_k, \quad \boldsymbol{y}_k = \boldsymbol{g}_{k+1} - \boldsymbol{g}_k. \tag{3.4}$$

Since the cost of storing and manipulating the inverse Hessian approximation $\boldsymbol{H}_k$ is prohibitive when the number of the parameters is large, a frequently used alternative is to only store the latest M curvature pairs $\{\boldsymbol{s}_k, \boldsymbol{y}_k\}$ in memory: limited memory BFGS (LBFGS). The matrix $\boldsymbol{H}_k$ is not calculated explicitly, and the product $\boldsymbol{H}_k \boldsymbol{g}_k$ is obtained based on a 2-rank BFGS update, which uses a two loop recursion [74]. The initial inverse Hessian approximation usually takes the following form, which we also use in this chapter:

$$\boldsymbol{H}_k^0 = \theta_k \boldsymbol{I}, \quad \theta_k = \frac{\boldsymbol{s}_{k-1}^T \boldsymbol{y}_{k-1}}{\boldsymbol{y}_{k-1}^T \boldsymbol{y}_{k-1}}. \tag{3.5}$$

3.2.2 Stochastic quasi-Newton

A large part of the computation time of quasi-Newton methods is in the computation of the curvature pairs $\{\boldsymbol{s}_k, \boldsymbol{y}_k\}$. The pairs are computed deterministically using all samples from the fixed image. A straightforward way to obtain a stochastic version of the quasi-Newton method is to construct the curvature pairs using stochastic gradients, using only a small number of samples at each iteration. This however introduces too much noise in the curvature estimation, caused by the fact that stochastic gradients are inherently noisy and for each iteration are also evaluated on different subsets of image voxels, both of which may yield a poor Hessian approximation. This leads to instability in the optimization.

To cope with this problem, Byrd *et al.* [74] proposed a scheme to eliminate the noise by averaging the optimization parameters for a regular interval of L iterations and obtain the curvature through a direct Hessian calculation on a random subset $\mathscr{S}_2$.

This is combined with a series of L iterations performing LBFGS using the thus obtained inverse Hessian estimate together with *stochastic* gradients (using a small random subset $\mathscr{S}_1$). Inspired by this scheme, we propose a method suitable for medical image registration and avoiding manual tuning the step size. First, more samples are used for the curvature pair update than for the stochastic gradient evaluation. Second, the curvature information is obtained using a gradient difference instead of second order derivatives evaluated at an identical subset of samples, i.e. $\boldsymbol{y}_t = \boldsymbol{g}(\bar{\boldsymbol{\mu}}_I; \mathscr{S}_2) - \boldsymbol{g}(\bar{\boldsymbol{\mu}}_J; \mathscr{S}_2)$ and the curvature condition $\boldsymbol{y}^T\boldsymbol{s} > 0$ is checked to ensure positive definiteness of the LBFGS update [74]. Finally, the initial step size at the beginning of each L iterations is automatically determined, with or without restarting. Restarting is a recent development [75] showing improved rate of convergence, which in this chapter we apply to the step size selection.

Instead of manual constant step size selection as in [74], we employ an automatic method. A commonly used function for the step size which fulfils the convergence conditions [76] is $\gamma_k = \eta a/(t_k + A)^{\alpha}$, with $A \geq 1$, $a > 0$, $0 \leq \eta \leq 1$ and $0 \leq \alpha \leq 1$. The step size factor a and the noise compensation factor η are automatically determined through the statistical distribution of voxel displacements [45], while $A = 20$ according to [15] and $\alpha = 1$ is theoretically optimal [15]. Different strategies for the artificial time parameter t_k are tested: a constant step size $t_k = 0$, a regularly decaying step size $t_k = k$, and an adaptive step size $t_k = f(\cdot)$. Here, f is a sigmoid function with argument of the inner product of the gradients $\tilde{\boldsymbol{g}}_k^T \cdot \tilde{\boldsymbol{g}}_{k-1}$ for gradient descent. For s-LBFGS, it can be derived that the search direction is needed as argument, i.e. $\boldsymbol{d}_k^T \cdot \boldsymbol{d}_{k-1}$ with $\boldsymbol{d}_k = \boldsymbol{B}_k^{-1}\boldsymbol{g}_k$.

An overview of the proposed s-LBFGS method is given in Algorithm 1.

3.3 Experiment

The proposed method was integrated in the open source software package `elastix` [10]. The experiments were performed on a workstation with 8 cores running at 2.4 GHz and 24 GB memory, with an Ubuntu Linux OS.

3D lung CT scans of 19 patients acquired during the SPREAD study [77] were used to test the performance. Each patient had a baseline and a follow-up scan with an image size around $450 \times 300 \times 150$ and the voxel size around $0.7 \times 0.7 \times 2.5$ mm. For each image, one hunred anatomical corresponding points were chosen semi-automatically using Murphy's method in consensus by two experts, to obtain a ground truth.

To evaluate the method, each follow-up image was registered to the baseline image using mutual information and a B-spline transformation model. The maximum number of iterations for each resolution was 500. A three-level multi-resolution framework was employed using a Gaussian smoothing filter with standard deviations of 2, 1 and 0.5 mm for each resolution. The grid spacing of the B-spline control points was halved between each resolution resulting in a final grid spacing of 10 mm in each direction. After initial testing, we chose the update frequency $L = 10, 20, 40$ for each resolution, respectively, the memory $M = 5$ from [25, 74], the number of samples for stochastic gradient computation $|S_1| = 5000$, and the number of samples for the curvature pair update $|S_2| = 50000$.

To measure the registration accuracy, the anatomical points from each baseline image were transformed using the obtained transformation parameters and then

Algorithm 1 Stochastic LBFGS (s-LBFGS) with and without restarting

```
Require: initial parameters $\boldsymbol{\mu}_0$, memory size $M$, update frequency $L$, iteration number $K$
Set $t = 0$, $\bar{\boldsymbol{\mu}}_J = \boldsymbol{\mu}_0, \bar{\boldsymbol{\mu}}_I = \mathbf{0}$
Automatically estimate the initial step size $\lambda_0$    ▷ According to [45]
for $k = 1,2,3,\dots,K$ do
    Compute $\tilde{\boldsymbol{g}}_k(\boldsymbol{\mu}_k;\mathscr{S}_1)$    ▷ stochastic gradient
    $\bar{\boldsymbol{\mu}}_I = \bar{\boldsymbol{\mu}}_I + \boldsymbol{\mu}_k$    ▷ Update the mean parameters
    if $k <= 2L$ then    ▷ ASGD update
        Update the step size $\lambda_k$    ▷ According to [45]
        $\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k - \lambda_k \tilde{\boldsymbol{g}}_k$
    else    ▷ s-LBFGS update
        Compute $\boldsymbol{d}_k = \boldsymbol{H}_t \tilde{\boldsymbol{g}}_k$    ▷ s-LBFGS search direction, see [74] and (3.2)
        if mod $(k, L) = 0$ and restarting then
            Automatically estimate the initial step size $\lambda'_0$
            Reset $\lambda_k = \lambda'_0$
        Update the step size $\lambda_k$    ▷ According to [45] but using $\boldsymbol{d}_k^T \cdot \boldsymbol{d}_{k-1}$
        $\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k - \lambda_k \boldsymbol{d}_k$
    if mod $(k, L) = 0$ then    ▷ Curvature pairs update
        $\bar{\boldsymbol{\mu}}_I = \bar{\boldsymbol{\mu}}_I / L$    ▷ Update the mean parameters
        $\boldsymbol{s}_t = \bar{\boldsymbol{\mu}}_I - \bar{\boldsymbol{\mu}}_J, \quad \boldsymbol{y}_t = \boldsymbol{g}(\bar{\boldsymbol{\mu}}_I;\mathscr{S}_2) - \boldsymbol{g}(\bar{\boldsymbol{\mu}}_J;\mathscr{S}_2)$    ▷ New curvature pair
        $\bar{\boldsymbol{\mu}}_J = \bar{\boldsymbol{\mu}}_I, \bar{\boldsymbol{\mu}}_I = \mathbf{0}, t = t + 1$
    return $\boldsymbol{\mu}_K$
```

compared to the corresponding points of the follow-up image. We used the Euclidean distance between the corresponding points $\boldsymbol{p}_F \in \Omega_F$ and $\boldsymbol{p}_M \in \Omega_M$ to measure the accuracy using the following equation:

$$ED = \frac{1}{n} \sum_{i=1}^{n} \| \boldsymbol{T}(\boldsymbol{p}_F^i) - \boldsymbol{p}_M^i \|. \tag{3.6}$$

For 19 patients, we first obtained the mean distance error of 100 points for each patient then performed Wilcoxon signed rank test to these mean errors. For convergence testing we computed the cost function value after each iteration deterministically, i.e. based on full sampling. The registration time in the first resolution is presented to compare the algorithm speeds.

3.4 Results

To gain insight in the proposed method, we investigated some aspects that influence registration performance. The restarting scheme (Restart) was compared with a scheme without restarting. We evaluated different step size selection strategies all based on automatic initial step size selection [45]: a constant scheme (Constant, $t_k = 0$), a regularly decaying scheme (Decaying, $t_k = k$), and the proposed adaptive scheme (Adaptive, $t_k = f(\cdot)$). The proposed method is further compared with the ASGD method [15] and with deterministic LBFGS [25].

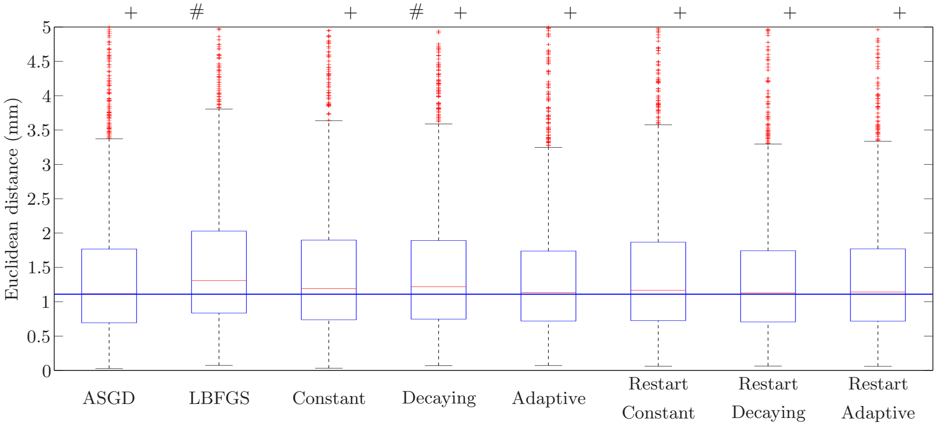
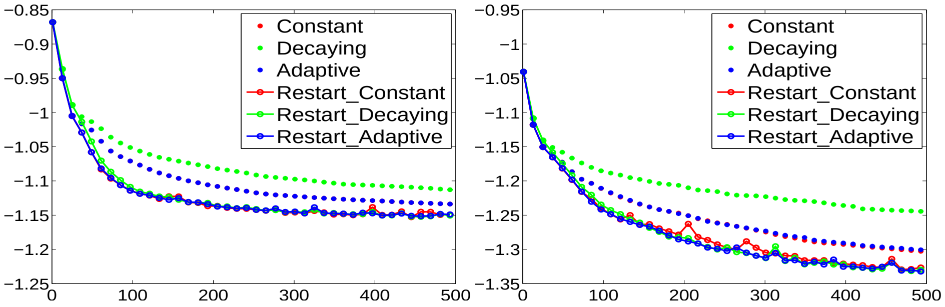


Figure 3.1: Euclidean distance error in mm. The symbols # and + indicate a statistically significant difference with ASGD and LBFGS, respectively.

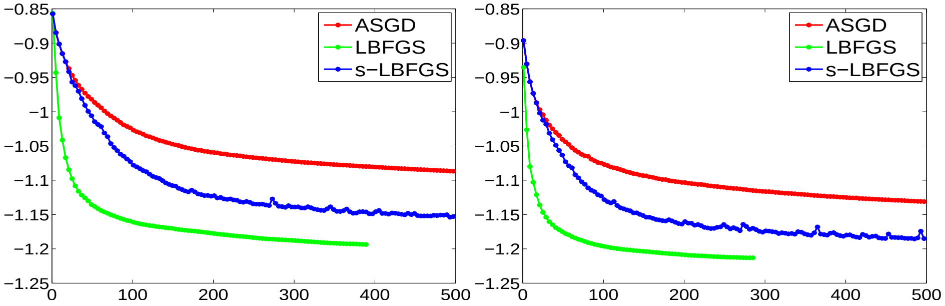
	Time at $k = 500$	Iterations I	Time (s) at $k = I$	Speed-up
ASGD	27.2 ± 0.7	500	-	-
LBFGS	26838 ± 9965	21 ± 1	8081 ± 1580	0.004 ± 0.0005
s-LBFGS-NR	74.3 ± 4.8	190 ± 93	30.6 ± 13.9	1.0 ± 0.4
s-LBFGS	75.8 ± 1.0	107 ± 17	18.1 ± 2.6	1.5 ± 0.2

Table 3.1: Run time in the first resolution. I indicates how many iterations are needed to reach the same metric value as ASGD after 500 iterations. s-LBFGS and s-LBFGS-NR are with and without restarting, both using adaptive step sizes. The speed-up is relative to ASGD.

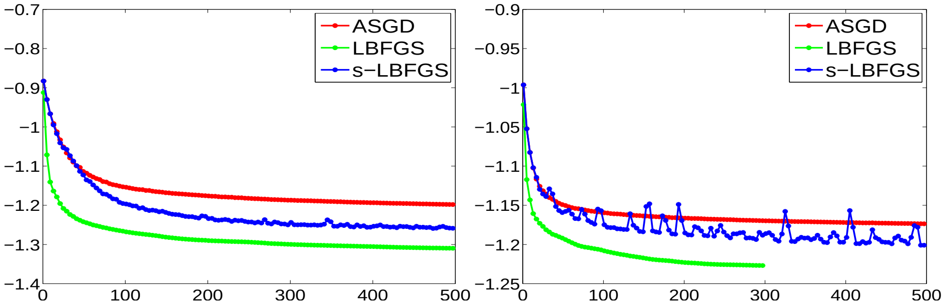
From Fig. 3.1 we can see that all methods have very similar final registration error, for LBFGS regularization may improve the results [59]. Fig. 3.2 shows the convergence plots of the methods for several patients. Comparing the three step size strategies in Fig. 3.2a and 3.2b, the regularly decaying method has suboptimal convergence, while the constant and the adaptive scheme behave similarly. The restarting scheme shows a substantial improvement in convergence rate, therefore in Fig. 3.2c~3.2f we only show the result of restarting scheme with adaptive step size (s-LBFGS). Some small spikes are visible in Fig. 3.2b and Fig. 3.2f, which we attribute to noise in the curvature pair estimation: an experiment using 1.5 million samples for the curvature estimation yielded smooth results (not shown). In terms of iterations, s-LBFGS always obtains faster convergence than ASGD, but slower than LBFGS. The registration time of ASGD, LBFGS and s-LBFGS is shown in Table 3.1. The LBFGS method is very costly, as expected. To obtain the same metric value as ASGD at iteration 500, the proposed method always takes fewer iterations resulting in an average speedup of two, while the proposed method without restarting requires more iterations and therefore more time.



(a) Patient 1 (b) Patient 2



(c) Patient 3 (d) Patient 4



(e) Patient 5 (f) Patient 6

Figure 3.2: Convergence plots, showing the negated mutual information metric against the iteration number.

3.5 Conclusion

In this chapter, we present for the first time a stochastic quasi-Newton optimization method (s-LBFGS) for non-rigid image registration. It uses the classical limited memory BFGS method in combination with noisy estimates of the gradient. Curvature information of the cost function is estimated robustly once every L iterations and then used for the next L iterations in combination with stochastic gradients. A novel restarting procedure, automatically selecting the optimization step size, is shown to be beneficial for accelerated convergence.

The new optimization routine is validated on follow-up data of 3D chest CT scans (19 patients). Compared to ASGD the proposed method uses about 5 times fewer iterations to reach the same metric value, resulting in an overall reduction in runtime of a factor of two. Compared to deterministic LBFGS, s-LBFGS is almost 500 times faster. Future work will focus on developing a stopping condition for stochastic second order procedures, on a more robust estimation of the initial approximation of H_0 more resilient against noise, on alterative quasi-Newton schemes such as the symmetric rank-one update [78], and more extensive validation.

4

An efficient preconditioner for stochastic gradient descent optimization of image registration

This chapter was adapted from:

Y. Qiao, B.P.F. Lelieveldt and M. Staring. **An efficient preconditioner for stochastic gradient descent optimization of image registration**, submitted

Abstract

Stochastic gradient descent (SGD) is commonly used to solve (parametric) image registration problems. In case of ill-conditioned problems, SGD however only exhibits sublinear convergence properties. In this chapter we propose an efficient preconditioner estimation method to improve the convergence rate of SGD. Based on the observed distribution of voxel displacements in the registration, we estimate the diagonal entries of a preconditioning matrix, thus rescaling the optimization cost function. The preconditioner is suitable for stochastic and not only deterministic optimization. It is efficient to compute and employ, and can be used for mono-modal as well as multi-modal cost functions, in combination with different transformation models like the rigid, affine and B-spline model. Experiments on different clinical data sets show that the proposed method indeed improves the convergence rate compared to SGD with speedups around 5 in all tested settings, while retaining the same level of registration accuracy.

4.1 Introduction

Image registration is widely used in medical image analysis and has ample application, e.g. in radiation therapy and segmentation [19, 2, 3]. This procedure can be used to align images from different modalities or different time points following a continuous deformation strategy. The strategy can be formulated as a (parametric) optimization problem to minimize the dissimilarity between a d-dimensional fixed image I_F and moving image I_M:

$$\widehat{\boldsymbol{\mu}} = \arg\min_{\boldsymbol{\mu}} \mathscr{C}(I_F, I_M \circ \boldsymbol{T}_{\boldsymbol{\mu}(\boldsymbol{x})}), \tag{4.1}$$

in which $\boldsymbol{T}_{\boldsymbol{\mu}}(\boldsymbol{x})$ is a coordinate transformation parameterized by $\boldsymbol{\mu}$. An iterative scheme is typically used to solve this problem:

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k - \gamma_k \boldsymbol{d}_k, \tag{4.2}$$

where k is the iteration number, γ_k is the step size at iteration k, and $\boldsymbol{d}_k$ is a search direction in the parameter space. Commonly used methods to determine the search direction $\boldsymbol{d}_k$ are of first order (gradient descent) or second order (Newton or quasi-Newton) descent type. Gradient descent, however, only achieves a sublinear convergence rate for nonconvex problems or a linear convergence rate for convex problems [79, 25]. Especially for badly scaled cost functions these methods converge slowly. Second order derivative methods such as the quasi-Newton method converge faster, however, the computation of the Hessian matrix update is very time consuming, especially when the number of image voxels and transformation parameters are large [80]. To overcome these shortcomings, preconditioning techniques were proposed to turn a badly scaled cost function into a properly scaled cost function, considering the curvature of the cost function [79, 81, 82, 83]. The construction of these preconditioners can however be computationally expensive in themselves, which can easily mitigate the positive effect of faster convergence.

Two major groups of preconditioning techniques are widely used in iterative optimization. One, sometimes named variable preconditioning, uses the update rule:

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k - \gamma_k \boldsymbol{P}_k \boldsymbol{g}_k. \tag{4.3}$$

The preconditioner $\boldsymbol{P}_k$ is updated at each iteration (or at least regularly) to adapt to the local shape of the cost function [84, 85, 86, 87, 88, 89, 90]. This group of methods is typically used in machine learning to solve a linear system [91, 92, 85, 87, 93, 94], but is also popular in image registration [95, 96, 25, 97]. Popular preconditioners, such as Newton or quasi-Newton methods [82, 89], indeed exhibit superior convergence rate compared to the standard gradient descent methods. These improvements, however, come at a cost of the estimation of the inverse Hessian, which alleviates some of the advantages and can even lead to a net deceleration. Zikic *et al.* [89] proposed a diagonal preconditioner for Demons registration. They applied the preconditioner before the dense gradient of the energy function using the inverse of the gradient magnitude. Besides its extra computation efforts at each iteration, its performance mainly depends on the choice of parameter ρ. This parameter is problem specific for different dissimilarity measures, different modalities and different transformation models, which may limit its practicality.

Another group of preconditioning techniques, sometimes called traditional preconditioning, use a static $\boldsymbol{P}$, i.e. the preconditioner $\boldsymbol{P}$ is only calculated once before the start of the optimization [83, 79, 85]. The Krylov subspace method, sparse approximate inverse and Jacobi preconditioning techniques are often used [83]. Klein *et al.* proposed a preconditioner construction method only suitable for mono-modal image registration [70], which approximates the Hessian matrix of the cost function based on an assumption that the intensity difference between moving image and fixed image is zero after a perfect registration. This method is additionally very time-consuming when the number of transformation parameters and image size increase: the required matrix decomposition of the Hessian matrix takes more than 3 hours for $\sim 10^5$ parameters.

As image registration is time crucial for several clinical applications, for example online adaptive radiation therapy [98], it is advantageous to find an efficient way to obtain a search direction $\boldsymbol{d}_k$ and its preconditioner $\boldsymbol{P}$. For registration problems with large degrees of freedom and of large images, it is not very efficient to calculate the search direction in a deterministic way [25] (i.e. using all voxels to compute the gradient). Klein *et al.* proposed a stochastic gradient descent method for image registration, which approximates the gradient by only using a random subset of the image samples [15]. This approximation is much more efficient to compute, thereby outperforming deterministic gradient descent and even quasi-Newton methods [25]. For ill-conditioned problems, however, SGD does not provide a solution and would still suffer from a deteriorated convergence rate.

In this chapter, we consider the preconditioned stochastic gradient descent method (PSGD) that calculates the preconditioner only once. Based on a connection between the incremental displacement of a voxel and the gradient change between iterations, we propose an efficient method to construct a diagonal preconditioner for stochastic gradient descent methods. The chapter is organized as follows. The background and proposed method are given in Section 4.2 and Section 4.3. The dataset used to evaluate the proposed method is described in Section 4.4. This is followed by the experimental setup in Section 4.5 and the results in Section 4.6. The discussion and conclusion are given in Section 4.7 and Section 4.8.

4.2 Background

4.2.1 Preconditioned stochastic gradient descent

The preconditioned stochastic gradient descent method is established as:

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}_k - \gamma_k \boldsymbol{P} \tilde{\boldsymbol{g}}_k, \tag{4.4}$$

where γ_k is the step size, $\tilde{\boldsymbol{g}}_k$ is a stochastic gradient evaluated on a random subset of the image samples Ω_F^s and $\boldsymbol{P}$ is a positive definite $N_P \times N_P$ matrix, with N_P the number of parameters that model the transformation, i.e. $|\boldsymbol{\mu}|$. When $\boldsymbol{P} = \boldsymbol{I}$, PSGD will be reduced to the standard SGD method.

The convergence of PSGD is guaranteed when (1) $\boldsymbol{P}$ is positive definite; and (2) the step size sequence is a non-increasing and non-zero sequence with $\sum_{k=1}^{\infty} \gamma_k = \infty$ and $\sum_{k=1}^{\infty} \gamma_k^2 < \infty$ [99, 86, 85]. The step size sequence used here is defined as follows

[70]:

$$\begin{aligned}\gamma_k &= \frac{\eta}{(t_k+1)/A+1},\\ t_k &= \max(0, t_{k-1} + \mathrm{sigmoid}(-\tilde{\boldsymbol{g}}_{k-1}^T \boldsymbol{P} \tilde{\boldsymbol{g}}_{k-2})),\end{aligned} \tag{4.5}$$

in which η is a noise compensation factor and A controls the decay speed of the step size sequence and is typically set to 20. The noise introduced by the stochastic procedure will influence the convergence rate, so inspired from [70, 100] we use the following compensation factor:

$$\eta = \frac{E\|\boldsymbol{g}^T \boldsymbol{P} \boldsymbol{g}\|}{E\|\tilde{\boldsymbol{g}}^T \boldsymbol{P} \tilde{\boldsymbol{g}}\|} = \frac{E\|\boldsymbol{g}^T \boldsymbol{P} \boldsymbol{g}\|}{E\|\boldsymbol{g}^T \boldsymbol{P} \boldsymbol{g}\| + E\|\boldsymbol{\epsilon}^T \boldsymbol{P} \boldsymbol{\epsilon}\|}, \tag{4.6}$$

in which $\boldsymbol{g}$ is the exact gradient evaluated on all voxels in the image, $\boldsymbol{\epsilon}$ the random noise added to the exact gradient and $E\|\cdot\|$ is the expectation of the norm.

4.2.2 Related work

There are two related methods to estimate a preconditioner:

1. Hessian-type preconditioner (PSGD-H). The theoretical optimal choice for the preconditioner is the inverse Hessian at the optimal parameter $\widehat{\boldsymbol{\mu}}$. However, it is impossible to obtain the exact inverse Hessian beforehand because $\widehat{\boldsymbol{\mu}}$ is unknown [70]. Based on the assumption that the moving image is the same as the fixed image after successful registration: $\boldsymbol{F} \approx \boldsymbol{M}(\boldsymbol{T}(\boldsymbol{x};\widehat{\boldsymbol{\mu}}))$, and the assumption that the deformation is small: $\partial \boldsymbol{T}/\partial \boldsymbol{\mu} \approx \boldsymbol{I}$, Klein *et al.* proposed a method to approximate the Hessian-type preconditioner [70]. This method requires an implementation to calculate a Hessian matrix and a decomposition to construct the preconditioner. This method is only suitable for mono-modal image registration. Moreover, the computation time of this preconditioner is very long when solving large scale problems, which defeats the improvements in the convergence.

2. Jacobi-type preconditioner (PSGD-J). For rigid and affine registration problems, Klein *et al.* [70] assumed that the rotation parameters were scaled by the average voxel displacement caused by a small perturbation of the rotation angle, and proposed a method to construct a diagonal Jacobi-type preconditioner for PSGD. The elements p_i of the diagonal preconditioner $\boldsymbol{P}$ are calculated as follows:

$$p_i = \left(\int_{\Omega_F} \left\| \frac{\partial \boldsymbol{T}}{\partial \theta_i}(\boldsymbol{x};\boldsymbol{\mu}_0) \right\|^2 \mathrm{d}x \Big/ \int_{\Omega_F} \mathrm{d}x \right)^{-\frac{1}{2}}. \tag{4.7}$$

   This method can be used for multi-modal image registration, however, it was proposed for rigid and affine registration only.

4.3 Method

4.3.1 Preliminaries

The aim of the preconditioner $\boldsymbol{P}$ is to scale the parameter space to make ill-posed cost functions easier to optimize. The ideal preconditioner should take care of the relative scaling between the parameters. Construction of a suitable preconditioner is a challenge for a given problem. First, different transformation models and different dissimilarity measures result in different characteristic of the cost function, making the determination of a preconditioner problem-specific. Second, the computation of the preconditioner should be efficient performance-wise, otherwise the overhead of the preconditioner computation will defeat the advantage in runtime reductions obtained from the improvements of the convergence rate.

To find a suitable approximation of $\boldsymbol{P}$ in a computationally efficient way, and robust for different cost functions, a diagonal preconditioner $\boldsymbol{P} = \mathrm{diag}(\boldsymbol{p})$, with $\boldsymbol{p} = (p_1, \ldots, p_N)$, is preferred. In this chapter, we propose a novel way to construct this diagonal preconditioner, which is suitable for both stochastic and deterministic optimization and can be used for mono-modal as well as multi-modal cost functions, in combination with different transformation models like the rigid, affine and B-spline model.

The intuition of the proposed preconditioner is that a gradient change will result in incremental voxel displacements, which is inspired by [15, 100]. In the following we will derive the i-th entry p_i of the preconditioner corresponding to the i-th entry of the transformation parameters $\boldsymbol{\mu}$, such that the displacement induced by a change in that parameter is equal to a predefined value δ. The incremental displacement of a voxel $\boldsymbol{x}_j$ in the fixed image domain Ω_F between iteration k and $k+1$ for an iterative optimization scheme is defined as:

$$\boldsymbol{d}_k(\boldsymbol{x}_j) = \boldsymbol{T}\left(\boldsymbol{x}_j, \boldsymbol{\mu}_{k+1}\right) - \boldsymbol{T}\left(\boldsymbol{x}_j, \boldsymbol{\mu}_k\right), \quad \forall \boldsymbol{x}_j \in \Omega_F. \tag{4.8}$$

We approximate the incremental displacement $\boldsymbol{d}_k$ using the first-order Taylor expansion around $\boldsymbol{\mu}_k$:

$$\begin{aligned} \boldsymbol{d}_k(\boldsymbol{x}_j) &\approx \frac{\partial \boldsymbol{T}}{\partial \boldsymbol{\mu}}\left(\boldsymbol{x}_j, \boldsymbol{\mu}_k\right) \cdot \left(\boldsymbol{\mu}_{k+1} - \boldsymbol{\mu}_k\right) \\ &= \boldsymbol{J}(\boldsymbol{x}_j) \cdot \left(\boldsymbol{\mu}_{k+1} - \boldsymbol{\mu}_k\right), \end{aligned} \tag{4.9}$$

in which $\boldsymbol{J}(\boldsymbol{x}_j) = \frac{\partial \boldsymbol{T}}{\partial \boldsymbol{\mu}}\left(\boldsymbol{x}_j, \boldsymbol{\mu}_k\right)$ is the Jacobian matrix of size $d \times N_P$. Using the optimization scheme (4.4), we obtain $\boldsymbol{\mu}_{k+1} - \boldsymbol{\mu}_k = -\gamma_k \boldsymbol{P} \boldsymbol{g}_k$, and we can rewrite $\boldsymbol{d}_{k+1}$ as:

$$\boldsymbol{d}_k(\boldsymbol{x}_j) \approx -\gamma_k \boldsymbol{J}(\boldsymbol{x}_j) \boldsymbol{P} \boldsymbol{g}_k. \tag{4.10}$$

4.3.2 Diagonal preconditioner estimation

From Equation (4.10), we notice that the preconditioner matrix $\boldsymbol{P}$ can be estimated prior to registration, i.e. at iteration $k = 0$. After choosing $\gamma_0 = 1$, we obtain $\boldsymbol{d}_1(\boldsymbol{x}_j) \approx -\boldsymbol{J}(\boldsymbol{x}_j)\mathrm{diag}(\boldsymbol{p})\boldsymbol{g}_0$. In the remainder of the chapter, we use the notation $\boldsymbol{d}$ and $\boldsymbol{g}$ for simplification.

The Jacobi-type preconditioner from Equation (4.7) can be rewritten to:

$$p_i = \left(E \|\boldsymbol{J}^i(\boldsymbol{x}_j)\|^2\right)^{-1/2}, \tag{4.11}$$

where $\boldsymbol{J}^i(\boldsymbol{x}_j)$ denotes the i-th column of the Jacobian matrix. Inspired by Equation (4.11) we inspect the displacement $\|\boldsymbol{d}^i\|$ that is induced by a change $\triangle \boldsymbol{\mu}_i$ in the i-th transformation parameter, i.e. the displacement generated by g^i only:

$$\|\boldsymbol{d}^i(\boldsymbol{x}_j)\| \approx \left\| -\boldsymbol{J}^i(\boldsymbol{x}_j) p_i g^i \right\| = p_i \left\| -\boldsymbol{J}^i(\boldsymbol{x}_j) g^i \right\|, \tag{4.12}$$

Algorithm 2 Proposed preconditioner estimation

```
Require: $N_s$ the number of samples, $\delta$ the maximum allowed voxel displacement, $\tau$
    the regularization factor, $\kappa_{\max}$ the maximum condition number
Compute the gradient $\boldsymbol{g}$ of size $N_P$
Randomly take $N_s$ samples $\{\boldsymbol{x}_j\}$ from the fixed image
$\boldsymbol{p} = \boldsymbol{I}$, $\boldsymbol{t} = \boldsymbol{0}$, $\boldsymbol{z} = \boldsymbol{0}$, $\boldsymbol{y} = \boldsymbol{0}$        ▷ initialization
for $j = 1, 2, \ldots, N_s$ do                          ▷ loop over the samples $\boldsymbol{x}_j$
    Calculate the Jacobian $\boldsymbol{J}(\boldsymbol{x}_j)$
    for $i = 1, 2, \ldots, N_P$ do                      ▷ loop over the parameters
        $s_i = \|\boldsymbol{J}^i(\boldsymbol{x}_j) g^i\|$
        Regularize $s_i$ with $\tau$ using Section 4.3.3
        $z_i = z_i + s_i$                               ▷ update for the mean
        $y_i = y_i + s_i^2$                             ▷ update for the variance
        $t_i = t_i + 1$                                 ▷ increase counter
for $i = 1, 2, \ldots, N_P$ do                          ▷ loop over the parameters
    $q_i = z_i / t_i + 2\sqrt{(y_i / t_i) - (z_i / t_i)^2}$
    $p_i = \delta / q_i$
    Constrain the condition number of $p_i$ using $\kappa_{\max}$ (see Section 4.3.4)
Return $\boldsymbol{p}$
```

in which $\|\cdot\|$ used in this chapter is the ℓ_1 norm. In medical image registration, we expect a continuous and homogenous transformation and moreover assume that the voxel displacement $\boldsymbol{d}^i$ is to be not larger than δ: i.e $\|\boldsymbol{d}^i(\boldsymbol{x}_j)\| \le \delta, \quad \forall \boldsymbol{x}_j \in \Omega_F$. Based on the distribution of the voxel displacements, there is a weakened form for this assumption: $P(\|\boldsymbol{d}^i(\boldsymbol{x}_j)\| > \delta) < \rho$, where ρ is a small probability value often 0.05. According to the Vysochanskij-Petunin inequality [46], we have the following expression:

$$E\|\boldsymbol{d}^i(\boldsymbol{x}_j)\| + 2\sqrt{Var\|\boldsymbol{d}^i(\boldsymbol{x}_j)\|} \le \delta, \quad \forall \boldsymbol{x}_j \in \Omega_F. \tag{4.13}$$

Combined with Equation (4.12), we obtain the relationship between the i-th entry p_i of the preconditioner and the maximum voxel displacement as follows:

$$p_i \left(E \left\| s_i(\boldsymbol{x}_j) \right\| + 2\sqrt{Var \left\| s_i(\boldsymbol{x}_j) \right\|} \right) \le \delta, \tag{4.14}$$

where $s_i(\boldsymbol{x}_j) = \| - \boldsymbol{J}^i(\boldsymbol{x}_j) g^i \|$. The i-th entry of the preconditioner is then defined as:

$$p_i = \frac{\delta}{E \left\| s_i(\boldsymbol{x}_j) \right\| + 2\sqrt{Var \left\| s_i(\boldsymbol{x}_j) \right\|}}. \tag{4.15}$$

Finally, the full preconditioner $\boldsymbol{P}$ is obtained by repeating the above procedure for each p_i. The procedure is sketched in Algorithm 2.

4.3.3 Regularization

The assumption used to approximate a preconditioner, that all transformation parameters should independently induce a maximum voxel displacement δ, may be too strict or too sensitive to noise in the measurements. For the B-spline transformation, fore example, this assumption forces all regions to have a displacement δ, even regions that do not require registration. Noise could come from an insufficient number of samples $\boldsymbol{x}_j$ used for the estimation, or from inexact evaluation of the gradient. This could result in differences in the estimated entries of the preconditioner that are expected to have similar value. For the B-spline transformation model one would expect that nearby control points would be scaled similarly, without sudden sharp transitions. For the affine transformation on the other hand, one would expect that scalings related to translation parameters are more similar than those related to rotational parameters.

We therefore propose to optionally regularize the procedure from Section 4.3.2, such that the i-th entry p_i of the preconditioner is not treated completely independent, but also takes into account the estimates of the related parameters. Related parameters are those jointly affected by a voxel $\boldsymbol{x}_j$ (for an affine transformation these are all parameters; for the B-spline only parameters in the compact support region of $\boldsymbol{x}_j$), and secondly by their similarity in Jacobian contribution (for the affine transformation, intuitively rotations and translations are to be treated separately). The proposed regularization procedure is as follows:

$$s_i(\boldsymbol{x}_j) = \tau \cdot s_i(\boldsymbol{x}_j) + \underbrace{(1-\tau) \cdot \frac{1}{\sum \omega_m} \sum_{m \neq i} s_m(\boldsymbol{x}_j) \cdot \omega_m}_{\text{regularization term}}, \tag{4.16}$$

where ω_m weighs the contributions of similar parameters and τ balances the contribution of entry i with the contributions of the other parameters. The weights ω_m are calculated using a Gaussian function:

$$\omega_m = \exp\left(- \frac{(\|\boldsymbol{J}^i(\boldsymbol{x}_j)\| - \|\boldsymbol{J}^m(\boldsymbol{x}_j)\|)^2}{2\sigma^2} \right), \tag{4.17}$$

in which σ is chosen as $\min(\|\boldsymbol{J}^i(\boldsymbol{x}_j)\| - \|\boldsymbol{J}^m(\boldsymbol{x}_j)\|)/\max(\|\boldsymbol{J}^i(\boldsymbol{x}_j)\| - \|\boldsymbol{J}^m(\boldsymbol{x}_j)\|)$, $\forall m \neq i$.

While for the B-spline transformation model such a choice would also be valid, a simplification is possible. For the B-spline model the displacement of a voxel is only determined by the control points in its support region. Furthermore, we expect the influence on the displacement to be almost equal for each control point in the support region. We therefore assume for the B-spline model that the weights $\omega_m = 1$, simplifying Equation (4.16) to $s_i(\boldsymbol{x}_j) = \tau \times s_i(\boldsymbol{x}_j) + (1-\tau) \cdot \|\sum(\boldsymbol{J}^i(\boldsymbol{x}_j) g^i)\|$.

4.3.4 Condition number

Even if the resulting preconditioner is symmetric and positive definite, it could be ill-conditioned, especially for nonrigid image registration problems. The convergence rate of the algorithm can be measured by the so-called condition number:

$$\kappa = \lambda_{\max}/\lambda_{\min}, \tag{4.18}$$

where $\lambda_{\max}$ and $\lambda_{\min}$ are the largest and smallest eigenvalue of $\boldsymbol{P}$, respectively. It is common to constrain the eigenvalues, such that the condition number will be closer to 1 [70, 88]. We introduce a user-defined maximum condition number $\kappa_{\max}$ for this purpose.

Define a diagonal eigenvalue matrix $\Lambda = diag(\lambda_1, \ldots, \lambda_{N_p})$ for the preconditioner $\boldsymbol{P}$. In this study, as our preconditioner $\boldsymbol{P}$ is diagonal, the entries of $\boldsymbol{P}$ are equal to the eigenvalues of Λ: $p_i = \lambda_i, \forall i$. To constrain the eigenvalues, we replace small eigenvalues of $\boldsymbol{P}$ that make $\kappa > \kappa_{\max}$ using the following equation:

$$p_i = \begin{cases} \lambda_{\max}/\kappa_{\max}, & \text{if } \lambda_{\max}/\lambda_i > \kappa_{\max}, \\ \lambda_i, & \text{otherwise.} \end{cases} \tag{4.19}$$

The thus constrained matrix constitutes then the finally proposed static preconditioner. Combined with Equation (4) this defines the Fast Preconditioned Stochastic Gradient Descent method (FPSGD).

4.4 Data sets

The proposed FPSGD method is tested on mono-modal as well as multi-modal data. An overview of the used data sets is presented in Table 4.1.

4.4.1 Mono-modal lung data: SPREAD

3D lung Computed Tomography (CT) images of 19 patients were acquired during the SPREAD study [77]. A follow-up scan was acquired for each patient after the baseline scan with image sizes around $450 \times 300 \times 150$ and voxel sizes around $0.7 \times 0.7 \times 2.5$ mm. The ground truth consists of 100 anatomical corresponding points, which were semi-automatically extracted using Murphy's method [49]. The algorithm first automatically selects 100 evenly distributed landmarks at characteristic locations in the baseline image, and then predicts the corresponding points in the follow-up image. The corresponding points are then inspected and corrected by two experts using a graphical user interface [50].

4.4.2 Multi-modal brain data: RIRE and BrainWeb

Two multi-modal datasets are used to evaluate the performance of the proposed method.

4.4.2.1 RIRE brain data

This brain dataset was acquired during the Retrospective Image Registration Evaluation (RIRE) project. CT scans and Magnetic Resonance Imaging (MRI-T1) are available for 9 patients. The CT images have sizes of $512 \times 512 \times 50$ with voxel sizes of $0.45 \times 0.45 \times 3$ mm, while the MRI-T1 image is of size $256 \times 256 \times 50$ with voxel sizes of $0.85 \times 0.85 \times 3$ mm. Fiducial markers were implanted in each patient and served as a ground truth [47]. These markers were manually erased from the images and replaced with a simulated background pattern.

4.4.2.2 BrainWeb simulated brain data

T1 and T2 weighted 3D brain MR images were created using the Simulated Brain Database from BrainWeb [101]. To generate brain image pairs, default settings provided by BrainWeb were used with 3% noise and 20% intensity non-uniformity. The brain images are of sizes $181 \times 217 \times 181$ and a voxel spacing of 1 mm isotropically. A mask of the brain was extracted from the T1 image by FSL-BET [102] and the same mask was used for the T2 image. 100 randomly generated displacement vector fields (DVFs) serve as the ground truth deformation fields. The DVFs are isotropically

Table 4.1: Overview of data sets and experimental setup.

	Mono-modal	Multi-modal	
Dataset	SPREAD	RIRE	BrainWeb
Anatomy	Lung	Brain	Brain
Modality	CT	CT and 1.5T MR T1	MR T1 and T2
Dimensions	450 × 300 × 130	CT: 512 × 512 × 50 MR: 256 × 256 × 50	181 × 217 × 181
Voxel size (mm)	~ 0.7 × 0.7 × 2.5	CT: 0.45 × 0.45 × 3 MR: 0.85 × 0.85 × 3	1 × 1 × 1
Number of patients	21	9	1 × 100
Similarity measure	MSD	MI	MI
Transformation	Affine, B-spline	Rigid	B-spline
B-spline control point grid spacing (mm)	10 × 10 × 10	-	10 × 10 × 10
Number of parameters (last resolution)	~ 90k	6	36300
Ground truth	100 corresponding points	8 corner points	100 simulated deformations
Evaluation measure	Euclidean distance	Euclidean distance	Residuals

generated in three dimensions within the brain mask and the maximum magnitude of DVFs is chosen as 8 and 15 mm. These DVFs are then smoothed by a Gaussian filter with a standard deviation between 10 and 30 mm.

4.5 Experiments

In this section, experimental settings are given to test the performance of the proposed method. The proposed FPSGD method is compared with the following methods:

1. Fast adaptive stochastic gradient descent (FASGD) [100], which is a state-of-the-art first order stochastic optimization method that does not use preconditioning. For rigid and affine registration, the diagonal of the preconditioner $\boldsymbol{P}$ is chosen as 1 for the translational parameters and 1/100000 for the others. This reflects that the parameters $\boldsymbol{\mu}$ corresponding to rotation have in general a much smaller range than parameters corresponding to translation.

2. Jacobi-type preconditioner (PSGD-J) [70], where a diagonal preconditioner is chosen according to Equation (4.7). This method was only proposed for rigid and affine registration.

3. Hessian-type preconditioner (PSGD-H) [70], see Section 4.2.2. This preconditioner is only suitable for mono-modal registration, and therefore only implemented for the mean squared intensity difference (MSD) dissimilarity measure.

All these methods, including the proposed method, were implemented in C++ and are available as open source software via the `elastix` package [10]. All experiments were performed on a workstation with an Ubuntu Linux OS, which has 8 cores running at 2.4 GHz and 24 GB of memory. Detailed settings are presented in Section 4.5.3 and 4.5.4, and an overview of the experimental setup is given in Table 4.1.

4.5.1 Experimental setup

To validate the generality of the proposed preconditioner, the experiments are performed on mono-modal as well as multi-modal image registration. For each group, different transformation models are used, namely the rigid, affine and B-spline transformation models [10]. For rigid and affine image registration, only one resolution of 500 iterations is used, to be able to more easily compare convergence properties. For B-spline image registration, a three-level multi-resolution framework is used on the SPREAD data with a standard deviation of the Gaussian smoothing filter of 2, 1 and 0.5 mm, and 500 iterations for each resolution. For the BrainWeb data, we used only one resolution of 1000 iterations.

The number of samples used for computing $\tilde{\boldsymbol{g}}$ was the same for all methods and set to 5000 [100]. Different methods used different number of samples for the preconditioner estimation. For FPSGD 50000 samples were used at each resolution. For PSGD-H the number of samples were 100000, 100000 and 500000 for the three resolutions, respectively. For PSGD-J, 1000 samples were used. To estimate the step size of FASGD, the number of samples was chosen equal to the number of transformation parameters $\|\boldsymbol{\mu}\|$ at each resolution, for instance in the SPREAD experiment around 4000, 15000 and 90000 samples for the three resolutions, respectively. The user pre-defined value δ for FASGD and FPSGD is chosen as the mean length of the voxel size. $A = 20$ is used for all tested methods.

In Section 4.3, we introduced two free parameters of the proposed FPSGD method: the regularization factor τ and the maximum condition number $\kappa_{\max}$. To assess the influence of these two parameters on the results, we first vary the regularization factor τ while using a fixed $\kappa_{\max}$, and then vice versa. The regularization factor τ was selected between 0 and 1, using increments of 0.2, so there were 6 variations. For these tests, $\kappa_{\max} = 2$ was chosen for the B-spline registration, while for rigid and affine registration no restriction is needed on the condition number, i.e. $\kappa_{\max} = \infty$. In the second group of tests, a fixed $\tau = 0.6$ was chosen and $\kappa_{\max} \in \{1, 2, 4, 8, 16\}$ were tested for the B-spline registrations of the SPREAD data and the BrainWeb data. The results are reported in Section 4.6.1.

4.5.2 Convergence and runtime performance

The performance of the tested methods is first evaluated in terms of the convergence rate and the resulting speed-up in runtime. To measure the convergence rate, the dissimilarity measure (MSD or MI) was calculated at each 5th iteration. This calculation was performed deterministically using all samples from the fixed image. FASGD is chosen as the baseline method and we compare the exact cost function value of all other methods against the exact cost function value of FASGD at its final solution $\widehat{\boldsymbol{\mu}}_{\mathrm{ref}}$. For each method, we counted the number of iterations I required to obtain a cost function value that is equal to or smaller than that of the baseline method using $\mathscr{C}(\boldsymbol{\mu}_k) \leq \mathscr{C}(\widehat{\boldsymbol{\mu}}_{\mathrm{ref}})$ for the first time.

To assess runtime performance, several computations are timed and recorded: the time t_{est} it takes to estimate the preconditioner $\boldsymbol{P}$ and the time t_{iter} each iteration takes. When I equals the number of iterations needed for reaching the same cost function value as FASGD, then the pure registration time is defined as $t_{\mathrm{pure}} = t_{\mathrm{iter}} \cdot I$. The total registration time is then $t_{\mathrm{total}} = t_{\mathrm{est}} + t_{\mathrm{pure}}$. The time t_{est} consists of the time to estimate the preconditioner and/or the step size γ_0 for the different methods, i.e. for FASGD t_{est} is the estimation time of the step size, for PSGD-J and PSGD-H both are included and for FPSGD t_{est} is the estimation time of the preconditioner.

4.5.3 Mono-modal image registration: SPREAD

In this experiment we compare the proposed method compared to all three alternative methods: FASGD, PSGD-H and PSGD-J. The baseline and follow-up image were treated as fixed image and moving image, respectively. The Euclidean distance of the 100 corresponding points is computed to evaluate the registration accuracy using $\mathrm{ED} = \frac{1}{100}\sum_{i=1}^{100}\|\boldsymbol{T}_{\widehat{\boldsymbol{\mu}}}(\boldsymbol{p}_F^i) - \boldsymbol{p}_M^i\|$, with $\boldsymbol{p}_F$ and $\boldsymbol{p}_M$ the corresponding points, and $\boldsymbol{T}$ the transformation at iteration I. A Wilcoxon signed-rank test on the registration accuracy is used to evaluate statistical differences of these methods compared to FASGD method.

We use the mean squared intensity difference (MSD) as a dissimilarity measure, and test for affine as well as B-spline transformations.

4.5.4 Multi-modal image registration: RIRE and BrainWeb

For multi-modal image registration, real clinical brain data is used for rigid registration and simulated brain data is used for nonrigid registration. These datasets are used to compare the performance of FPSGD with FASGD and PSGD-J, as PSGD-H is not suitable for multi-modal registration.

4.5.4.1 RIRE brain data

We registered the MR T1 image (moving image) to the CT image (fixed image) using the rigid transformation model and mutual information (MI) dissimilarity measure. The registration accuracy is evaluated using $\mathrm{ED} = \frac{1}{8}\sum_{i=1}^{8} \| \boldsymbol{T}_{\hat{\boldsymbol{\mu}}}(\boldsymbol{p}_F^i) - \boldsymbol{p}_M^i \|$, with $\boldsymbol{p}_F$ and $\boldsymbol{p}_M$ the corner points defined by RIRE and annotated in the fixed and moving image, respectively.

4.5.4.2 BrainWeb simulated brain data

Pairwise B-spline registration was performed using these randomly generated DVFs as the initial transformation $\boldsymbol{T}_{\mathrm{init}}$. The registration accuracy is evaluated using the average residual deformation inside the brain mask Ω_F [103]:

$$Residual(\boldsymbol{T}_{\mathrm{init}}, \boldsymbol{T}_{\hat{\boldsymbol{\mu}}}) = \frac{1}{|\Omega_F|} \sum_{\boldsymbol{x}_i \in \Omega_F} \| \boldsymbol{T}_{\hat{\boldsymbol{\mu}}}(\boldsymbol{T}_{\mathrm{init}}(\boldsymbol{x}_i) - \boldsymbol{x}_i) \|. \tag{4.20}$$

The statistical differences of FASGD and PSGD-J compared to FASGD method were evaluated using a Wilcoxon signed-rank test on the registration accuracy.

4.6 Results

4.6.1 Parameter sensitivity analysis

4.6.1.1 Selection of the regularization factor τ

For all datasets we varied the parameter τ. The results can be found in Table 4.2, Table 4.3, Table 4.4, and Table 4.5. It can be seen that the regularization factor $\tau = 1.0$ (no regularization) gave the worst performance for rigid and affine registration on all datasets. For B-spline registration $\tau = 1.0$ did work for the SPREAD data, but failed again on the BrainWeb data. Setting the regularization factor $\tau = 0.0$ is another extreme meaning that the regularization term completely determines the estimation of the preconditioner. From the results in the tables, it can be seen that the convergence rate is much slower than for other choices of τ, even though the registration accuracy is almost similar.

The experimental results on the different datasets show that there is no statistical difference between the different choices of τ $(0.0 < \tau < 1.0)$ regarding the accuracy. However, the convergence rate is improved when taking a larger value of τ. We therefore conclude that a regularization factor τ between 0.6 and 0.8 gives the best results. In the remainder of the chapter we use $\tau = 0.6$.

4.6.1.2 Influence of the condition number $\kappa_{\max}$

The maximum condition number $\kappa_{\max}$ is especially important for non-rigid registration. Table 4.6 presents the registration accuracy with respect to $\kappa_{\max}$ for the SPREAD study. As we can see, different $\kappa_{\max}$ obtained the similar accuracy. However, less iterations were needed for a larger $\kappa_{\max}$. From the convergence plot in Figure 4.1, it can be observed that the optimization converged faster for $\kappa_{\max} \geq 2$. However, for $\kappa_{\max} \geq 8$, the plot exhibits more oscillating behavior, suggesting a less stable optimization.

For the BrainWeb data in Table 4.7, we again see that registration accuracy is similar for different $\kappa_{\max}$. In Figure 4.2, all choices of $\kappa_{\max}$ converged faster than FASGD, while for $\kappa_{\max} \geq 2$ the convergence rate does not improve further. From Table 4.7 and Figure 4.2, we can see that $\kappa_{\max} = 2$ or 4 gave the best results, which is

Table 4.2: Overall results of affine registration for the SPREAD lung CT data. We used the MSD dissimilarity measure, 3 resolutions, 500 iterations and $\kappa_{\max} = \infty$.

Optimizer	Iterations I avg ± std	t_{est} (s) avg ± std	t_{pure} (s) avg ± std	t_{total} (s) avg ± std	Speed-up avg ± std	ED (mm) avg ± std	p-value
FASGD	489 ± 12	0.11 ± 0.03	1.83 ± 0.19	1.95 ± 0.21	-	4.99 ± 3.42	-
PSGD-J	153 ± 75	0.11 ± 0.02	0.55 ± 0.28	0.66 ± 0.29	3.5 ± 1.6	5.51 ± 3.83	0.036
PSGD-H	31 ± 24	4.14 ± 0.76	0.11 ± 0.09	4.25 ± 0.79	0.5 ± 0.1	4.75 ± 3.21	0.049
FPSGD $\tau = 0.0$	80 ± 92	0.25 ± 0.02	0.29 ± 0.34	0.55 ± 0.34	4.3 ± 1.4	5.02 ± 3.64	0.520
FPSGD $\tau = 0.2$	52 ± 54	0.25 ± 0.02	0.19 ± 0.20	0.43 ± 0.21	5.0 ± 1.3	5.04 ± 3.72	0.260
FPSGD $\tau = 0.4$	39 ± 23	0.25 ± 0.02	0.14 ± 0.08	0.39 ± 0.09	5.2 ± 1.0	5.08 ± 3.60	1.000
FPSGD $\tau = 0.6$	41 ± 21	0.25 ± 0.02	0.15 ± 0.08	0.40 ± 0.09	5.1 ± 1.2	5.14 ± 3.53	0.355
FPSGD $\tau = 0.8$	50 ± 43	0.26 ± 0.03	0.18 ± 0.16	0.44 ± 0.17	4.9 ± 1.5	5.05 ± 3.36	0.658
FPSGD $\tau = 1.0$	109 ± 141	0.25 ± 0.03	0.40 ± 0.53	0.65 ± 0.53	4.2 ± 1.7	4.99 ± 3.24	0.469

Table 4.3: Overall results of B-spline registration for the SPREAD lung CT data. We used the MSD dissimilarity measure, 3 resolutions, 500 iterations and $\kappa_{\max} = 4$.

	Optimizer	Iterations I avg ± std	t_{est} (s) avg ± std	t_{pure} (s) avg ± std	t_{total} (s) avg ± std	Speed-up avg ± std	ED (mm) avg ± std	p-value
Resolution 1	FASGD	496 ± 0	0.92 ± 0.12	11.0 ± 0.16	11.9 ± 0.22	-		
	PSGD-H	23 ± 6	24.2 ± 4.14	0.73 ± 0.22	25.0 ± 4.19	0.5 ± 0.1		
	FPSGD $\tau = 0.0$	414 ± 61	1.15 ± 0.14	9.15 ± 1.29	10.3 ± 1.33	1.2 ± 0.1		
	FPSGD $\tau = 0.2$	358 ± 75	1.14 ± 0.13	7.90 ± 1.72	9.03 ± 1.78	1.4 ± 0.2		
	FPSGD $\tau = 0.4$	294 ± 66	1.15 ± 0.13	6.50 ± 1.49	7.66 ± 1.53	1.6 ± 0.3		
	FPSGD $\tau = 0.6$	225 ± 50	1.13 ± 0.13	4.97 ± 1.10	6.10 ± 1.15	2.0 ± 0.4		
	FPSGD $\tau = 0.8$	157 ± 37	1.14 ± 0.14	3.47 ± 0.85	4.61 ± 0.93	2.7 ± 0.5		
	FPSGD $\tau = 1.0$	97 ± 37	1.13 ± 0.13	2.14 ± 0.90	3.27 ± 0.98	3.8 ± 0.8		
Resolution 2	FASGD	496 ± 0	0.98 ± 0.13	11.3 ± 0.13	12.2 ± 0.21	-		
	PSGD-H	9 ± 15	174 ± 80	1.36 ± 2.47	175 ± 80	0.1 ± 0.0		
	FPSGD $\tau = 0.0$	306 ± 101	1.38 ± 0.16	6.99 ± 2.25	8.36 ± 2.29	1.6 ± 0.5		
	FPSGD $\tau = 0.2$	258 ± 88	1.38 ± 0.19	5.87 ± 1.99	7.25 ± 2.05	1.8 ± 0.5		
	FPSGD $\tau = 0.4$	205 ± 79	1.36 ± 0.16	4.67 ± 1.76	6.03 ± 1.78	2.2 ± 0.6		
	FPSGD $\tau = 0.6$	145 ± 63	1.37 ± 0.16	3.32 ± 1.46	4.68 ± 1.46	2.8 ± 0.8		
	FPSGD $\tau = 0.8$	103 ± 47	1.38 ± 0.17	2.36 ± 1.09	3.75 ± 1.11	3.5 ± 1.0		
	FPSGD $\tau = 1.0$	116 ± 103	1.36 ± 0.17	2.64 ± 2.30	4.00 ± 2.33	3.6 ± 1.3		
Resolution 3	FASGD	491 ± 17	1.33 ± 0.18	12.0 ± 0.43	13.3 ± 0.47	-	1.67 ± 1.68	-
	PSGD-H	11 ± 22	9959 ± 7269	22.2 ± 45.3	9981 ± 7265	0.0 ± 0.0	1.59 ± 1.59	0.445
	FPSGD $\tau = 0.0$	398 ± 94	2.52 ± 0.36	9.78 ± 2.35	12.3 ± 2.38	1.1 ± 0.3	1.64 ± 1.65	0.243
	FPSGD $\tau = 0.2$	319 ± 101	2.52 ± 0.38	7.85 ± 2.44	10.4 ± 2.40	1.4 ± 0.3	1.64 ± 1.64	0.295
	FPSGD $\tau = 0.4$	260 ± 95	2.52 ± 0.36	6.42 ± 2.37	8.94 ± 2.31	1.6 ± 0.4	1.62 ± 1.61	0.049
	FPSGD $\tau = 0.6$	215 ± 76	2.53 ± 0.36	5.28 ± 1.86	7.81 ± 1.82	1.8 ± 0.5	1.58 ± 1.56	0.007
	FPSGD $\tau = 0.8$	202 ± 117	2.52 ± 0.37	4.99 ± 2.87	7.51 ± 2.82	2.1 ± 1.1	1.52 ± 1.48	0.004
	FPSGD $\tau = 1.0$	215 ± 184	2.51 ± 0.37	5.25 ± 4.48	7.76 ± 4.39	2.5 ± 1.6	1.48 ± 1.41	0.006

Table 4.4: Overall results for the RIRE brain dataset. We used the MI dissimilarity measure, 3 resolutions, 500 iterations and $\kappa_{\max} = \infty$.

$\mathscr{C}, \boldsymbol{T}$	Optimizer	Iterations I avg ± std	$t_{\text{est}}(s)$ avg ± std	$t_{\text{pure}}(s)$ avg ± std	t_{total} (s) avg ± std	Speed-up avg ± std	ED (mm) avg ± std	p-value
MI, Rigid	FASGD	485 ± 22	0.26 ± 0.02	2.77 ± 0.16	3.02 ± 0.17	-	5.88 ± 3.64	-
	PSGD-J	65 ± 19	0.26 ± 0.02	0.38 ± 0.11	0.65 ± 0.11	4.8 ± 0.8	4.81 ± 2.88	0.004
	FPSGD $\tau = 0.0$	27 ± 14	0.34 ± 0.02	0.16 ± 0.08	0.49 ± 0.08	6.2 ± 0.9	4.02 ± 2.56	0.004
	FPSGD $\tau = 0.2$	27 ± 14	0.35 ± 0.03	0.17 ± 0.08	0.52 ± 0.08	6.0 ± 1.0	3.95 ± 2.75	0.008
	FPSGD $\tau = 0.4$	32 ± 17	0.34 ± 0.02	0.19 ± 0.10	0.54 ± 0.10	5.8 ± 0.9	4.08 ± 2.93	0.004
	FPSGD $\tau = 0.6$	38 ± 20	0.34 ± 0.03	0.23 ± 0.13	0.57 ± 0.14	5.5 ± 1.2	3.69 ± 2.69	0.004
	FPSGD $\tau = 0.8$	50 ± 29	0.34 ± 0.01	0.30 ± 0.17	0.63 ± 0.16	5.0 ± 1.1	4.16 ± 2.96	0.004
	FPSGD $\tau = 1.0$	117 ± 132	0.34 ± 0.03	0.67 ± 0.76	1.01 ± 0.76	3.9 ± 1.5	8.56 ± 12.3	0.129

Table 4.5: Overall results for the BrainWeb dataset. We used the MI dissimilarity measure, 3 resolutions, 1000 iterations and $\kappa_{\max} = 4$. For $\tau = 1.0$, most registrations failed.

$\mathscr{C}, \boldsymbol{T}$	Optimizer	Iterations I avg ± std	$t_{\text{est}}(s)$ avg ± std	$t_{\text{pure}}(s)$ avg ± std	$t_{\text{total}}(s)$ avg ± std	Speed-up avg ± std	Residuals avg ± std	p-value
MI, BSpline	FASGD	996 ± 0	1.72 ± 0.06	49.3 ± 0.90	51.05 ± 0.90	-	2.48 ± 1.43	-
	FPSGD $\tau = 0.0$	289 ± 130	2.29 ± 0.07	13.8 ± 6.17	16.06 ± 6.16	3.6 ± 1.1	2.48 ± 1.42	0.518
	FPSGD $\tau = 0.2$	234 ± 92	2.29 ± 0.07	11.4 ± 4.52	13.66 ± 4.53	4.1 ± 1.2	2.48 ± 1.43	0.921
	FPSGD $\tau = 0.4$	202 ± 62	2.31 ± 0.07	9.89 ± 2.99	12.20 ± 2.99	4.4 ± 1.0	2.48 ± 1.40	0.264
	FPSGD $\tau = 0.6$	211 ± 115	2.36 ± 0.09	11.0 ± 6.42	13.32 ± 6.43	4.2 ± 1.1	2.50 ± 1.39	0.031
	FPSGD $\tau = 0.8$	354 ± 218	2.41 ± 0.05	19.5 ± 12.0	21.95 ± 12.0	3.0 ± 1.4	2.76 ± 1.35	0.000
	FPSGD $\tau = 1.0$	-	-	-	-	-	-	

Table 4.6: The influence of $\kappa_{\max}$ on B-spline registration for the SPREAD study. We used the MI dissimilarity measure, 3 resolutions, 500 iterations, and $\tau = 0.6$.

Optimizer	Resolution 1 Iterations I avg ± std	Resolution 2 Iterations I avg ± std	Resolution 3 Iterations I avg ± std	ED (mm) avg ± std	p-value
FASGD	496 ± 0	496 ± 0	489 ± 14	1.67 ± 1.68	-
FPSGD $\kappa_{\max} = 1$	440 ± 72	426 ± 82	481 ± 44	1.71 ± 1.70	0.001
FPSGD $\kappa_{\max} = 2$	294 ± 85	292 ± 66	378 ± 120	1.66 ± 1.64	0.968
FPSGD $\kappa_{\max} = 4$	180 ± 87	226 ± 50	241 ± 74	1.58 ± 1.56	0.003
FPSGD $\kappa_{\max} = 8$	149 ± 86	224 ± 68	202 ± 73	1.52 ± 1.49	0.000
FPSGD $\kappa_{\max} = 16$	153 ± 89	222 ± 68	200 ± 74	1.51 ± 1.49	0.001

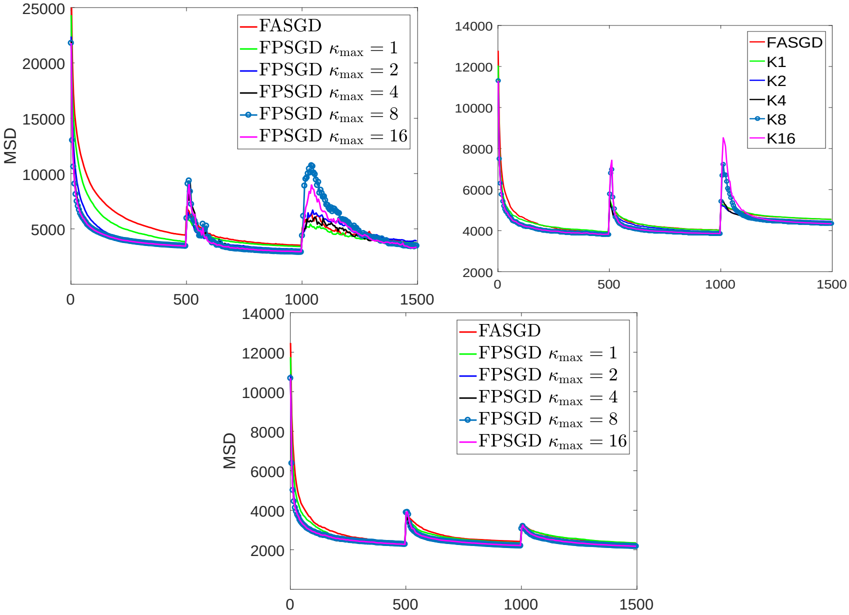


Figure 4.1: Convergence plots for three different patients in the experiments of different $\kappa_{\max}$ for the SPREAD dataset, showing the cost function value (MSD) against the iteration number for different $\kappa_{\max}$ using $\tau = 0.6$.

consistent with the results in [70]. In the remainder of the chapter we set $\kappa_{\max} = 4$ for B-spline registration (and $\kappa_{\max} = \infty$ for rigid and affine registration).

4.6.2 Results of mono-modal image registration

4.6.2.1 Affine registration

The overall results of the experiments on affine registration for the SPREAD lung CT data are given in Table 4.3. It shows that the proposed FPSGD method took less iterations to obtain the same cost function value $\mathscr{C}(\widehat{\boldsymbol{\mu}}_{\mathrm{ref}})$ than FASGD and PSGD-J. The speed-up in terms of number of iterations of FPSGD is about 10. The improvements

Table 4.7: The influence of $\kappa_{\max}$ on B-spline registration for the BrainWeb study. We used the MI dissimilarity measure, 1 resolution, 500 iterations, and $\tau = 0.6$.

Optimizer	Iterations I avg ± std	Speed-up avg ± std	Residuals avg ± std	p-value
FASGD	996 ± 0	1.0 ± 0.0	2.48 ± 1.43	-
FPSGD $\kappa_{\max} = 1$	333 ± 151	3.5 ± 1.3	2.48 ± 1.43	0.561
FPSGD $\kappa_{\max} = 2$	230 ± 85	4.8 ± 1.5	2.49 ± 1.42	0.080
FPSGD $\kappa_{\max} = 4$	211 ± 115	5.4 ± 1.6	2.50 ± 1.39	0.031
FPSGD $\kappa_{\max} = 8$	208 ± 118	5.6 ± 1.9	2.50 ± 1.37	0.017
FPSGD $\kappa_{\max} = 16$	220 ± 128	5.3 ± 1.8	2.53 ± 1.37	0.001

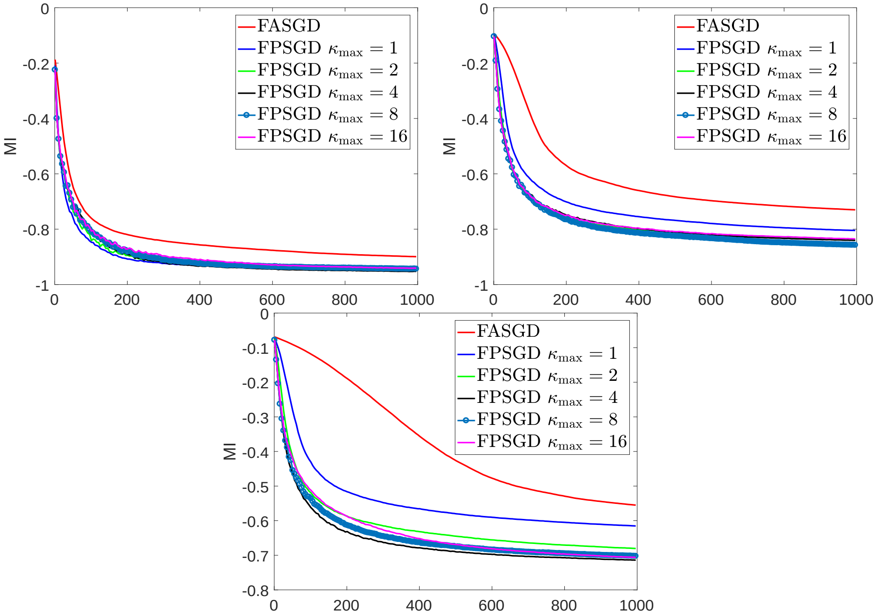


Figure 4.2: Convergence plots for three different patients for the BrainWeb dataset, showing the cost function value (negative MI dissimilarity measure) against the iteration number, using $\tau = 0.6$.

of FPSGD compared to FASGD and PSGD-J in the convergence rate are also shown in Figure 4.5a and Figure 4.5b. These methods have the same runtime per iteration (~3.5 ms). PSGD-H required less iterations than the proposed FPSGD method. The computation of the preconditioner however took somewhat longer, resulting in an overall decrease in performance. For the affine consistently use transformation the runtime per iteration is similar for PSGD-H and FPSGD (~2 ms and ~1 ms, respectively). The overall speed-up in terms of runtime is about 5 for FPSGD, compared to 0.5 for PSGD-H.

It can be seen from Table 4.2 that the Euclidean distance error of all methods is around 5 mm. The p-value of the Wilcoxon signed-rank test of PSGD-J and PSGD-H

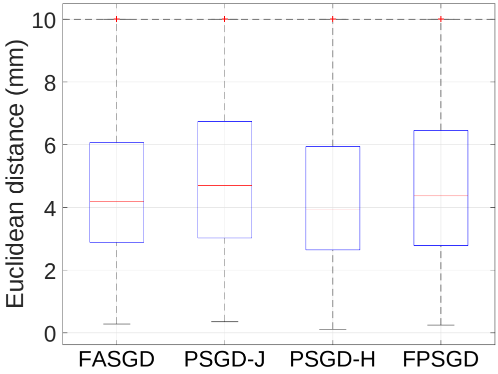


Figure 4.3: Euclidean distance error in mm for the different methods with the SPREAD lung CT data. The experiments were performed using MSD dissimilarity measure and affine transformation model. For FPSGD, $\tau = 0.6$ and $\kappa_{\max} = \infty$ are used.

compared to FASGD is smaller than 0.05, indicating a statistically significant difference. Although significant, the differences are very small, i.e. less than 0.5 mm. The Wilcoxon signed-rank tests of FPSGD (all settings of τ) compared to FASGD show no statistical difference ($p > 0.05$). A boxplot of the Euclidean distance error of 100 corresponding points is given in Figure 4.3, using $\tau = 0.6$ for FPSGD.

4.6.2.2 B-spline registration

The overall results of the experiments on B-spline registration for the SPREAD lung CT data are given in Table 4.3. For all three resolutions, the proposed method took less iterations to obtain the same cost function value as FASGD. Although the proposed method took somewhat longer to estimate the preconditioner compared to FASGD, less iterations were required, resulting in an overall improvement of runtime. For FPSGD ($\tau = 0.6$), the overall speed-up is of a factor of 2. The number of iterations used for PSGD-H to obtain the same cost function value is less than both FASGD and FPSGD, which can also be observed from the convergence plots in Figure 4.5c and Figure 4.5d. However, the overhead of computing the preconditioner increased substantially for the PSGD-H method: around 10^4 seconds for $\sim 10^5$ parameters in resolution 3, while the FPSGD method required ~2s.

The ED errors in Table 4.3 are evaluated at the end of resolution 3. All three methods FASGD, PSGD-H and FPSGD obtained a mean ED error around 1.65 mm, which is within one voxel. The p-value of the Wilcoxon signed-rank test of PSGD-H and FPSGD compared to FASGD is 0.445 and 0.968, respectively, indicating no statistical difference. Figure 4.4 presents the boxplot of the Euclidean distance error for the different methods, where for FPSGD we used $\tau = 0.6$ and $\kappa_{\max} = 4$.

4.6.3 Results of multi-modal image registration

4.6.3.1 RIRE brain data

Table 4.4 presents the runtime differences and the mean Euclidean distance error of the RIRE experiments for all methods. We can observe that much less iterations are required for PSGD-J and FPSGD compared to FASGD. The speed-up in iterations is

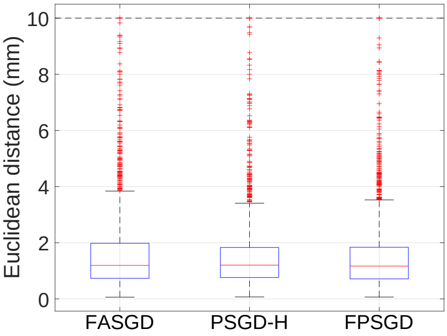
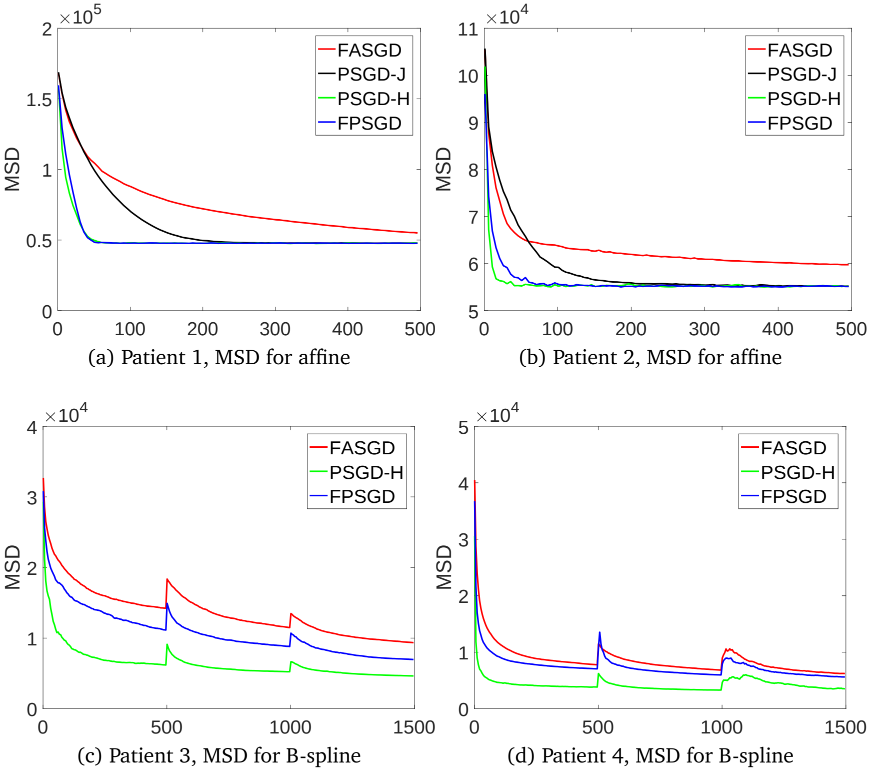


Figure 4.4: Euclidean distance error in mm for the different methods with the SPREAD lung CT data. The experiments were performed using MSD dissimilarity measure and B-spline transformation model. For FPSGD, $\tau = 0.6$ and $\kappa_{\max} = 4$ are used.



(a) Patient 1, MSD for affine

(b) Patient 2, MSD for affine

(c) Patient 3, MSD for B-spline

(d) Patient 4, MSD for B-spline

Figure 4.5: Convergence plots of four cases in the experiments of the SPREAD lung CT data, showing the cost function value (MSD) against the iteration number. For B-spline registration of FPSGD, $\tau = 0.6$ and $\kappa_{\max} = 4$ are used.

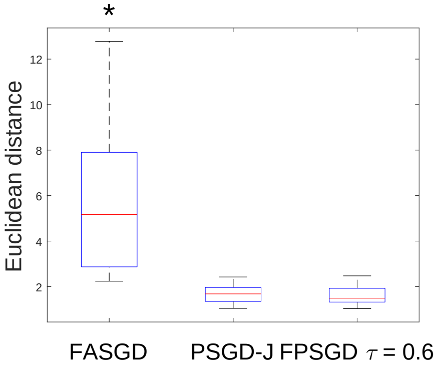


Figure 4.6: Euclidean distance error in mm for different methods with the RIRE brain data. The experiments were performed using MI dissimilarity measure and rigid transformation model. For FPSGD, $\tau = 0.6$ and $\kappa_{\max} = \infty$ are used.

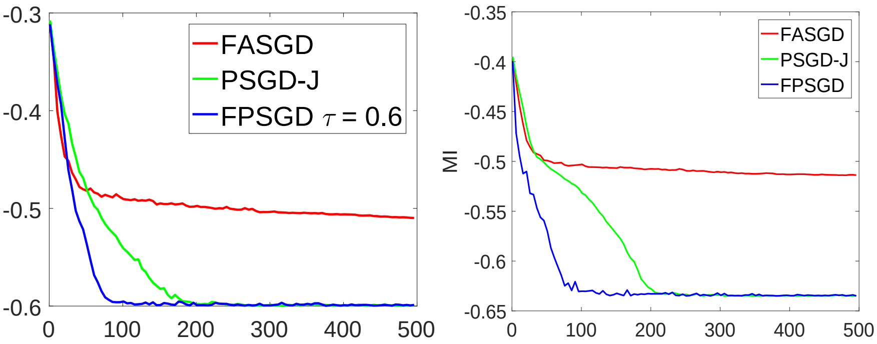


Figure 4.7: Convergence plots of two patients of the RIRE brain dataset, showing the cost function value (negative mutual information measure) against the iteration number. For FPSGD, $\tau = 0.6$ and $\kappa_{\max} = \infty$ are used.

a factor of 10. It can also be seen that the speedup in runtime is around 5 for the FPSGD method. The convergence plots in Figure 4.7 show substantial improvement in convergence rate for FPSGD.

The boxplots of the Euclidean distance error for the RIRE data are shown in Figure 4.6. The median Euclidean distance of 9 patients before registration is 21.7 mm. As we can see, the FASGD method that manually chooses a scaling factor is inferior to the other two methods. From Table 4.4, it can be seen that the Wilcoxon signed-rank tests between FASGD and FPSGD with different τ show significant statistical differences ($p < 0.05$), except for $\tau = 1.0$.

4.6.3.2 BrainWeb simulated brain data

The results of the BrainWeb experiment are shown in Table 4.5, Figure 4.8 and Figure 4.9. The number of iterations for FPSGD ($\tau = 0.6$) to obtain the same cost function value (MI) as FASGD is around 200, resulting in a runtime speed-up of about a factor of 5, as can be seen in Table 4.5. These improvements can also be observed from the

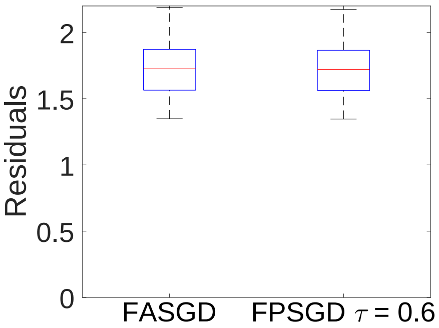


Figure 4.8: Residuals in mm for the different methods with the BrainWeb simulated brain data. The experiments were performed using MI dissimilarity measure and B-spline transformation model. The parameter settings of FPSGD are $\tau = 0.6$ and $\kappa_{\max} = 4$.

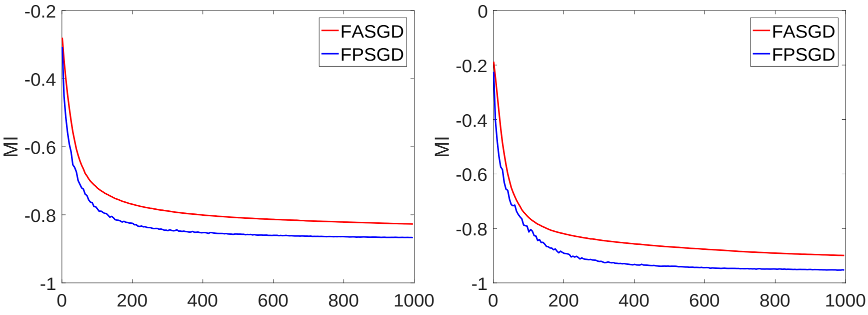
convergence plots in Figure 4.9.

The mean residuals of the different methods show a similar result. The Wilcoxon signed-rank test between FASGD and FPSGD ($\tau = 0.6$) shows a significant statistical difference ($p = 0.031$). However, from Table 4.5, it can be seen that the difference is very small (around 0.02). Increasing the regularization factors τ can achieve a faster convergence rate, however, most registrations failed for $\tau = 1.0$. The boxplots of the residuals of both FASGD and FPSGD ($\tau = 0.6$, $\kappa_{\max} = 4$) are shown in Figure 4.8.

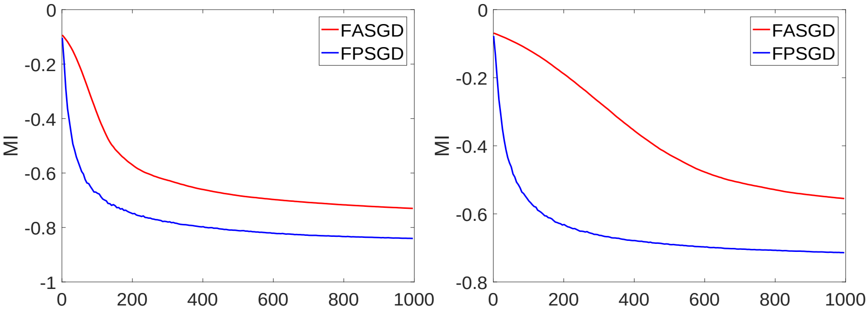
4.7 Discussion

The experimental results show that the proposed FPSGD method works well in both mono-modal as well as multi-modal image registration, in combination with different transformation models and dissimilarity measures, showing that the proposed method is generic for different registration problems. The proposed FPSGD method can be used for different transformation models, unlike PSGD-J which was proposed only for rigid and affine registration problems. Compared to FASGD which is not preconditioned, the proposed FPSGD method not only obtains the same registration accuracy, moreover improves the convergence. Without the computational burden of the Hessian matrix calculation and decomposition, the proposed FPSGD method takes much less time than PSGD-H to construct a preconditioner. Additionally, the proposed method requires only a cost function gradient and a set of transformation Jacobians, while PSGD-H also needs the implementation of the self-Hessian. Most importantly, the proposed FPSGD method is more generic for different modalities and not limited to mono-modal problems like PSGD-H.

Compared to FASGD, the main improvement of the proposed FPSGD method is in the convergence rate, inducing a speedup in runtime of a factor of 2.0-6.0 depending on the application. Specifically, the proposed FPSGD method used half a second to obtain the same registration accuracy as FASGD for the affine registration on the SPREAD lung CT with image sizes of $450 \times 300 \times 130$, while FASGD took 2 seconds. The proposed FPSGD method needs much less computation time for the preconditioner estimation than PSGD-H: ~2 seconds vs ~10^4 seconds for ~10^5 transformation parameters, see



(a) Patient 1 (b) Patient 2



(c) Patient 3 (d) Patient 4

Figure 4.9: Convergence plots of the experiment of the BrainWeb simulated dataset, showing the cost function value (negative MI dissimilarity measure) against the iteration number. For FPSGD, $\tau = 0.6$ and $\kappa_{\max} = 4$ are used.

Table 4.3. This large difference between different methods in the computation time of preconditioner estimation can be attributed to the complexity of different methods. For PSGD-H, the complexity is highly due to the Cholesky decomposition of $\mathcal{O}(N_p^3)$, i.e. depending on the number of transformation parameters, while for the FPSGD method the complexity is only linear in the number of samples $\mathcal{O}(N_p)$. In addition, the runtime per iteration for the PSGD-H method increased to ~5 seconds for $N_P \approx 10^5$ transformation parameters, due to the multiplication of a full matrix $\boldsymbol{P}$ instead of only a diagonal matrix for FPSGD (~24 ms per iteration for MI dissimilarity measurement). We therefore conclude that the proposed FPSGD method converges faster than the FASGD method and is more time-efficient than the PSGD-H method.

There are two parameters that influence the performance of the proposed FPSGD method: the regularization factor τ and the maximum condition number $\kappa_{\max}$. We validated the influence of both parameters experimentally. We showed that the extreme cases ($\tau = 0$ and $\tau = 1$) yielded suboptimal results, indicating that regularization of the preconditioner is required. The proposed regularization method performs a Gaussian smoothing, considering entries with a similar Jacobian response. This choice reflects the observation that transformation parameters that have a similar effect on the displacement, require similar preconditioning, and vice versa. For example, for the affine transformation rotation and translation require different scaling. The experiments showed that the choice $\tau = 0.6$ yielded good results for all applications.

For ill-conditioned problems, $\kappa_{\max}$ serves as a safe guard to prevent extreme values in the preconditioner. In the experiment on the SPREAD data, different $\kappa_{\max}$ obtained a similar registration accuracy, however, the convergence has some oscillations for $\kappa_{\max} > 4$ in the second and third resolution in Figure 4.1. For the BrainWeb data, best results were acquired with $\kappa_{\max} = 4$ and the convergence plots are also very stable. Overall, the best choice of $\kappa_{\max}$ is between 2 and 4 for nonrigid registration, while $\kappa_{\max} = \infty$ can be used for rigid and affine registration.

To further improve the proposed FPSGD method the following may be considered. Firstly, the proposed preconditioning scheme detailed in Algorithm 2 is very suitable for further acceleration on a Graphics Processing Unit (GPU). It could be easily applied for the parallel computing of the gradient and the preconditioner [22], therefore this will be beneficial when going to variable preconditioning. Secondly, our method can be combined with the variable preconditioning techniques for difficult problems where the curvature of the cost function changes iteratively. Instead of estimating the preconditioner once at the beginning of each resolution, we may regularly update it. A GPU implementation is then warranted to keep the runtime per iteration low. Furthermore, a stopping condition other than the number of iterations will be required to practically take advantage of the convergence improvements. An interesting option suitable in a stochastic setting is a moving average of the noisy gradients over a few iterations.

4.8 Conclusion

In this chapter, we proposed a generic preconditioner estimation method for the stochastic gradient descent optimizers used in medical image registration. Based on the observed distribution of the voxel displacements, this method automatically constructs a diagonal preconditioner, avoiding the computationally complex calculation of the Hessian matrix. All tested methods obtained a similar final registration accuracy in all tested datasets. The proposed FPSGD optimizer, however, outperforms FASGD and PSGD-J in terms of convergence rate, while yielding a similar computational overhead. While a previous method (PSGD-H) even further reduces the required number of iterations, it comes at a substantial overhead in computing the preconditioner, especially for high dimensional transformations. Additionally, PSGD-H can only be used in mono-modal problems and requires the implementation of a Hessian matrix computation.

We conclude that the proposed method can act as a generic preconditioner for optimization in registration methods, yielding similar accuracy as gradient descent routines while substantially improving the convergence rate.

Acknowledgments

This research was supported by ZonMw, the Netherlands Organization for Health Research and Development [grant number 104003012]; and the China Scholarship Council [grant number 201206130066]. The RIRE project is acknowledged for providing a platform for rigid registration evaluation. Dr. M.E. Bakker and J. Stolk are acknowledged for providing a ground truth for the SPREAD study data used in this chapter.

5

Evaluation of an open source registration package for automatic contour propagation in online adaptive intensity-modulated proton therapy of prostate cancer

This chapter was adapted from:

Y. Qiao, T. Jagt, M. Hoogeman, B.P.F. Lelieveldt and M. Staring. **Evaluation of an open source registration package for automatic contour propagation in online adaptive intensity-modulated proton therapy of prostate cancer**, submitted

Abstract

Purpose: To investigate the performance of an open source deformable image registration package, `elastix`, for fast and robust contour propagation in the context of online-adaptive IMPT for prostate cancer.
Material and Methods: A planning and 7-10 repeat CT scans were available of 18 prostate cancer patients. Automatic contour propagation of repeat CT scans was performed using `elastix` and compared with manual delineations in terms of geometric accuracy and runtime. Dosimetric accuracy was quantified by generating IMPT plans using the propagated contours expanded with a 2-mm (prostate) and 3.5-mm margin (seminal vesicles and lymph nodes) and calculating coverage based on the manual delineation. A coverage of $V_{95\%} \geq 98\%$ was considered clinically acceptable.
Results: Contour propagation runtime varied between 13 and 55 seconds for different registration settings. For the fastest setting, 83 in 93 (89.2%), 73 in 93 (78.5%), and 91 in 93 (97.9%) registrations yielded clinically acceptable dosimetric coverage of the prostate, seminal vesicles, and lymph nodes, respectively. For the prostate, seminal vesicles, and lymph nodes the Dice Similarity Coefficient (DSC) = 0.88 ± 0.03, 0.66 ± 0.16, 0.88 ± 0.03 and the mean surface distance (MSD) = 1.4 ± 0.3 mm, 1.8 ± 0.8 mm, 1.5 ± 0.4 mm, respectively.
Conclusions: With a dosimetric success rate of 78.5% to 97.9%, this software may facilitate online adaptive IMPT of prostate cancer using a fast, free and open implementation.

5.1 Introduction

Intensity-modulated proton therapy (IMPT) for prostate cancer treatment has the potential to deliver a highly localized dose distribution to the target volume. However, IMPT is also sensitive to treatment-related uncertainties that may distort the planned dose distribution. These include uncertainties in patient set-up, inter-fraction and intra-fraction variations in the shape and position of the target volume and organs at risk (OARs), and uncertainties in the range of the proton beams [4, 5, 6, 7, 8].

The uncertainties are usually accounted in the clinical-target-volume to planning-target-volume (CTV-to-PTV) margin, while proton-therapy specific effects are accounted for by including robustness in the optimization of the treatment plan. Both come at a price in terms of sparing of OARs. Therefore, ideally, these uncertainties should be tackled at each treatment fraction by re-optimizing the treatment plan, based on a new CT scan-of-the-day. This requires new contours for the target and OARs. Manual re-contouring, however, takes a substantial amount of time, which would give rise to new shape and position uncertainties. Fast automatic methods are therefore mandated.

Deformable image registration (DIR) provides an efficient way to automatically re-contour the repeat CT scan by establishing the spatial correspondence with the planning CT scan. The manual contours from the planning CT are then propagated to the repeat CT, thereby compensating for anatomical changes that may have occurred in the meantime. In combination with fast IMPT treatment replanning this enables the use of small margins and limited amount of robustness without losing dose coverage. The important step of DIR in an online-adaptive IMPT procedure (re-contouring, re-planning, patient-specific QA), however, is currently rather time-consuming. In this chapter we therefore developed and evaluated a fast and automatic DIR method, and performed a dosimetric evaluation for IMPT.

Many DIR algorithms implemented in commercial or open source software packages could be used clinically [104]. Commercial software packages are, however, frequently black boxes for users and have limited choices for parameter customization. Open source packages are much more flexible and provide fully customizable algorithms [105, 106, 10]. Moreover, they support the fundamental scientific principle of reproducibility, sharing of knowledge and thereby promote opportunities for scientific advancement [107, 108, 109].

The validation of DIR for radiation therapy has been performed in terms of dosimetric coverage of the prostate [110, 111, 112] and other anatomical areas [113, 114]. The relation between registration settings and geometric accuracy was also investigated [115]. However, the time cost of image registration [105, 106] is also important for online-adaptive IMPT. Kupelian *et al.* [116] found the prostate having a shift larger than or equal to 5 mm within 30 seconds in 15% of the fractions. Therefore, these shifts should be accounted for by DIR within this time span [117]. In 2009, Godley *et al.* reported 9 minutes to register two CT images [118]. A recent clinical practice report [98] mentioned an average rigid registration time of 79 seconds in their treatment planning, while the computational complexity for DIR was not stated. Another paper presented a matching time of 147 seconds for prostate cancer patients [119], which included the acquisition time and the time for manually matching fiducial markers. Although a graphics processing unit (GPU) and other computational techniques [106, 22, 120] can be used to accelerate image registration, this has not yet led to real-time and robust algorithms. To our knowledge, the validation of open source packages on registration accuracy in relation to runtime has not been investigated so far for prostate cancer. In this chapter, we investigate the performance of a DIR package, in terms of runtime, geometric and dosimetric performance in IMPT. The presented package, `elastix`, is open source (Apache 2.0 license) and freely available for commercial and clinical application, research and further development.

5.2 Materials and Methods

5.2.1 Patients and imaging

Eighteen patients treated for prostate cancer with IMPT at Haukeland University Hospital in 2007 were included in this study. A planning CT and 7 to 10 repeat CTs were acquired out-of-room for each patient using a Philips Brilliance Big Bore CT scanner and anonymized with DicomWorks version: 2.2.1. Each CT scan contained 90 to 180 slices and were reconstructed with a slice thickness of 2-3 mm. Each slice was of size 512 × 512 pixels and had an in-slice pixel resolution ranging from 0.84 × 0.84 mm to 0.95 × 0.95 mm. Golden fiducial markers (2 to 3) were implanted in the prostate for daily set-up to align the target with the treatment beams [121].

For each CT scan, the prostate, seminal vesicles, lymph nodes, bladder and rectum were delineated by an expert, and independently reviewed by another expert [4]. The original images and delineations are in DICOM-RT format and were converted to meta image format and VTK meshes using MevisLab (`http://www.mevislab.de/`). Manual delineations of the bowels and femoral heads were available for 11 patients, which were included for dosimetric evaluation.

5.2.2 Image registration

In this study, DIR was performed using `elastix` [10] (`http://elastix.isi.uu.nl`). All experiments were performed on a PC with 16 GB memory, Windows 7 operating system and an Intel Xeon E5-1620 CPU with 4 cores (3.6 GHz), utilizing only the CPU, without GPU acceleration.

The planning CT (moving image) was registered to the repeat CT scans (fixed image), after which the manual delineations from the planning CT were propagated based on the DIR results. Registrations were initialized based on the centers of gravity of the bony anatomy (tissue with HU > 200) of the fixed and moving image. A mask of the torso was generated automatically using in-house software Pulmo (commercialized by Medis specials, Leiden, The Netherlands), to eliminate the influence of the couch on image registration quality [50]. The registration procedure includes an affine registration to tackle large movements of organs and is followed by a deformable registration to compensate for local deformations. A fast recursive implementation of B-spline transformation model was used [57, 122]. Mutual information was used as a similarity measure [123]. For optimization we used an accelerated version of adaptive stochastic gradient descent [100]. A three level multi-resolution scheme was chosen to deal with local minima and to reduce calculation burden. Detailed parameter settings are available at the `elastix` website. In the experiments we varied the number of iterations between 100, 500, 1000 and 2000 iterations per resolution, and inspected the influences on DIR quality and runtime.

5.2.3 Evaluation measures

For quantitative evaluation of the automatic DIR method we considered runtime, recontouring quality and dosimetric coverage. Runtime is measured by the system clock, in seconds. The recontouring quality of the prostate, seminal vesicles, lymph nodes, bladder and rectum is measured by comparing the automatically propagated contour from the planning CT with the manual delineation of the repeat CT. We consider the Dice Similarity Coefficient (DSC) [13]:

$$\mathrm{DSC} = \frac{1}{R}\sum \frac{2|\boldsymbol{M}\cap \boldsymbol{F}|}{|\boldsymbol{M}|+|\boldsymbol{F}|}, \tag{5.1}$$

where R is the total number of segmentations, $\boldsymbol{F}$ and $\boldsymbol{M}$ are the manually delineated regions in the fixed image and the propagated regions in the moving image, respectively.

Two types of symmetric surface distances are used, namely the mean surface distance (MSD) and the 95% percentile Hausdorff distance (95%HD). Let $F = \{a_1, a_2, \ldots, a_n\}$, and $M = \{b_1, b_2, \ldots, b_m\}$ represent the mesh points from two surfaces, then we have [124]:

$$\mathrm{MSD} = \frac{1}{2}\left(\frac{1}{n}\sum_{i=1}^{n} d(a_i, \boldsymbol{M}) + \frac{1}{m}\sum_{i=1}^{m} d(b_i, \boldsymbol{F})\right), \tag{5.2}$$

$$\mathrm{HD} = \max\{\max_i\{d(a_i, \boldsymbol{M})\}, \max_j\{d(b_i, \boldsymbol{F})\}\}, \tag{5.3}$$

in which $d(a_i, M) = \min_j \|b_j - a_i\|$. Both distances are computed in 3D. The geometrical success rate γ is defined as the percentage of registrations which have an MSD < 2mm (slice thickness) for the prostate: $\gamma = n\{\mathrm{MSD} < 2\mathrm{mm}\}/N, N = 159$.

To measure the dosimetric impact of differences in manual delineations and automatically delineations, IMPT plans were generated on each repeat CT for both sets of delineations for the 11 patients where delineations of the femoral heads and bowels are available. To evaluate the effect these different delineations have on the dose distributions, both IMPT plans are evaluated on the manual contours, which therefore acts as the ground truth. All IMPT plans were generated using Erasmus-iCycle, an in-house developed treatment planning system [125, 126]. Erasmus-iCycle uses a multi-criteria optimization to generate a clinically desirable Pareto optimal treatment plan on the basis of a wishlist consisting of hard constraints and objectives. A small margin is used (2 mm around the prostate and 3.5 mm around the seminal vesicles and lymph nodes) to compensate for inevitable inaccuracies of the contour-propagation and to account for intra-observer variations in the manual contouring. Note that the margins are far from sufficient to account for shape and positions changes of the target volume, for which clinically typically a margin of 7 mm is used [6, 127, 128]. Dose was prescribed according to a simultaneously integrated boost scheme in which the high-dose PTV (prostate + 2 mm margin) was assigned 74 Gy and the low-dose PTV (seminal vesicles and lymph nodes + 3.5 mm margin) 55 Gy, to be delivered using two laterally opposed beams. The optimization ensures that at least 98% of the target volumes receive at least 95% of the prescribed dose ($V_{95\%} \geq 98\%$). To avoid overdose the optimization ensures that less than 2% of the target volumes receive more than 107% of the highest prescribed dose ($V_{107\%} \leq 2\%$). For the recontouring to be clinically

Table 5.1: Dice overlap of different organs for different registration settings.

	Prostate	Seminal vesicles	Lymph nodes	Rectum	Bladder
Nr. it.	mean ± std	mean ± std	mean ± std	mean ± std	mean ± std
Affine	0.85 ± 0.07	0.47 ± 0.25	0.90 ± 0.04	0.71 ± 0.08	0.78 ± 0.09
100	0.88 ± 0.03	0.66 ± 0.16	0.88 ± 0.03	0.77 ± 0.07	0.88 ± 0.09
500	0.88 ± 0.03	0.68 ± 0.14	0.88 ± 0.03	0.79 ± 0.06	0.89 ± 0.09
1000	0.88 ± 0.04	0.68 ± 0.13	0.87 ± 0.03	0.79 ± 0.06	0.89 ± 0.09
2000	0.87 ± 0.03	0.67 ± 0.13	0.87 ± 0.03	0.80 ± 0.06	0.89 ± 0.10

Table 5.2: Mean surface distance (mm) of different organs for different registration settings.

	Prostate	Seminal vesicles	Lymph nodes	Rectum	Bladder
Nr. it.	mean ± std	mean ± std	mean ± std	mean ± std	mean ± std
Affine	1.64 ± 0.71	2.91 ± 1.65	1.27 ± 0.48	3.92 ± 1.48	4.42 ± 2.10
100	1.36 ± 0.30	1.75 ± 0.84	1.49 ± 0.44	3.16 ± 1.28	2.48 ± 1.77
500	1.40 ± 0.37	1.68 ± 0.79	1.57 ± 0.42	2.97 ± 1.22	2.06 ± 1.46
1000	1.42 ± 0.46	1.67 ± 0.74	1.59 ± 0.42	2.94 ± 1.22	1.99 ± 1.40
2000	1.44 ± 0.50	1.69 ± 0.74	1.61 ± 0.42	2.90 ± 1.20	1.92 ± 1.33

acceptable the automatically generated treatment plans should still fulfill these criteria. The clinical success rate η is defined as the percentage of registrations for which the prostate directly meets the dose treatment criteria: $\eta = n\{V_{95\%} \geq 98\%\}/N$, $N = 93$. A second more conservative measure of clinical success is when all target volumes (the prostate, seminal vesicles and lymph nodes) meet this dosimetric criterium.

5.3 Results

5.3.1 Image registration performance

Examples of automatically propagated contours using DIR are given in Figure 5.1. Table 5.1 presents the overlap after DIR for different number of iterations. For the prostate, we obtained a DSC of 0.88 ± 0.03 for each patient and all settings, and a similar overlap for the lymph nodes. The most difficult structures are the seminal vesicles, which have small volume and only achieved an overlap of 0.66 ± 0.16 for 100 iterations, and 0.67 ± 0.13 for more than 500 iterations. For the OARs, we obtained a DSC of 0.77 ± 0.07 for the rectum and 0.88 ± 0.09 for the bladder for 100 iterations, and small improvements are observed when the number of iterations increased to at least 500. DSC scores generally improved from 100 to 500 iterations, but not after that.

The MSD results are shown in Table 5.2. The MSD of the target organs were smaller than 1.8 mm which was within one voxel ($0.9 \times 0.9 \times 2$ mm). Note that for an increasing number of iterations the MSD slightly increased for the prostate and lymph nodes, likely due to the reduction in MSD of the bladder, rectum and seminal vesicles. The geometrical success rate of the registrations was 96% (153 in 159) for 100 and 500 iterations and 95% (152 in 159) for the other settings, while this value was 77% for affine registration. The 95%HD between the propagated and manual contour is shown in Table 5.3, which shows a similar pattern as the MSD.

Table 5.3: 95% percentile Hausdorff distance (mm) of different organs for different registration settings.

	Prostate	Seminal vesicles	Lymph nodes	Rectum	Bladder
Nr. it.	mean ± std	mean ± std	mean ± std	mean ± std	mean ± std
Affine	3.89 ± 1.67	6.47 ± 3.28	3.12 ± 1.20	11.84 ± 5.70	12.43 ± 6.60
100	3.23 ± 0.98	4.38 ± 2.31	3.66 ± 0.98	10.53 ± 5.75	7.65 ± 6.59
500	3.46 ± 1.44	4.23 ± 2.20	3.93 ± 0.97	10.22 ± 5.82	6.34 ± 5.88
1000	3.51 ± 1.79	4.27 ± 2.26	4.02 ± 0.99	10.27 ± 5.93	6.13 ± 5.63
2000	3.58 ± 1.96	4.38 ± 2.45	4.10 ± 1.03	10.24 ± 5.96	5.93 ± 5.40

The total runtime in seconds for each registration setting was 13.5 ± 1.7, 22.4 ± 1.9, 33.0 ± 2.3, and 54.3 ± 3.1 seconds, for 100, 500, 1000, and 2000 iterations, respectively. Figure 5.2 illustrates the registration accuracy with respect to the mean runtime for different anatomical structures. Boxplots of the Dice overlap, mean surface distance and 95% Hausdorff distance are shown for all registrations ($N = 159$).

5.3.2 Dosimetric validation

All treatment plans were evaluated by visual inspection of the dose distributions, the DVHs of the target volumes and OARs, and the clinical constraints. For the prostate, seminal vesicles and lymph nodes, we report the $V_{95\%}$ and $V_{107\%}$ of each treatment plan that used the DIR-generated contours. For the rectum, we consider V_{45Gy}, V_{60Gy}, V_{75Gy} and D_{mean}, while for the bladder V_{45Gy}, V_{65Gy} and D_{mean} are used, where D_{mean} means the average dose to the structure.

Figure 5.4 shows a boxplot depicting the difference in dosimetric parameters between the automatically generated delineations (100 iterations setting) and the manual delineations, in the treatment plan that was based on the automatically generated delineations. For all dosimetric parameters the median of the differences are close to 0. However, there are some scans for which larger differences occur for especially the $V_{95\%}$ of the seminal vesicles. Table 5.4 shows the percentage of scans for which $V_{95\%} \geq 98\%$ and $V_{107\%} \leq 2\%$ for the treatment plans based on the automatically contoured structures. Note that the success rate when using the manual delineations is close to 100% for all organs. As one can see, DIR using 100 iterations obtained a success rate of 89.2% for the prostate and 78.5% for the seminal vesicles, and these numbers are improved to 89.2% and 88.2%, respectively for 500 iterations. The conservative success rate based on all three target volumes increased from 68.8% to 77.4%, for 100 and 500 iterations, respectively. The 10 out of 93 cases that did not directly meet our definition of clinical success had a $V_{95\%}$ for the prostate between 90% and 97% with a mean of 95% for DIR using 100 iterations. More details are given in the Discussion.

5.4 Discussion

The purpose of this study was to investigate if automatic recontouring for prostate cancer IMPT would be possible, considering the clinical requirements for accuracy, robustness and speed. The overall goal of online adaptive IMPT is to be able to treat with a small margin to spare OARs. This can only be done by daily re-planning,

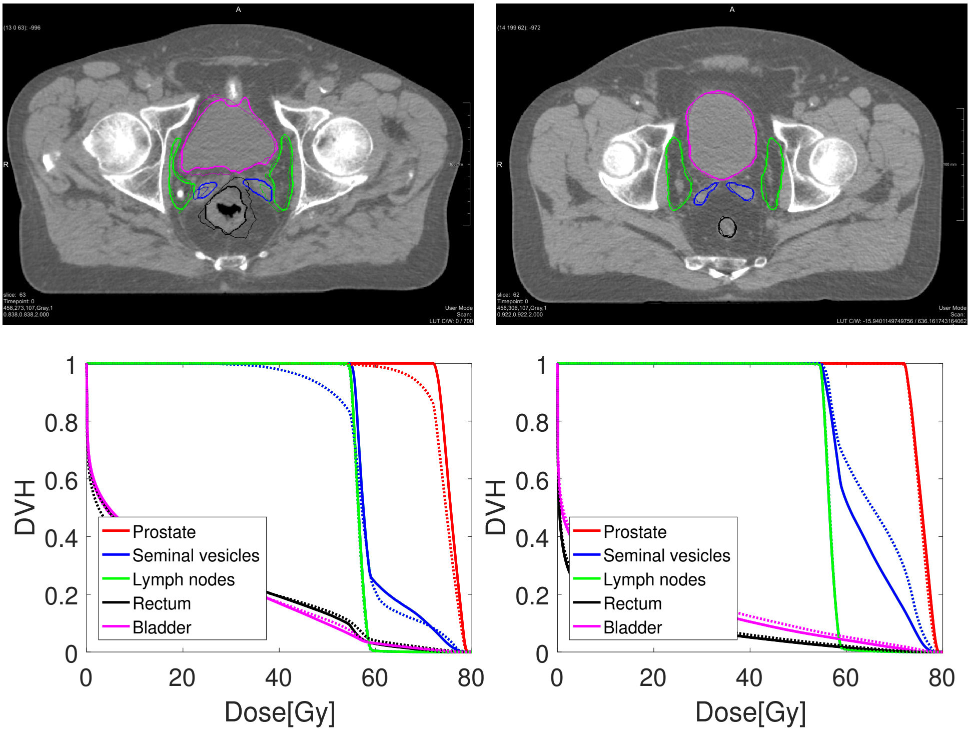


Figure 5.1: An example of one failed case (left) and one successful case (right). The bottom figures are dose volume histograms. The solid line represents the manual contouring results while the dot line is the automatically propagated one with the setting of 100 iterations. For the prostate, the MSD is 2.26 mm and 1.12 mm, while $V_{95\%}$ is 90.80% and 99.83%, respectively. For seminal vesicles, the MSD is 2.74 mm and 1.00 mm, while $V_{95\%}$ is 99.79% and 99.82%, respectively. For lymph nodes, the MSD is 1.45 mm and 0.99 mm, while $V_{95\%}$ both are 100%, respectively.

otherwise coverage loss or underdosage may occur, which is unacceptable. Such daily re-planning warrants automatic recontouring, in this study by DIR. To quantify the clinically more relevant dosimetric impact of such re-planning, we performed a dosimetric validation. The chosen endpoint is therefore $V_{95\%} \geq 98\%$ for each of the target volumes. In general, the registration package `elastix` can automatically re-contour repeat CT scans of the prostate with a desirable accuracy in 13 seconds.

Several aspects were important for registration performance: 1) A correct initialization of DIR was necessary. Alignment of bony anatomy, as used in this study, yielded satisfactory results [4, 129], but exploitation of the implanted gold markers could also be an option [128]; 2) The couch is disturbing the registration and should therefore be removed by masking or cropping. In this study both were used, where cropping was also beneficial for runtime performance; 3) As we had compared the registration accuracy with and without mask, we found that masking is helpful for small volume organs such as the seminal vesicles and rectum, while no differences were observed

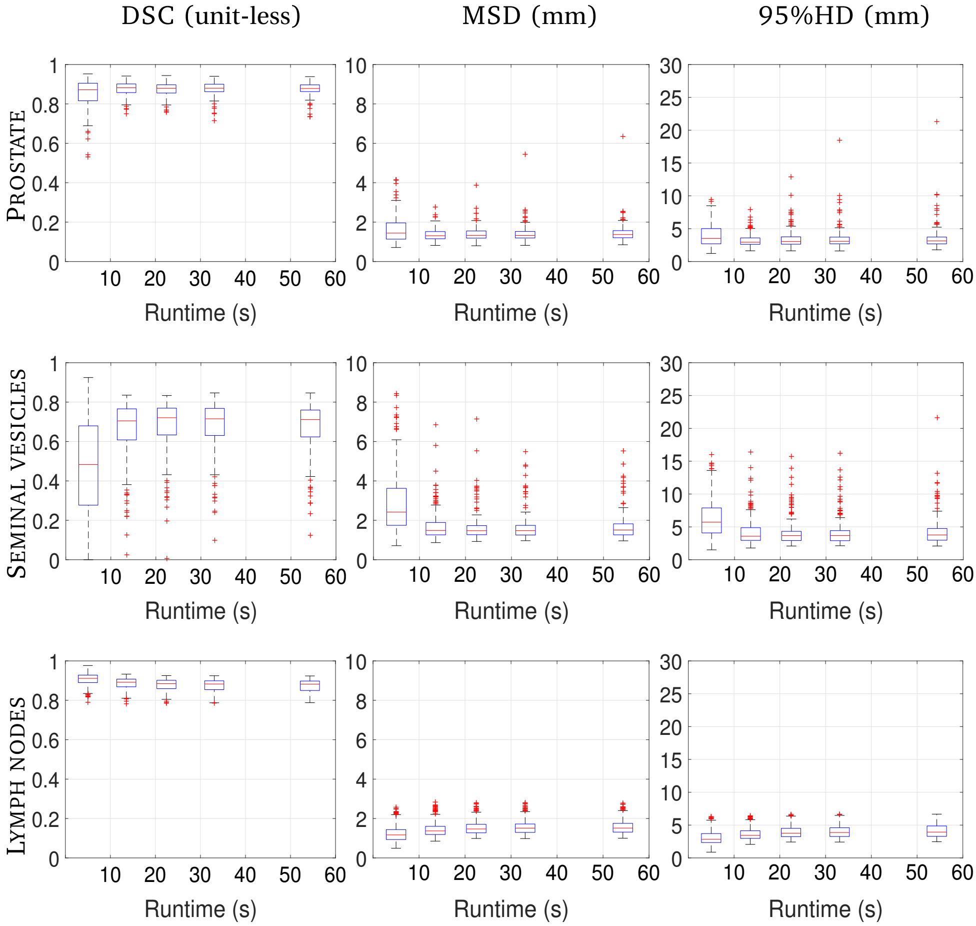


Figure 5.2: Boxplot of registration performance against run time in seconds. From left column to right column the DSC, MSD and 95%HD are shown, respectively. From top to bottom the prostate, seminal vesicles and lymph nodes are shown, respectively. Within one boxplot, from left to right the affine registration and B-spline registrations with 100, 500, 1000 and 2000 iterations are shown, respectively. Each boxplot contains results of 159 registrations.

for the prostate and lymph nodes. This finding is consistent with previous studies [4, 105].

In this study special attention was given to the registration runtime in relation to achieved accuracy, determined by the number of iterations. Overall, registration accuracy increased only slightly when gradually increasing the number of iterations from 100 to 2000, suggesting that early convergence was obtained in most cases. Only for the seminal vesicles an improvement in dose coverage was observed when using 500 iterations, see Table 5.4. The geometrical success rate as expressed by the percentage of registrations with an MSD below the slice thickness of 2 mm was 96%

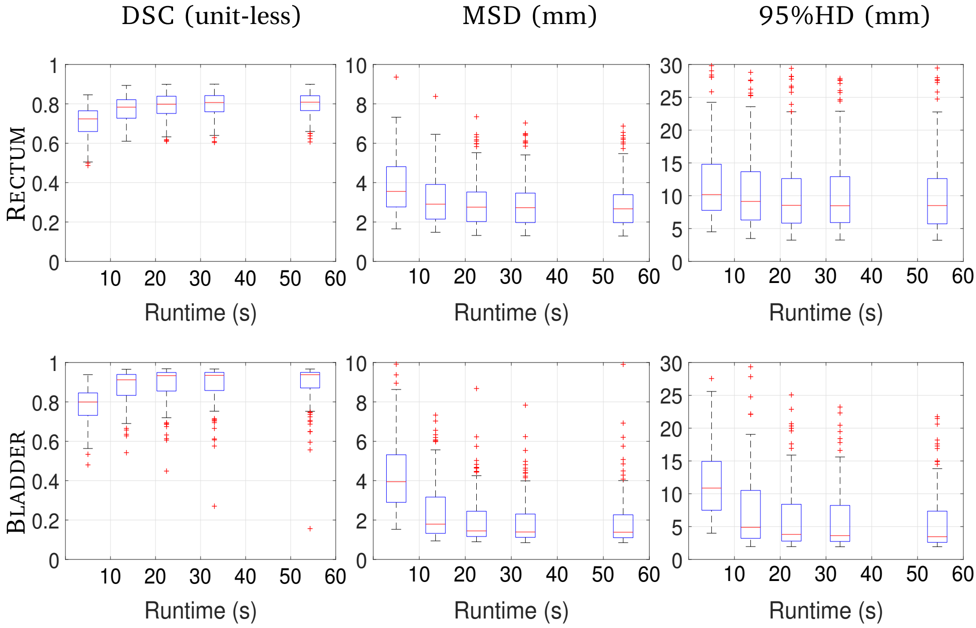


Figure 5.3: Boxplot of registration performance against run time in seconds. From left column to right column the DSC, MSD and 95%HD are shown, respectively. Within one boxplot, from left to right the affine registration and B-spline registrations with 100, 500, 1000 and 2000 iterations are shown, respectively. Each boxplot contains results of 159 registrations.

Table 5.4: Percentage of registrations that meet the dose constraints for the different contours. Conservative success rate (CSR) refers to the percentage of registrations for which all target volumes (the prostate, seminal vesicles and lymph nodes) meet the dose constraints.

		Prostate	Seminal vesicles	Lymph nodes	CSR
$V_{95\%} \geq 98\%$	100	89.2	78.5	97.9	68.8
	500	89.2	88.2	97.9	77.4
	1000	89.2	88.2	98.9	78.5
	2000	90.3	88.2	97.9	77.4
$V_{107\%} \leq 2\%$	100	100.0	100.0	100.0	
	500	100.0	100.0	100.0	
	1000	100.0	100.0	100.0	
	2000	100.0	100.0	100.0	

for the prostate. Clinical success rate, expressed by the dose coverage criteria, was 88% for the prostate. This means that in a high percentage of cases the automatically generated contours can be directly used for online adaptive IMPT. For those patients, smaller margins can be used and less robustness can be included than when using conventional non-adaptive planning, resulting in less dose for the OARs and potentially

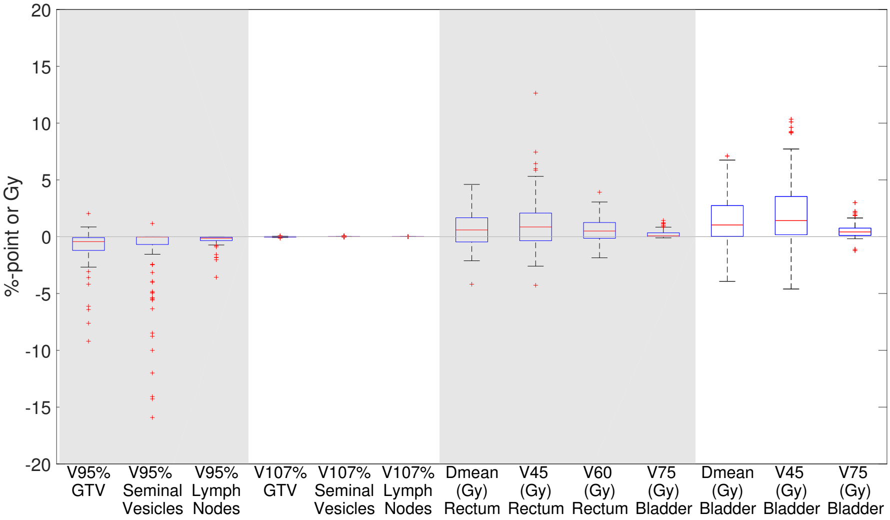


Figure 5.4: Boxplot depicting the difference in dosimetric parameters between the automatically generated delineations and the manual delineations in the treatment plan based on the automatically generated delineations using 100 iterations for 94 scans. Each boxplot indicates the median and the 25^{th} and 75^{th} percentiles of the obtained differences. The line depicts the remaining differences which are not outliers. Values are defined outliers if they are more than 1.5 times the distance between the 25^{th} and 75^{th} quartiles away from the quartiles. The red marks indicate the outliers.

less complications for the patients. For the remaining cases interaction is warranted, for example by manually supplying corresponding points at anatomical regions that require improvement [130, 131].

From the 93 registrations that were assessed in terms of target coverage, 10 (12%) did not directly meet the dose conformity constraints for the prostate. These cases were inspected visually, and we found that 2 cases had many gas pockets in the rectum, while for the other 8 cases no apparent reason was found. For the former we may consider specialized DIR methodology using an intensity modification technique [132]. The MSD of these two cases was around 2.3 mm, and therefore also did not meet the geometrical criteria. Of the 8 remaining cases, one case had a $V_{95\%}$ of 97.99%, which increased to ≥ 98% when 500 or more iterations were used. Two cases had a $V_{95\%}$ around 97% for all settings, which is very close the threshold of 98%; both had an MSD of 1.3 mm, meeting the geometrical criterion for success. Two cases had a $V_{95\%}$ of 96%, which improved to 98% and 99% when 500 iterations or more were used. The remaining three cases obtained a $V_{95\%}$ in the range 92%-96% for the prostate and an MSD in the range 1.6-1.8 mm, so were not far from success.

In order to use `elastix` in a clinical setting, one should consider quality control. This can be done via visual inspection of the generated contours, but assistance by automatic techniques for uncertainty estimation of image registration may be of interest [133, 134]. These techniques may pinpoint areas of possible misregistration, thus enabling a quicker assessment of registration quality. Secondly, for 12% of cases manual assistance or fall-back strategies are needed. Registration can for example be efficiently improved by manual indicating a few landmarks on structure boundaries [131]. Robustness may be further improved by taking into account automatic estimates of the bladder [135] in the registration by optimizing a joint functional. Thirdly, in this study we used clinical quality repeat CT scan, which assumes the availability of an in-room CT-on-rails system. Since such a system is not available in all hospitals, alternatively Cone Beam CT (CBCT) may be used in-room [114, 136]. However, the reduced soft-tissue contrast of CBCT images may increase the uncertainty of DIR, which therefore may influence the quality of the IMPT plans. Fourthly, the registration time assessed in this chapter is determined by a fixed number of iterations, which is not case-specific [100, 137]. A patient-specific stopping condition for stochastic gradient decent, such as considering a moving average of the noisy cost function values (or gradient), may remedy this. Lastly, a further reduction in runtime may be obtained with the help of a GPU and other computational techniques [106, 22, 120].

5.5 Conclusion

In this study we showed that the open source registration package `elastix` can automatically re-contour repeat CT scans of the prostate in 13 seconds, yielding treatment plans that directly meet the dose conformity constraints in 78.5% to 97.9% of cases, and a geometrical criteria of success in 96% of cases. This software may therefore facilitate online adaptive proton therapy of prostate cancer, enabling a reduction in treatment margins.

Acknowledgement

This study was financially supported by ZonMw, the Netherlands Organization for Health Research and Development [grant number 104003012]; Varian Medical Systems, and the China Scholarship Council [grant number 201206130066]. Erasmus MC Cancer Institute also has research collaborations with Elekta AB, Stockholm, Sweden and Accuray Inc, Sunnyvale, USA. The CT-data with contours were collected at Haukeland University Hospital, Bergen, Norway and were provided to us by responsible oncologist Svein Inge Helle and physicist Liv Bolstad Hysing; they are gratefully acknowledged.

6

Discussion and conclusion

Image registration is important for medical image analysis. However, its clinical application is sometimes limited by the speed of the algorithm. For example, in online adaptive radiation therapy a few seconds is ideal, while it usually takes several minutes, at the least. In this thesis, we consider acceleration techniques for parametric intensity-based image registration problems focussing on the optimization routine, specifically the step size and the search direction. The different proposed methods are thoroughly evaluated on different datasets across modalities, subject, similarity measures and transformation models. Depending on the registration settings, the estimation time of the step size is reduced from 40 seconds to less than 1 second when the number of parameters is 10^5, almost 40 times faster. The total registration time of new acceleration techniques (FASGD) is reduced by a factor of 2.5-7x compared with ASGD for the experiments in this thesis. All methods were implemented using C++ in the open source registration package `elastix`. Based on these acceleration schemes we evaluated `elastix` on the application of automatic contour propagation in online adaptive intensity modulated proton therapy for prostate cancer.

6.1 Summary

A summary of the thesis is given below:

Chapter 2 Step size selection is important for gradient descent optimization. It is difficult to perform manually, because for image registration different fixed or moving images, different similarity measures or transformation models require a different step size. Klein *et al.* proposed a method to automatically estimate the step size, however, for a large number of transformation parameters, i.e. in the order of 10^5 or higher, the runtime is unacceptable and the time used in estimating the step size will dominate the optimization [45]. In this chapter, a new automatic method (FASGD) for estimating the optimization step size parameter a, needed for gradient descent optimization methods, has been presented for image registration. The parameter a is automatically estimated from the magnitude of voxel displacements, randomly sampled from the fixed image. A relation between the step size and the expectation and variance of the observed voxels displacement is derived. The proposed method has a free parameter δ, defining the maximally allowed incremental displacement between iterations. Unlike a, it can be interpreted in terms of the voxel size (mm). In addition, it is mostly independent of the application domain, i.e. setting it equal to the voxel size provided good results for all applications evaluated in this chapter. Compared to the original ASGD method, the time complexity of the FASGD method is reduced from quadratic to linear with respect to the dimension of the transformation parameters P. For the B-spline transformation, due to its compact support, the time complexity is further reduced, making the proposed method independent of P. The FASGD method is publicly available via the open source image registration toolbox `elastix` [10]. The FASGD method was evaluated on a large number of registration scenario's and shows a similar accuracy as the original ASGD method. It however improves the time complexity of the step size estimation from 40 seconds to no more than 1 second, when the number of parameters is $\sim 10^5$: almost 40 times faster. Depending on the registration settings, the total registration time is reduced by a factor of 2.5-7x for the experiments in this chapter.

Chapter 3 This chapter presents a stochastic quasi-Newton optimization method (s-LBFGS) for non-rigid image registration. It uses the classical limited memory BFGS method in combination with noisy estimates of the gradient. Curvature information of the cost function is estimated robustly once every L iterations and then used for the next L iterations in combination with stochastic gradients. A novel restarting procedure, automatically selecting the optimization step size, is shown to be beneficial for accelerated convergence. The new optimization routine is validated on follow-up data of 3D chest CT scans (19 patients). Compared to ASGD the proposed method uses about 5 times fewer iterations to reach the same metric value, resulting in an overall reduction in run time of a factor of two. Compared to deterministic LBFGS, s-LBFGS is almost 500 times faster.

Chapter 4 A generic preconditioner estimation method was proposed in this chapter for the stochastic gradient descent optimizers used in medical image registration. Based on the observed distribution of the voxel displacements, this method automatically constructs a diagonal preconditioner, avoiding the computationally complex calculation of the Hessian matrix. We performed experiments to compare our method with FASGD and also other preconditioning techniques: Jacobian type preconditioned stochastic gradient descent (PSGD-J) [70] and Hessian type preconditioned stochastic gradient descent (PSGD-H) [70]. All tested methods obtained a similar final registration accuracy in all tested datasets. The proposed FPSGD optimizer, however, outperforms FASGD and PSGD-J in terms of convergence rate, while yielding a similar computational overhead. While a previous method (PSGD-H) even further reduces the required number of iterations, it comes at a substantial overhead in computing the preconditioner, especially for high dimensional transformations. Additionally, PSGD-H can only be used in mono-modal problems and requires the implementation of a Hessian matrix computation. We conclude that the proposed method can act as a generic preconditioner for optimization in registration methods, yielding similar accuracy as gradient descent routines while substantially improving the convergence rate with a speedup by a factor of 2-4.

Chapter 5 In this chapter we showed that by integration of our algorithms in the open source registration package `elastix`, repeat CT scans of the prostate can be automatically re-contoured for adaptive online IMPT. The online adaption of IMPT could allow for small margins of the target (the prostate), leading to less complications. The dosimetric performance was evaluated with a margin of 2mm, 3.5mm and 3.5mm for the prostate, seminal vesicles and lymph nodes, respectively. The fastest setting of 13 seconds yielded a promising clinical acceptance rate of 83 in 93 cases (89.2%), 73 in 93 (78.5%), and 91 in 93 (97.9%) in dosimetric coverage for the prostate, seminal vesicles and lymph nodes, respectively. We conclude that the fast setting of open source `elastix` can automatic re-contour the daily scans that meet the dose conformity constraints in 78.5% to 97.9% of cases, and a geometrical criteria of success in 96% of cases. This software may therefore facilitate online adaptive proton therapy of prostate cancer, enabling a reduction in treatment margins.

6.2 Discussion

The aim of this thesis was to accelerate the procedure of image registration for clinical applications, such as online adaptive radiation therapy. The image registration procedure includes sampling, transformation, optimization, interpolation and similarity measure selection [10], where the core part and most time-consuming component is optimization. From the evaluation of different optimization strategies performed by Klein *et al.* [25], we realized that the speed of image registration could be improved by subsampling. Later he proposed adaptive stochastic gradient descent (ASGD), which is powerful and achieved a good registration accuracy and fast convergence speed in terms of runtime. However, this method is still not fast enough for large scale problems, for example for 10^6 transformation parameters and 3D volumetric image registration. If the runtime of ASGD could be further reduced, the speed of image registration will be accelerated.

We first focus on the core part of the optimization algorithm, specifically in finding a suitable initial step size and determining a search direction yielding a faster convergence rate. These two parts are not only important for stochastic gradient type methods but also for deterministic gradient type methods. In Chapter 2 we found that the good initialization of the step size is important for the optimization and especially for stochastic gradient type methods. Therefore, different schemes to choose a suitable step size are still an actively pursued topic in the optimization field. Considering search direction schemes, first order gradient methods have the inherit shortcoming of a linear or sublinear convergence rate. A new scheme using second order gradient information with stochastic gradient was thereby proposed and first used in medical image registration problems in Chapter 3. This chapter provides an insight to take use of second order gradient information with an averaging and restarting scheme, both of which are useful to improve the optimization speed. Besides directly using the second order gradient information, we found that we could scale or transform multi-variate problems from an ill-conditioned status to a well-conditioned one at the very beginning of the optimization. A more generic preconditioning scheme was then proposed for (stochastic) gradient descent methods in Chapter 4. The experimental results evaluated on different clinical data had shown that the proposed methods work well for different image registration problems parameterized by different transformation models and different similarity measures. In summary, the step size and the search direction are both important, and essential parts for optimization of image registration.

The successful speedup of the proposed methods on the evaluated datasets encourages further research in this direction. The possibilities are either on the adaptive step size selection or on the efficient search direction scheme. There are many other adaptive step size selection methods such as Adagrad [138], Adadelta [139] and Adaptive Moment Estimation (Adam) [140], which use the properties of the current gradient together with the past gradient. For the search direction, the conjugate stochastic gradient descent [141], variance reduction stochastic gradient descent [142], stochastic gradient descent with momentum [143, 144] and others provide some avenues for future research. Reducing the variance of the stochastic gradient estimation may improve the convergence rate and averaging the stochastic gradient can reduce the noise. Finally, combining these two techniques may yield of further performance improvement. For iterative optimization schemes, these three proposed methods, fast initial step size estimation, stochastic second order gradient method and fast preconditioning scheme, are not independent. Our fast preconditioning scheme was based on the work of fast initial step size estimation, so this scheme could also be used to accelerate second order gradient methods.

In this thesis, we applied these proposed methods to online adaptive IMPT for the prostate cancer and achieved a registration time of 13 seconds for automatic propagation of the contours for most cases. We found that this speed is fast enough for the current procedure of online adaptive IMPT. There are still some aspects that should be considered for improvement. The registration time assessed in this thesis is determined by the number of iterations, which is not case-specific [137, 100]. For cases that are geometrically close, image registration may finish the task with less than the average required iterations, while for difficult cases the number of iterations may be much larger. An adaptive stopping condition for stochastic gradient descent, such as considering a moving average of the noisy cost function values (or gradients), may remedy this. To apply this in clinical practice, the robustness of image registration is also critical. Robustness may be further improved by taking into account automatic estimates of targets for instance the bladder in the prostate cancer [135] in the registration by optimizing a joint function.

Besides the speedup improvements in the optimization, there are several approaches to further accelerate the image registration procedure. The first is the importance-driven sampling strategy used in image registration [44], which could reduce the runtime and improve the registration accuracy. Second, more efficient calculation techniques in the transformation models have become available, such as using a non-uniform cubic B-spline transformation model [145], using a fast recursive implementation [122], and using a random perturbation to smooth the B-spline control grid [146, 103]. Thirdly, fast implementation of for example the interpolator is also useful for acceleration, for example, Shamonin *et al.* [22] proposed to use the GPU for the acceleration. Lastly, other strategies with learning could be applied, such as fast image registration using prior knowledge [147].

6.3 Conclusion

In this thesis we developed several stochastic optimization methods for fast image registration, leading to a 5-10 fold speedup over previous approaches. All proposed methods are implemented using C++ and integrated in the open source registration package `elastix`. We also exploited the usage of high performance computation resources – the life science grid (`lsgrid`) to perform over 10^6 registrations, which significantly reduced computation time for large scale computational tasks. As we have evaluated the proposed method in the application of online adaptive IMPT for prostate cancer, we expect that these methods can achieve the desirable performance for use in clinical practice.

Samenvatting

Beeldregistratie is belangrijk voor medische beeldanalyse. Het toepassen ervan in de kliniek is echter soms beperkt vanwege de snelheid van het algoritme. Voor bijvoorbeeld *online* adaptieve radiotherapie zou een aantal seconden ideaal zijn, terwijl beeldregistratie normaal gesproken op zijn minst een paar minuten duurt. In dit proefschrift worden versnellingstechnieken voor parametrische intensiteit-gebaseerde beeldregistratieproblemen behandeld, waarbij gefocust wordt op de optimalisatieroutine en met name de stapgrootte en de zoekrichting. De verschillende aangedragen methodes zijn grondig geëvalueerd op verscheidene datasets variërend in modaliteit, patiëntengroep, gelijkenismaat en transformatiemodel. De tijd om de stapgrootte te schatten wordt, afhankelijk van de registratieinstellingen, gereduceerd van 40 seconden tot minder dan 1 seconde als het aantal parameters 10^5 bedraagt; dat is bijna 40 keer sneller. De totale registratietijd met de nieuwe versnellingstechnieken (FASGD) is met een factor 2.5-7 reduceerd ten opzichte van ASGD voor de experimenten uitgevoerd in dit proefschrift. Alle methodes zijn geïmplementeerd in C++ in het *open source* registratiepakket `elastix`. Met behulp van deze versnellingen hebben we `elastix` geëvalueerd op het automatisch propageren van intekeningen in *online* adaptief intensiteitsgemoduleerde protontherapie voor prostaatkanker.

Samenvatting

Hieronder volgt de samenvatting van dit proefschrift:

Hoofdstuk 2 Stapgrootteselectie is belangrijk voor gradiëntafdalings-optimalisatie. Handmatig is dit moeilijk uit te voeren, omdat in beeldregistratie verschillende *fixed* en *moving* beelden, beeldgelijkenismaten of transformatiemodellen verschillende stapgroottes vereisen. Klein et al. stelt een automatische stapgrootteschattingsmethode voor, maar in geval van een groot aantal transformatieparameters, dat wil zeggen van ordegrootte 10^5, is de rekentijd onacceptabel en de tijd nodig voor het schatten van de stapgrootte de optimalisatie [45] domineert. In dit hoofdstuk wordt een nieuwe automatische methode (FASGD) voor het schatten van de optimalisatiestapgrootteparameter a, nodig voor gradiëntafdalings-optimalisatiemethodes, gepresenteerd voor beeldregistratie. De parameter a wordt automatisch geschat aan de hand van de magnitude van *voxel* verplaatsingen die willekeurig in het *fixed* beeld bemonsterd worden. Een verband tussen de stapgrootte en de verwachtingswaarde en variantie van de geobserveerde *voxel* verplaatsingen is afgeleid. De voorgestelde methode heeft een vrije parameter δ die de maximaal toegestane incrementele verplaatsing tussen iteraties definieert. In tegenstelling tot a kan deze wel geïnterpreteerd worden in termen van *voxel* grootte (mm). Daarnaast is deze vrijwel geheel onafhankelijk van het toepassingsdomein, dat wil zeggen dat een δ gelijk aan de *voxel* grootte goede resultaten gaf voor alle toepassingen die geëvalueerd zijn in dit hoofdstuk. In vergelijking met de originele ASGD-methode is de tijdscomplexiteit van de FASGD-methode gereduceerd van kwadratisch tot lineair afgezet tegen de dimensie van transformatieparameters P. In het geval van de *B-spline* transformatie is, door zijn beperkte *support*, de complexiteit van de voorgestelde methode verder gereduceerd en onafhankelijk van P. De FASGD-methode is publiekelijk beschikbaar via het *open source* beeldregistratiepakket `elastix` [10]. De FASGD-methode werd geëvalueerd op een groot aantal registratiescenario's en laat een vergelijkbare nauwkeurigheid als de originele ASGD-methode zien. Daarentegen verbetert het de tijdscomplexiteit van de stapgrootteschatting van 40 seconden tot niet meer dan 1 seconde, voor een parameteraantal van $\sim 10^5$: bijna 40 keer sneller. Afhankelijk van de registratieinstellingen is de totale registratietijd gereduceerd met een factor 2.5-7x voor de experimenten in dit hoofdstuk.

Hoofdstuk 3 Dit hoofdstuk presenteert een stochastische quasi-Newton optimalisatiemethode (s-LBFGS) voor niet-rigide beeldregistratie. Het gebruikt de klassieke beperkt-geheugen BFGS in combinatie met ruizige schattingen van de gradiënt. Krommingsinformatie over de kostenfunctie wordt robuust geschat elke L iteraties en vervolgens gebruikt voor de volgende L iteraties in combinatie met stochastische gradiënten. Voor een vernieuwende herstartprocedure die de optimale stapgrootte automatisch selecteert, wordt aangetoond dat deze gunstig is voor versnelde convergentie. De nieuwe optimalisatieroutine is gevalideerd op vervolgdata van 3D-CT-scans van de longen. Vergeleken met ASGD gebruikt de voorgestelde methode ongeveer 5 keer minder iteraties om dezelfde gelijkheidsmaatwaarde te bereiken, resulterende in een factor twee reductie van de algehele rekentijd. Vergeleken met deterministische LBFGS is s-LBFGS bijna 500 keer sneller.

Hoofdstuk 4 Een generieke preconditieschattingsmethode is in dit hoofdstuk voorgesteld voor de stochastische gradiëntafdalingsoptimaliseerder die gebruikt wordt in medische beeldregistratie. Gebaseerd op de geobserveerde verdeling van *voxel* verplaatsingen construeert deze methode een diagonale preconditionering, waarbij een computationeel complexe berekening van de Hessiaanmatrix vermeden wordt. We hebben experimenten uitgevoerd om onze methode te vergelijken met FASGD en ook met andere preconditioneringstechnieken: Jacobiaantype gepreconditioneerde stochastische gradiëntsafdaling (PSGD-J) [70] en Hessiaantype gepreconditioneerde stochastische gradiëntsafdaling (PSGD-H) [70]. Alle geteste methodes verkregen een vergelijkbare registratienauwkeurigheid in alle geteste datasets. De voorgestelde FPSGD-optimaliseerder daarentegen, overtrof FASGD en PSGD-J in termen van convergentiegraad terwijl het een vergelijkbare overhead opleverde. Ondanks dat een vorige methode (PSGD-H) het benodigd aantal iteraties zelfs verder reduceert, brengt dit een substantiële overhead met zich mee voor het berekenen van de preconditionering, met name voor hoog dimensionele transformaties. Daarnaast kan de PSGD-H alleen gebruikt worden voor mono-modale problemen en vereist het een implementatie voor

Hessiaanmatrix berekening. We concluderen dat de voorgestelde methode als een generieke preconditioneerder voor optimalisatie in registratie methodes kan fungeren en daarbij een vergelijkbare nauwkeurigheid oplevert als gradiëntsafdalingsroutines, terwijl de convergentiegraad substantieel verbeterd wordt met een versnellingsfactor van 2-4.

Hoofdstuk 5 In dit hoofdstuk hebben we laten zien dat door integratie van onze algoritmes in het *open source* registratiepakket `elastix` herhaal-CT-scans van de prostaat automatisch ingetekend kunnen worden voor adaptieve *online* IMPT. Het *online* aanpassen van IMPT maakt het mogelijk om kleine marges te hanteren voor de doelstructuur (prostaat), hetgeen tot minder complicaties leidt. De dosimetrische prestaties zijn geëvalueerd met een marge van 2 mm, 3.5 mm en 3.5 mm voor respectievelijk de prostaat, zaadblaasjes en lymfeklieren. De snelste instelling van 13 seconden leverde een veel belovende klinische acceptatiegraad op van 83 van de 93 gevallen (89.2%), 73 van de 93 (78.5%) en 91 van de 93 (97.9%) in dosimetrische dekking van respectievelijk de prostaat, zaadblaasjes en de lymfeklieren. We concluderen dat de snelste instelling voor *open source* `elastix` de dagelijkse scans automatisch kan intekenen en daarbij voldoet aan de conformiteitsbeperkingen in 78.5% tot 97.9% van de gevallen en met een geometrisch criterium voor succes in 96% van de gevallen. Deze software kan daarmee *online* adaptieve protonentherapie voor prostaatkanker faciliteren en een reductie van behandelingsmarges mogelijk maken.

Discussie

Het doel van dit proefschrift was om de procedure van beeldregistratie te versnellen voor klinische toepassingen, zoals *online* adaptieve radiotherapie. De beeldregistratieprocedure omvat bemonsteren, transformeren, optimaliseren, interpoleren en keuze voor de ge-lijk-heids-maat [10], waarbij het kerndeel en meest tijdrovende component de optimalisatie is. Aan de hand van de evaluatie van verschillende optimalisatiestrategiën, uitgevoerd door Klein et al. [25], realiseerden we dat de snelheid van beeldregistratie verbeterd kon worden door onderbemonstering. Later stelde hij adaptieve stochastische gradiëntsafdaling (ASGD) voor die krachtig is en een goede registratienauwkeurigheid en een hoge convergentiesnelheid in termen van rekentijd verkrijgt. Echter, deze methode is niet snel genoeg voor problemen van grote schaal, zoals bijvoorbeeld transformaties met 10^6 parameters en 3D-volumetrische beeldregistratie. Als de rekentijd van ASGD verkort kan worden, zal de beeldregistratie versneld worden. Als eerste richten we ons op de kern van het optimalisatiealgoritme, met name op het vinden van een geschikte initiële stapgrootte en het bepalen van de zoekrichting om een hogere convergentiegraad te behalen. Deze twee delen zijn niet alleen belangrijk voor stochastische gradiënttype methodes maar ook voor deterministische gradiënttype methodes. In hoofdstuk 2 hebben we ontdekt dat een goede initialisatie van de stapgrootte belangrijk is voor de optimalisatie en met name voor stochastische gradiënttype methodes. Om die reden zijn de verschillende aanpakken voor het kiezen van een geschikte stapgrootte een actief onderzoeksveld binnen de optimalisatie. Wat betreft de aanpak van de zoekrichting hebben eerste orde gradiëntmethodes de inherente tekortkoming van een lineaire of sublineaire convergentie. Een nieuwe aanpak van tweede orde gradiëntinformatie met stochastische gradiënten werd daarom voorgesteld en als eerste gebruikt in medische beeldregistratieproblemen in hoofdstuk 3. Dat hoofdstuk geeft inzicht om gebruik te maken van tweede orde gradiëntinformatie met een middelings- en herstartstrategie die beide helpen om de optimalisatiesnelheid te verbeteren. Naast dat we de tweede orde gradiëntsinformatie direct gebruiken, vonden we dat multi-variate problemen van een slecht geconditioneerde toestand geschaald of getransformeerd konden worden tot een goed geconditioneerde aan het begin van de optimalisatie. Een generiekere preconditioneringsstrategie werd vervolgens voorgesteld in hoofdstuk 4. De resultaten van de experimenten, geëvalueerd op verschillende klinische datasets, hebben laten zien dat de voorgestelde methode goed werkt voor verschillende registratieproblemen geparametriseerd door verschillende transformatiemodellen en verschillende gelijkenismaten. Samengevat, de stapgrootte en zoekrichting zijn beide belangrijk en zijn essentiële onderdelen van optimalisatie in beeldregistratie.

De succesvolle versnelling op de geëvalueerde datasets door de aangedragen methodes moedigen aan tot verder onderzoek in deze richting. De mogelijkheden liggen dan wel op het gebied van de adaptieve stapgrootteselectie of wel bij effectieve zoekrichtingsstrategieën. Er bestaan vele andere adaptieve stapgrootteselectiemethodes zoals Adagrad [138], Adadelta [139] en adaptieve impulsschatting (Adam) [140], die de eigenschappen van de huidige gradiënt met de vorige gradiënt combineren. In het geval van de zoekrichting bieden de geconjugeerde stochastische gradiëntsafdaling [141], de variantie-reducerende stochastische gradiëntsafdaling [142], de stochastische gradiëntsafdaling met impuls en ook andere methodes perspectief voor verder onderzoek. Het reduceren van de variantie van de stochastische gradiëntsschatting kan de convergentie versnellen en het middelen van de stochastische gradiënten kan de ruis verminderen. De combinatie van deze twee technieken kan uiteindelijk een verdere prestatieverbetering opleveren. Omtrent iteratieve optimalisaties zijn de drie methodes, de snelle initiëlestapgrootteschatting, de stochastische tweede orde gradiëntsmethode en de snelle preconditioneringsstrategie niet onafhankelijk. Onze snelle preconditioneringsstrategie is gebaseerd op het werk van de snelle initiëlestapgrootteschatting en daarom kan deze strategie ook gebruikt worden om andere tweede orde gradiëntsmethodes te versnellen.

In dit proefschrift hebben we de voorgestelde methodes toegepast op *online* adaptieve IMPT voor prostaatkanker en hebben voor de meeste gevallen een registratietijd van 13 seconden behaald voor het automatische propageren van intekeningen. Deze snelheid achtten we hoog genoeg voor de huidige procedure van *online* adaptieve IMPT. Er zijn nog een aantal aspecten die overwogen dienen te worden voor verbetering. De registratietijd die bepaald werd in dit proefschrift, hangt af van het aantal iteraties, hetgeen niet specifiek is per geval [137, 100]. Voor gevallen die geometrisch dichtbij zijn, kan beeldregistratie de taak in minder dan de gemiddelde vereiste iteraties afronden, terwijl bij moeilijke gevallen het aantal iteraties veel groter dient te zijn. Een adaptieve stopconditie voor stochastische gradiëntsafdaling, zoals het bijhouden van een lopend gemiddelde van de ruisachtige kostenfunctiewaarden (of -gradiënten), kan overwogen worden om dit verhelpen. Om dit toe te passen in de kliniek is ook de robuustheid van beeldregistratie kritiek. Robuustheid kan verder worden verbeterd door rekening te houden met automatische schattingen van de doelstructuren zoals de blaas bij prostaatkanker [135] door dit in een gezamenlijke functie te optimaliseren in de registratie.

Naast de snelheidsverbeteringen van de optimalisatie zijn er verscheidene andere methodes om de registratieprocedure verder te versnellen. Als eerste is er de importantie-gedreven bemonsteringstechniek voor beeldregistratie [44], die de rekentijd kan ver-kort-en en de registratienauwkeurigheid kan verbeteren. Ten tweede zijn er efficiëntere berekeningsmethoden voor de transformatiemodellen beschikbaar gekomen, zoals een snelle recursieve implementatie [145] voor het niet-uniforme kubische *B-spline*-transformatiemodel [122] en het gebruikmaken van willekeurige perturbaties op het *B-spline*-coëfficiëntenraster [146, 103]. Als derde zijn ook de implementaties van onder andere de interpoleerder bruikbaar voor versnelling; Shamonin *et al.* [22] stelde bijvoorbeeld een GPU-acceleratie voor. Ten slotte kunnen lerende strategieën worden toegepast, zoals snelle beeldregistratie met behulp van voorkennis [147].

Conclusie

In dit proefschrift ontwikkelden we diverse stochastische optimalisatiemethodes voor snelle beeldregistratie die een 5-10-voudige versnelling ten opzichte van vorig werk bewerkstelligden. Alle voorgestelde methoden werden geïmplementeerd met behulp van C++ en geïntegreerd in het *open source* registratiepakket `elastix`. Om meer dan 10^6 registraties uit te voeren, benutten we ook de *high performance computation* ondersteuning van het *life science grid* (`lsgrid`) dat de rekentijd voor grootschalige computertaken aanzienlijk verminderd. Aan de hand van de evaluaties van de voorgestelde methode binnen de toepassing van *online* adaptieve IMPT voor prostaatkanker, verwachten we dat deze methoden de gewenste prestatie voor gebruik in de klinische praktijk kunnen bereiken.

Bibliography

[1] M. A. Viergever, J. A. Maintz, S. Klein, et al. *A survey of medical image registration–under review*. 2016.

[2] B. Zitova and J. Flusser. “Image registration methods: A survey”. In: *Image and vision computing* 21.11 (2003), pages 977–1000.

[3] J. P. Pluim, J. A. Maintz, and M. A. Viergever. “Mutual-information-based registration of medical images: a survey”. In: *IEEE Transactions on Medical Imaging* 22.8 (2003), pages 986–1004.

[4] S. Thörnqvist, J. B. Petersen, M. Høyer, et al. “Propagation of target and organ at risk contours in radiotherapy of prostate cancer using deformable image registration”. In: *Acta Oncologica* 49.7 (2010), pages 1023–1032.

[5] M Zhang, D. Westerly, and T. Mackie. “Introducing an on-line adaptive procedure for prostate image guided intensity modulate proton therapy”. In: *Physics in Medicine and Biology* 56.15 (2011), page 4947.

[6] S. Thörnqvist, L. P. Muren, L. Bentzen, et al. “Degradation of target coverage due to inter-fraction motion during intensity-modulated proton therapy of prostate and elective targets”. In: *Acta Oncologica* 52.3 (2013), pages 521–527.

[7] A. Lomax. “Intensity modulated proton therapy and its sensitivity to treatment uncertainties 1: the potential effects of calculational uncertainties”. In: *Physics in Medicine and Biology* 53.4 (2008), page 1027.

[8] A. Lomax. “Intensity modulated proton therapy and its sensitivity to treatment uncertainties 2: the potential effects of inter-fraction and inter-field motions”. In: *Physics in Medicine and Biology* 53.4 (2008), page 1043.

[9] S. van de Water, H. M. Kooy, B. J. Heijmen, and M. S. Hoogeman. “Shortening delivery times of intensity modulated proton therapy by reducing proton energy layers during treatment plan optimization”. In: *International Journal of Radiation Oncology* Biology* Physics* 92.2 (2015), pages 460–468.

[10] S. Klein, M. Staring, K. Murphy, et al. “Elastix: a toolbox for intensity-based medical image registration”. In: *IEEE Transactions on Medical Imaging* 29.1 (2010), pages 196–205.

[11] P. Wolfe. “Convergence conditions for ascent methods”. In: *SIAM review* 11.2 (1969), pages 226–235.

[12] H. Robbins and S. Monro. “A stochastic approximation method”. In: *Ann. Math. Statist.* 22.3 (Sept. 1951), pages 400–407.

[13] A. Klein, J. Andersson, B. A. Ardekani, et al. “Evaluation of 14 nonlinear deformation algorithms applied to human brain MRI registration”. In: *Neuroimage* 46.3 (2009), pages 786–802.

[14] A. Plakhov and P. Cruz. “A stochastic approximation algorithm with step-size adaptation”. In: *Journal of Mathematical Sciences* 120.1 (2004), pages 964–973.

[15] S. Klein, J. Pluim, M. Staring, and M. Viergever. “Adaptive stochastic gradient descent optimisation for iImage registration”. English. In: *International Journal of Computer Vision* 81.3 (2009), pages 227–239.

[16] H. J. Kushner and G. Yin. *Stochastic approximation and recursive algorithms and applications*. Volume 35. Springer Science & Business Media, 2003.

[17] R. Suri and Y. T. Leung. “Single run optimization of a SIMAN model for closed loop flexible assembly systems”. In: *Proceedings of the 19th Conference on Winter Simulation*. WSC ’87. Atlanta, Georgia, USA: ACM, 1987, pages 738–748.

[18] R. Brennan and P. Rogers. “Stochastic optimization applied to a manufacturing system operation problem”. In: *Simulation Conference Proceedings, 1995. Winter*. 1995, pages 857–864.

[19] J. Maintz and M. A. Viergever. “A survey of medical image registration”. In: *Medical Image Analysis* 2.1 (1998), pages 1–36.

[20] A. Sotiras, C. Davatzikos, and N. Paragios. “Deformable medical image registration: A survey”. In: *IEEE Transactions on Medical Imaging* 32.7 (2013), pages 1153–1190.

[21] R. Shams, P. Sadeghi, R. A. Kennedy, and R. I. Hartley. “A survey of medical image registration on multicore and the GPU”. In: *IEEE Signal Processing Magazine* 27.2 (2010), pages 50–60.

[22] D. P. Shamonin, E. E. Bron, B. P. Lelieveldt, et al. “Fast parallel image registration on CPU and GPU for diagnostic classification of Alzheimer’s disease”. In: *Frontiers in Neuroinformatics* 7 (2014), page 50.

[23] F. Maes, D. Vandermeulen, and P. Suetens. “Comparative evaluation of multiresolution optimization strategies for multimodality image registration by maximization of mutual information”. In: *Medical Image Analysis* 3.4 (1999), pages 373 –386.

[24] J. Kybic and M. Unser. “Fast parametric elastic image registration”. In: *IEEE Transactions on Image Processing* 12.11 (2003), pages 1427–1442.

[25] S. Klein, M. Staring, and J. P. Pluim. “Evaluation of optimization methods for non-rigid medical image registration using mutual information and B-splines”. In: *IEEE Transactions on Image Processing* 16.12 (2007), pages 2879–2890.

[26] R. C. Hardie, K. J. Barnard, and E. E. Armstrong. “Joint MAP registration and high-resolution image estimation using a sequence of undersampled images”. In: *IEEE Transactions on Image Processing* 6.12 (1997), pages 1621–1633.

[27] P. Thevenaz and M. Unser. “Optimization of mutual information for multiresolution image registration”. In: *IEEE Transactions on Image Processing* 9.12 (2000), pages 2083–2099.

[28] S. Kabus, T. Netsch, B. Fischer, and J. Modersitzki. “B-spline registration of 3D images with Levenberg-Marquardt optimization”. In: *Medical Imaging 2004*. International Society for Optics and Photonics. 2004, pages 304–313.

[29] M. Kisaki, Y. Yamamura, H. Kim, et al. “High speed image registration of head CT and MR images based on Levenberg-Marquardt algorithms”. In: *Soft Computing and Intelligent Systems (SCIS), 2014 Joint 7th International Conference on and Advanced Intelligent Systems (ISIS), 15th International Symposium on*. IEEE. 2014, pages 1481–1485.

[30] D. Mattes, D. R. Haynor, H. Vesselle, et al. "PET-CT image registration in the chest using free-form deformations". In: *IEEE Transactions on Medical Imaging* 22.1 (2003), pages 120–128.

[31] M. Sdika. "A fast nonrigid image registration with constraints on the Jacobian using large scale constrained optimization". In: *IEEE Transactions on Medical Imaging* 27.2 (2008), pages 271–281.

[32] S. Damas, O. Cordón, and J. Santamaría. "Medical imager registration using evolutionary computation: An experimental survey". In: *IEEE Computational Intelligence Magazine* 6.4 (2011), pages 26–42.

[33] M. P. Wachowiak, R. Smolíková, Y. Zheng, et al. "An approach to multimodal biomedical image registration utilizing particle swarm optimization". In: *IEEE Transactions on Evolutionary Computation* 8.3 (2004), pages 289–301.

[34] Y.-W. Chen, C.-L. Lin, and A. Mimori. "Multimodal medical image registration using particle swarm optimization". In: *Intelligent Systems Design and Applications, 2008. ISDA'08. Eighth International Conference on*. Volume 3. IEEE. 2008, pages 127–131.

[35] A. A. Cole-Rhodes, K. L. Johnson, J. LeMoigne, and I. Zavorin. "Multiresolution registration of remote sensing imagery by optimization of mutual information using a stochastic gradient". In: *IEEE Transactions on Image Processing* 12.12 (2003), pages 1495–1511.

[36] J. C. Spall. "Implementation of the simultaneous perturbation algorithm for stochastic optimization". In: *IEEE Transactions on Aerospace and Electronic Systems* 34.3 (1998), pages 817–823.

[37] L. Bottou. "Stochastic gradient learning in neural networks". In: *Proceedings of Neuro-Nımes* 91.8 (1991).

[38] A Harju, B Barbiellini, S Siljamäki, et al. "Stochastic gradient approximation: An efficient method to optimize many-body wave functions". In: *Physical Review Letters* 79.7 (1997), page 1173.

[39] H. Kesten. "Accelerated stochastic approximation". In: *The Annals of Mathematical Statistics* (1958), pages 41–59.

[40] A Gaivoronski. *Stochastic quasigradient methods and their implementation*. 1988.

[41] Y.-H. Dai and H. Zhang. "Adaptive two-point stepsize gradient algorithm". In: *Numerical Algorithms* 27.4 (2001), pages 377–385.

[42] J. C. Spall. *Introduction to stochastic search and optimization: estimation, simulation, and control*. Volume 65. John Wiley & Sons, 2005.

[43] A. P. George and W. B. Powell. "Adaptive stepsizes for recursive estimation with applications in approximate dynamic programming". In: *Machine Learning* 65.1 (2006), pages 167–198.

[44] R. Bhagalia, J. A. Fessler, and B. Kim. "Accelerated nonrigid intensity-based image registration using importance sampling". In: *IEEE Transactions on Medical Imaging* 28.8 (2009), pages 1208–1216.

[45] Y Qiao, B. Lelieveldt, and M Staring. "Fast automatic estimation of the optimization step size for nonrigid image registration". In: *SPIE Medical Imaging*. International Society for Optics and Photonics. 2014, 90341A–90341A.

[46] D. Vysochanskij and Y. I. Petunin. "Justification of the 3σ rule for unimodal distributions". In: *Theory of Probability and Mathematical Statistics* 21 (1980), pages 25–36.

[47] J. West, J. M. Fitzpatrick, M. Y. Wang, et al. "Comparison and evaluation of retrospective intermodality brain image registration techniques". In: *Journal of Computer Assisted Tomography* 21.4 (1997), pages 554–568.

[48] J. Stolk, H. Putter, E. M. Bakker, et al. "Progression parameters for emphysema: a clinical investigation". In: *Respiratory Medicine* 101.9 (2007), pages 1924–1930.

[49] K. Murphy, B. van Ginneken, S. Klein, et al. "Semi-automatic construction of reference standards for evaluation of image registration". In: *Medical Image Analysis* 15.1 (2011), pages 71–84.

[50] M. Staring, M. Bakker, J Stolk, et al. "Towards local progression estimation of pulmonary emphysema using CT". In: *Medical physics* 41.2 (2014), page 021905.

[51] A. Hammers, R. Allom, M. J. Koepp, et al. "Three-dimensional maximum probability atlas of the human brain, with particular reference to the temporal lobe". In: *Human Brain Mapping* 19.4 (2003), pages 224–247.

[52] I. S. Gousias, D. Rueckert, R. A. Heckemann, et al. "Automatic segmentation of brain MRIs of 2-year-olds into 83 regions of interest". In: *NeuroImage* 40.2 (2008), pages 672 –684.

[53] S. Vijayan, S. Klein, E. F. Hofstad, et al. "Motion tracking in the liver: Validation of a method based on 4D ultrasound using a nonrigid registration technique". In: *Medical physics* 41.8 (2014).

[54] P. Seroul and D. Sarrut. "VV: a viewer for the evaluation of 4D image registration". In: *MIDAS Journal (Medical Image Computing and Computer-Assisted Intervention MICCAI 2008, Workshop-Systems and Architectures for Computer Assisted Interventions)*. 2008, pages 1–8.

[55] F. Maes, A. Collignon, D. Vandermeulen, et al. "Multimodality image registration by maximization of mutual information". In: *IEEE Transactions on Medical Imaging* 16.2 (1997), pages 187–198.

[56] *Life Science Grid*. URL: `https://surfsara.nl/project/life-science-grid` (visited on 08/21/2014).

[57] D. Rueckert, L. I. Sonoda, C. Hayes, et al. "Nonrigid registration using free-form deformations: application to breast MR images". In: *IEEE Transactions on Medical Imaging* 18.8 (1999), pages 712–721.

[58] C. Metz, S. Klein, M. Schaap, et al. "Nonrigid registration of dynamic medical imaging data using nD+ t B-splines and a groupwise optimization approach". In: *Medical Image Analysis* 15.2 (2011), pages 238–249.

[59] W. Sun, W. Niessen, M. van Stralen, and S. Klein. "Simultaneous multiresolution strategies for nonrigid image registration". In: *IEEE Transactions on Image Processing* 22.12 (2013), pages 4905–4917.

[60] N. N. Schraudolph and T. Graepel. "Combining conjugate direction methods with stochastic approximation of gradients". In: *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics, AISTATS 2003*. 2003.

[61] E. Laure, A. Edlund, F. Pacini, et al. *Programming the Grid with gLite*. Mar. 2006. URL: `http://cds.cern.ch/record/936685` (visited on 08/21/2014).

[62] P. Andreetto, S. Andreozzi, G. Avellino, et al. "The gLite workload management system". en. In: *Journal of Physics: Conference Series* 119.6 (July 2008), page 062007. URL: `http://iopscience.iop.org/1742-6596/119/6/062007` (visited on 08/21/2014).

[63] *EGI site*. URL: `http://www.egi.eu/` (visited on 08/21/2014).

[64] C. Marco, C. Fabio, D. Alvise, et al. “The gLite workload management system”. en. In: *Advances in Grid and Pervasive Computing*. Edited by N. Abdennadher and D. Petcu. Lecture Notes in Computer Science 5529. Springer Berlin Heidelberg, Jan. 2009, pages 256–268. URL: `http://link.springer.com/chapter/10.1007/978-3-642-01671-4_24` (visited on 08/21/2014).

[65] A. Casajus, R. Graciani, S. Paterson, et al. “DIRAC pilot framework and the DIRAC workload management system”. en. In: *Journal of Physics: Conference Series* 219.6 (Apr. 2010), page 062049. URL: `http://iopscience.iop.org/1742-6596/219/6/062049` (visited on 08/21/2014).

[66] *jjbot/picasclient*. URL: `https://github.com/jjbot/picasclient` (visited on 08/21/2014).

[67] *RP3 / Grid training | GitLab*. URL: `https://git.lumc.nl/rp3/grid_training/tree/1c654deb90d85a4c62cbc1cfac6f2fb64572a78b` (visited on 08/21/2014).

[68] *Welcome to*. URL: `https://www.python.org/` (visited on 08/21/2014).

[69] *Ganga: Gaudi/Athena and Grid Alliance*. URL: `http://ganga.web.cern.ch/ganga/` (visited on 08/21/2014).

[70] S. Klein, M. Staring, et al. “Preconditioned stochastic gradient descent optimisation for monomodal image registration”. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2011*. Springer, 2011, pages 549–556.

[71] N. N. Schraudolph, J. Yu, S. Günter, et al. “A stochastic quasi-Newton method for online convex optimization.” In: *AISTATS*. Volume 7. 2007, pages 436–443.

[72] A. Bordes, L. Bottou, and P. Gallinari. “SGD-QN: careful quasi-Newton stochastic gradient descent”. In: *The Journal of Machine Learning Research* 10 (2009), pages 1737–1754.

[73] A. Mokhtari and A. Ribeiro. “RES: Regularized stochastic BFGS algorithm”. In: *Signal Processing, IEEE Transactions on* 62.23 (2014), pages 6089–6104.

[74] R. H. Byrd, S. Hansen, J. Nocedal, and Y. Singer. “A stochastic Quasi-Newton method for large-scale optimization”. In: *arXiv preprint arXiv:1401.7020* (2014).

[75] B. O’ Donoghue and E. Candès. “Adaptive restart for accelerated gradient schemes”. English. In: *Foundations of Computational Mathematics* (2013), pages 1–18.

[76] J. Kiefer and J. Wolfowitz. “Stochastic estimation of the maximum of a regression function”. In: *The Annals of Mathematical Statistics* 23.3 (1952), pages 462–466.

[77] J. Stolk, H. Putter, E. M. Bakker, et al. “Progression parameters for emphysema: a clinical investigation”. In: *Respiratory medicine* 101.9 (2007), pages 1924–1930.

[78] F. Modarres Khiyabani and W. Leong. “Limited memory methods with improved symmetric rank-one updates and its applications on nonlinear image restoration”. English. In: *Arabian Journal for Science and Engineering* 39.11 (2014), pages 7823–7838.

[79] J. Nocedal and S. J. Wright. “Numerical optimization, second edition”. In: *Numerical optimization* (2006), pages 497–528.

[80] V. Cevher, S. Becker, and M. Schmidt. “Convex optimization for big data: Scalable, randomized, and parallel algorithms for big data analytics”. In: *Signal Processing Magazine, IEEE* 31.5 (2014), pages 32–43.

[81] R. Szeliski and J. Coughlan. “Spline-based image registration”. In: *International Journal of Computer Vision* 22.3 (1997), pages 199–218.

[82] Y. Saad and H. A. Van Der Vorst. “Iterative solution of linear systems in the 20th century”. In: *Journal of Computational and Applied Mathematics* 123.1 (2000), pages 1–33.

[83] M. Benzi. “Preconditioning techniques for large linear systems: a survey”. In: *Journal of computational Physics* 182.2 (2002), pages 418–477.

[84] A. V. Knyazev and I. Lashuk. “Steepest descent and conjugate gradient methods with variable preconditioning”. In: *SIAM Journal on Matrix Analysis and Applications* 29.4 (2007), pages 1267–1280.

[85] C. Li, C. Chen, D. Carlson, and L. Carin. “Preconditioned stochastic gradient Langevin dynamics for deep neural networks”. In: *Thirtieth AAAI Conference on Artificial Intelligence*. 2016.

[86] H. Jiang, G. Huang, P. A. Wilford, and L. Yu. “Constrained and preconditioned stochastic gradient method”. In: *IEEE Transactions on Signal Processing* 63.10 (2015), pages 2678–2691.

[87] D. E. Carlson, E. Collins, Y.-P. Hsieh, et al. “Preconditioned spectral descent for deep learning”. In: *Advances in Neural Information Processing Systems*. 2015, pages 2971–2979.

[88] Y. Dauphin, H. de Vries, and Y. Bengio. “Equilibrated adaptive learning rates for non-convex optimization”. In: *Advances in Neural Information Processing Systems*. 2015, pages 1504–1512.

[89] D. Zikic, M. Baust, A. Kamen, and N. Navab. “A general preconditioning scheme for difference measures in deformable registration”. In: *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE. 2011, pages 49–56.

[90] J. L. Morales and J. Nocedal. “Automatic preconditioning by limited memory quasi-Newton updating”. In: *SIAM Journal on Optimization* 10.4 (2000), pages 1079–1096.

[91] T. P. Minka. “A comparison of numerical optimizers for logistic regression”. In: *Unpublished draft* (2003).

[92] J. Ngiam, A. Coates, A. Lahiri, et al. “On optimization methods for deep learning”. In: *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*. 2011, pages 265–272.

[93] J. Yu, S. Vishwanathan, S. Günter, and N. N. Schraudolph. “A quasi-Newton approach to nonsmooth convex optimization problems in machine learning”. In: *Journal of Machine Learning Research* 11.Mar (2010), pages 1145–1200.

[94] R. H. Byrd, S. L. Hansen, J. Nocedal, and Y. Singer. “A stochastic quasi-Newton method for large-scale optimization”. In: *SIAM Journal on Optimization* 26.2 (2016), pages 1008–1031.

[95] F. Maes, D. Vandermeulen, and P. Suetens. “Comparative evaluation of multiresolution optimization strategies for multimodality image registration by maximization of mutual information”. In: *Medical Image Analysis* 3.4 (1999), pages 373–386.

[96] D. Mattes, D. R. Haynor, H. Vesselle, et al. “PET-CT image registration in the chest using free-form deformations”. In: *IEEE transactions on medical imaging* 22.1 (2003), pages 120–128.

[97] Y. Qiao, Z. Sun, B. P. Lelieveldt, and M. Staring. “A stochastic quasi-Newton method for non-rigid image registration”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer International Publishing. 2015, pages 297–304.

[98] W. Li, D. A. Jaffray, G. Wilson, and D. Moseley. “How long does it take? An analysis of volumetric image assessment time”. In: *Radiotherapy and Oncology* 119.1 (2016), pages 150–153.

[99] T. N. Sainath, L. Horesh, B. Kingsbury, et al. “Accelerating Hessian-free optimization for deep neural networks by implicit preconditioning and sampling”. In: *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE. 2013, pages 303–308.

[100] Y. Qiao, B. van Lew, B. P. Lelieveldt, and M. Staring. “Fast automatic step size estimation for gradient descent optimization of image registration”. In: *IEEE Transactions on Medical Imaging* 35.2 (2016), pages 391–403.

[101] C. A. Cocosco, V. Kollokian, R. K.-S. Kwan, et al. “Brainweb: Online interface to a 3D MRI simulated brain database”. In: *NeuroImage*. Citeseer. 1997.

[102] S. M. Smith. “Fast robust automated brain extraction”. In: *Human Brain Mapping* 17.3 (2002), pages 143–155.

[103] W. Sun, D. H. Poot, I. Smal, et al. “Stochastic optimization with randomized smoothing for image registration”. In: *Medical Image Analysis* 35 (2017), pages 146–158.

[104] P. Castadot, J. A. Lee, A. Parraga, et al. “Comparison of 12 deformable registration strategies in adaptive radiation therapy for the treatment of head and neck tumors”. In: *Radiotherapy and Oncology* 89.1 (2008), pages 1–12.

[105] A. Kumarasiri, F. Siddiqui, C. Liu, et al. “Deformable image registration based automatic CT-to-CT contour propagation for head and neck adaptive radiotherapy in the routine clinical setting”. In: *Medical Physics* 41.12 (2014), page 121712.

[106] J. A. Shackleford, N Kandasamy, and G. Sharp. “On developing B-spline registration algorithms for multi-core processors”. In: *Physics in Medicine and Biology* 55.21 (2010), page 6329.

[107] D. Yang, S. Brame, I. El Naqa, et al. “Technical Note: DIRART–A software suite for deformable image registration and adaptive radiotherapy research”. In: *Medical Physics* 38.1 (2011), pages 67–77.

[108] D. C. Ince, L. Hatton, and J. Graham-Cumming. “The case for open computer programs”. In: *Nature* 482.7386 (2012), pages 485–488.

[109] B. C. Lowekamp, D. T. Chen, L. Ibáñez, and D. Blezek. “The design of SimpleITK”. In: *Frontiers in Neuroinformatics* 7.45 (2013).

[110] M. Thor, E. S. Andersen, J. B. Petersen, et al. “Evaluation of an application for intensity-based deformable image registration and dose accumulation in radiotherapy”. In: *Acta Oncologica* 53.10 (2014), pages 1329–1336.

[111] G. Cazoulat, A. Simon, A. Dumenil, et al. “Surface-constrained nonrigid registration for dose monitoring in prostate cancer radiotherapy”. In: *IEEE Transactions on Medical Imaging* 33.7 (2014), pages 1464–1474.

[112] M. Nassef, A. Simon, G. Cazoulat, et al. “Quantification of dose uncertainties in cumulated dose estimation compared to planned dose in prostate IMRT”. In: *Radiotherapy and Oncology* 119.1 (2016), pages 129–136.

[113] T. Zhang, Y. Chi, E. Meldolesi, and D. Yan. “Automatic delineation of on-line head-and-neck computed tomography images: toward on-line adaptive radiotherapy”. In: *International Journal of Radiation Oncology* Biology* Physics* 68.2 (2007), pages 522–530.

[114] G. Landry, R. Nijhuis, G. Dedes, et al. “Investigating CT to CBCT image registration for head and neck proton therapy as a tool for daily dose recalculation”. In: *Medical Physics* 42.3 (2015), pages 1354–1366.

[115] P. Castadot, X. Geets, J. A. Lee, et al. “Assessment by a deformable registration method of the volumetric and positional changes of target volumes and organs at risk in pharyngo-laryngeal tumors treated with concomitant chemo-radiation”. In: *Radiotherapy and Oncology* 95.2 (2010), pages 209–217.

[116] P. Kupelian, T. Willoughby, A. Mahadevan, et al. “Multi-institutional clinical experience with the Calypso system in localization and continuous, real-time monitoring of the prostate gland during external radiotherapy”. In: *International Journal of Radiation Oncology* Biology* Physics* 67.4 (2007), pages 1088–1098.

[117] H. Ariyaratne, H. Chesham, J. Pettingell, and R. Alonzi. “Image-guided radiotherapy for prostate cancer with cone beam CT: dosimetric effects of imaging frequency and PTV margin”. In: *Radiotherapy and Oncology* 121.1 (2016), pages 103–108.

[118] A. Godley, E. Ahunbay, C. Peng, and X. A. Li. “Automated registration of large deformations for adaptive radiation therapy of prostate cancer”. In: *Medical Physics* 36.4 (2009), pages 1433–1441.

[119] D. Robb, A. Plank, and M. Middleton. “Assessing the efficiency and consistency of daily image-guided radiation therapy in a modern radiotherapy centre”. In: *Journal of Medical Imaging and Radiation Sciences* 45.2 (2014), pages 72–78.

[120] G. Sharp, N Kandasamy, H Singh, and M. Folkert. “GPU-based streaming architectures for fast cone-beam CT image reconstruction and Demons deformable registration”. In: *Physics in Medicine and Biology* 52.19 (2007), page 5771.

[121] L. P. Muren, E. Wasbø, S. I. Helle, et al. “Intensity-modulated radiotherapy of pelvic lymph nodes in locally advanced prostate cancer: planning procedures and early experiences”. In: *International Journal of Radiation Oncology* Biology* Physics* 71.4 (2008), pages 1034–1041.

[122] W. Huizinga, S. Klein, and D. H. Poot. “Fast multidimensional B-spline interpolation using template metaprogramming”. In: *Biomedical Image Registration*. Springer, 2014, pages 11–20.

[123] P. Thevenaz and M. Unser. “Optimization of mutual information for multiresolution image registration”. In: *IEEE Transactions on Image Processing* 9.12 (2000), pages 2083–2099.

[124] C. Lu, S. Chelikani, X. Papademetris, et al. “An integrated approach to segmentation and nonrigid registration for application in image-guided pelvic radiotherapy”. In: *Medical Image Analysis* 15.5 (2011), pages 772–785.

[125] S. Breedveld, P. R. Storchi, P. W. Voet, and B. J. Heijmen. “iCycle: Integrated, multicriterial beam angle, and profile optimization for generation of coplanar and noncoplanar IMRT plans”. In: *Medical Physics* 39.2 (2012), pages 951–963.

[126] S. van de Water, A. Kraan, S. Breedveld, et al. “Improved efficiency of multi-criteria IMPT treatment planning using iterative resampling of randomly placed pencil beams”. In: *Physics in Medicine and Biology* 58.19 (2013), page 6969.

[127] M Moteabbed, A Trofimov, G. Sharp, et al. “Proton therapy of prostate cancer by anterior-oblique beams: implications of setup and anatomy variations”. In: *Physics in Medicine and Biology* 62.5 (2017), page 1644.

[128] S. Thörnqvist, L. Bentzen, J. B. Petersen, et al. “Plan robustness of simultaneous integrated boost radiotherapy of prostate and lymph nodes for different image-guidance and delivery techniques”. In: *Acta Oncologica* 50.6 (2011), pages 926–934.

[129] L. Bondar, M. Hoogeman, J. W. Mens, et al. “Toward an individualized target motion management for IMRT of cervical cancer based on model-predicted cervix–uterus shape and position”. In: *Radiotherapy and Oncology* 99.2 (2011), pages 240–245.

[130] A. Qin, Y. Sun, J. Liang, and D. Yan. “Evaluation of online/offline image guidance/adaptation approaches for prostate cancer radiation therapy”. In: *International Journal of Radiation Oncology* Biology* Physics* 91.5 (2015), pages 1026–1033.

[131] G. Gunay, L. M. Ha, T. van Walsum, and S. Klein. “Semi-automated registration of pre- and intra-operative liver CT for image-guided interventions”. In: *SPIE Medical Imaging*. International Society for Optics and Photonics. 2016, 97841N–97841N.

[132] S. Gao, L. Zhang, H. Wang, et al. “A deformable image registration method to handle distended rectums in prostate cancer radiotherapy”. In: *Medical physics* 33.9 (2006), pages 3304–3312.

[133] G. Saygili, M. Staring, and E. A. Hendriks. “Confidence estimation for medical image registration based on stereo confidences”. In: *IEEE Transactions on Medical Imaging* 35.2 (2016), pages 539–549.

[134] H. Sokooti, G. Saygili, B. Glocker, et al. “Accuracy Estimation for Medical Image Registration Using Regression Forests”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2016, pages 107–115.

[135] K. H. Cha, L. Hadjiiski, R. K. Samala, et al. “Urinary bladder segmentation in CT urography using deep-learning convolutional neural network and level sets”. In: *Medical physics* 43.4 (2016), pages 1882–1896.

[136] C. Veiga, G. Janssens, C.-L. Teng, et al. “First clinical investigation of cone beam computed tomography and deformable registration for adaptive proton therapy for lung cancer”. In: *International Journal of Radiation Oncology* Biology* Physics* 95.1 (2016), pages 549–559.

[137] W. Sun, D. H. Poot, I. Smal, et al. “Stochastic optimization with randomized smoothing for image registration”. In: *Medical Image Analysis* 35 (2017), pages 146 –158.

[138] J. Duchi, E. Hazan, and Y. Singer. “Adaptive subgradient methods for online learning and stochastic optimization”. In: *Journal of Machine Learning Research* 12.Jul (2011), pages 2121–2159.

[139] M. D. Zeiler. “ADADELTA: An adaptive learning rate method”. In: *CoRR* abs/1212.5701 (2012). URL: `http://arxiv.org/abs/1212.5701`.

[140] D. Kingma and J. Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[141] N. N. Schraudolph and T. Graepel. “Conjugate directions for stochastic gradient descent”. In: *International Conference on Artificial Neural Networks*. Springer. 2002, pages 1351–1356.

[142] R. Johnson and T. Zhang. “Accelerating stochastic gradient descent using predictive variance reduction”. In: *Advances in Neural Information Processing Systems*. 2013, pages 315–323.

[143] Y. Nesterov. “A method for unconstrained convex minimization problem with the rate of convergence O (1/k2)”. In: *Doklady an SSSR*. Volume 269. 3. 1983, pages 543–547.

[144] N. Qian. “On the momentum term in gradient descent learning algorithms”. In: *Neural networks* 12.1 (1999), pages 145–151.

[145] Y. Liu, R. Jin, M. Chen, et al. “Contour propagation using non-uniform cubic B-splines for lung tumor delineation in 4D-CT”. In: *International journal of computer assisted radiology and surgery* 11.12 (2016), pages 2139–2151.

[146] W. Sun, W. J. Niessen, and S. Klein. “Randomly perturbed B-splines for nonrigid image registration”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PP.99 (2016), pages 1–1.

[147] J. Kallwies, T. Engler, and H.-J. Wuensche. “A fast and accurate image-registration algorithm using prior knowledge”. In: *Digital Image Computing: Techniques and Applications (DICTA), 2016 International Conference on*. IEEE. 2016, pages 1–8.

Appendix

Parameter names of implementations in `elastix`

Chapter 2: fast adaptive stochastic gradient descent (FASGD)

```
(Optimizer "AdaptiveStochasticGradientDescent")
(ASGDParameterEstimationMethod "DisplacementDistribution")
```

Chapter 3: stochastic L-BFGS (s-LBFGS)

```
(Optimizer "AdaptiveStochasticLBFGS")
(StepSizeStrategy "Adaptive")
(CurvatureSampler "Random")
(NumberOfInnerLoopSamples 50000)
```

Chapter 4: fast preconditioned stochastic gradient descent (FPSGD)

```
(Optimizer "PreconditionedStochasticGradientDescent")
(NumberOfSamplesForPrecondition 50000)
(RegularizationKappa 0.6)
(ConditionNumber 1)
```

Publications

Journal articles

Y. Qiao, B.P.F. Lelieveldt and M. Staring. An efficient preconditioner for stochastic gradient descent optimization of image registration, submitted.

Y. Qiao, T. Jagt, M. Hoogeman, B.P.F. Lelieveldt and M. Staring. Evaluation of an open source registration package for automatic contour propagation in online adaptive intensity-modulated proton therapy of prostate cancer, submitted.

Y. Qiao, B. van Lew, B.P.F. Lelieveldt and M. Staring. Fast automatic step size estimation for gradient descent optimization of image registration, *IEEE Transactions on Medical Imaging*, Volume 35, Issue 2, Pages 391–403, 2016.

International conference proceedings

Y. Qiao, Z. Sun, B.P.F. Lelieveldt and M. Staring. A stochastic quasi-Newton method for non-rigid image registration, *Medical Image Computing and Computer-Assisted Intervention - MICCAI*, Volume 9350, Pages 297–304, 2015.

Y. Qiao, B.P.F. Lelieveldt and M. Staring. Fast automatic estimation of the optimization step size for nonrigid image registration, *SPIE Medical Imaging: Image Processing - SPIE*, Volume 9034, Pages 90341A, 2014.

Abstracts

Y. Qiao, Z. Sun, B.P.F. Lelieveldt and M. Staring. A stochastic quasi-Newton method for non-rigid image registration, *Dutch society of Pattern recognition and Image Processing - NVPHBV*, 2016.

Acknowledgements

Research is a procedure of re-search in the field of a certain topic.

It is a great pleasure to pursue my Ph.D. in Leiden University at LKEB. I would like to express my sincere gratitude and appreciations to my promotor Prof.dr.ir. Boudewijn Lelieveldt and my co-promotor Dr. ir. Marius Staring. Without their insightful and enthusiasm guidance in both scientific research and my personal life, it would be impossible for me to accomplish my Ph.D. study. And many thanks to my thesis committee and defense committee for your careful and detailed comments and suggestions.

I would like to thank all the persons who gave me selfless helps and supports. My first thank goes to Boudewijn, who provided me the position and gave me lots of freedom in research. Your endless passion and enthusiasm in the research of t-SNE and brain image analysis encouraged me a lot, and made me find out the meaning of the research life.

Marius, you are the one who I should send my supreme respect and thanks. I am your first fully-supervised Ph.D. student and you are the best daily supervisor to me. Your unreserved instructions and supervision helped me a lot during my studies and researches. Your rigorous attitude and critical comments urged me to be better and better. I really enjoyed the time we discussing the pros and cons of the new algorithms. Your working attitude "Your holiday is your holiday, you should take it." also inspired me a lot. And many blessings to your son Yiguang and your daughter Ying.

Here, I would like to acknowledge Dr. Berend C. Stoel, who provided many detailed discussions and instructions in my daily research life. I also enjoyed a lot on your Wednesday music concert together with Els Bakker, Leo Wolf. Your attitude and efforts in piano and sax made me realize that life can also be colorful after working time.

It was happy to work with LKEB people. Zhuo Sun, thanks for your kindly help and useful discussion everyday. I enjoyed the time working with you for four years. You have so many great ideas related to brain image processing, and I hope we could work together for more opportunities. Baldur van Lew, thanks for your support in LifeScienceGrid and your push during that tough time. Without your help, I didn't know if I could make it and had the paper published. Floris Berendsen, many thanks to your kindly helps and your useful tools for image processing. Good luck with your GPU speedup of `elastix` and deep learning methods, hope to collaborate with you a lot later. Denis Shamonin, you are so nice and helpful for providing many solutions regarding MevisLab and Python. Walid Abdelmoula, we had a long time very nice talk and discussions and you really did a very nice work in the field of registration. Gorkem Saygili and Hessam Sokooti, we had a very nice time to discuss and meet together in registration meeting. I would like to thank my colleagues in LKEB, Reiber, Rob, Jouke, Oleh, Ahmed, Patrick, Jeroen, Pieter, Alexander, Paulien, Rahil, Mohammed, Qian, Michéle, Niels, Leo. You are so nice people and provide so many grateful time in the lab. Shengnan, Xinpei, Shan, Qin, Zhiwei, Yingguang, Yuanbo, it was very happy to meet you in LKEB and we also enjoyed many good time in spare time. All in all, you were the solid supports for me in the solo science research path.

During the four years Ph.D. study, we also had a regular RAMBO meeting with EMC people, from which I learned a lot. It was really very nice to meet Dr. Stefan Klein, the main co-author of `elastix`. You are very wise and humble. Many thanks to you for your thoughtful ideas and kind helps. Dr. Wei Sun, I really appreciated your supports and helps no matter where you are. Thanks very much for giving many detailed suggestions. It was very nice to have discussions and brainstorms with Wyke, Gokhan, Jean-Marie and other RAMBOers.

I would also like to thank my co-authors for your contributions in the publications. Mischa and Thyrza, you are so nice and helpful in the ADAPTNOW project. I enjoyed and learned a lot in this project. Thanks for your patience and your careful suggestions and comments.

For the postdoc life in LONI, I really enjoy the new life in LA, its very nice weather, and my new chapter in brain image analysis. Thanks Dr. Yonggang Shi for providing me the position and opportunity. It is very happy to meet Jin Kyu Gahm, Dogu Baran Aydogan, Junyan Wang, Yuchun Tang, Sumiko Abe, Lu Zhao, Kaida Ning, Xingfeng Shao, Yelong Shen. And now I can work together with Wei Sun for brain image analysis.

Four years life in Leiden was a wonderful experience for me. Leiden is a very peaceful and historical city. The people I met here are really unforgettable. It was very nice to live in Olivabank during my whole study with Meng Sun, Min He, Jian Yang and Jinxian Wang, and I enjoyed a sincere friendship and appreciated it. And I also had good time with a lot of Chinese friends met in the Netherlands, Jinfeng Cao, Zhiwei Yang, Guangsheng Du, Zhenyu Xiao, Yaojin Peng, Fuyu Cai, Depang Zhao, Jianqiu Gong, Nan He, Yanmin Guo, Hua Ma, Chaoping Zhang, Yao Yao, Jiong Zhang, Tao Tan *et al.*

Finally, I would give my sincere respect and thanks to my family. Thanks my mother and my father for their kindly helps and supports for my study. Thanks my elder sister and brother-in-law to take care of my parents when I studied abroad, and also my nephew and niece giving our family so many happiness. I would also like to thank my wife, Jiangli, for her unconditional supports and selfless love.

Curriculum Vitae

Yuchuan Qiao was born in Hubei, China in 1986. In 2005, he started his studying in the major of Electrical Engineering at Hunan University and obtained a bachelor degree in the major of Automation in 2009. At the same year, he began his master study in the major of Control Science and Engineering. During his master study, he involved in the research of teleoperation techniques of cleaning robot for condenser in the power plant. In 2012, he obtained his master degree.

After his master study in China, he started his Ph.D. study in the Division of Image Processing (LKEB) under the Department of Radiology at Leiden University Medical Center in the Netherlands from September 2012. He first began with the project of "Fast Image Registration for Time-critical Medical Applications", then involved in the ADAPTNOW project to develop high-precision cancer treatment by online adaptive proton therapy. His work mainly focuses on new optimization schemes and its application in clinic, which results this thesis.

Since March 2017, he works as a post-doctoral researcher at the LONI in the field of brain image analysis for Alzheimer disease.