UNIVERSITÄT KOBLENZ · LANDAU
Fachbereich 4: Informatik

Multi-modal 3D Cochlea Images

Registration, Fusion, Segmentation and Analysis

by
Ibraheem Al-Dhamari
idhamari@uni-koblenz.de

Approved Dissertation thesis for the partial fulfilment of the requirements for a
Doctor of Natural Sciences (Dr. rer. nat.)
Fachbereich 4: Informatik
Universität Koblenz-Landau

Chair of PhD Board:	**Prof. Dr. Ralf Lämmel**
Chair of PhD Commission:	**Prof. Dr. Thomas Burkhardt**
Examiner and Supervisor:	**Prof. Dr.-Ing. Dietrich Paulus**
Further Examiners:	
	Prof. Dr.-Ing. Uwe Baumann
	PD. Dr.-Ing. habil. Thomas Wittenberg

Date of the doctoral viva: 14.9.2021

Abstract

Efficient Cochlear Implant (CI) surgery requires prior knowledge of the cochlea's size and its characteristics. This information helps to select suitable implants for different patients. Registered and fused images helps doctors by providing more informative image that takes advantages of different modalities. The cochlea's small size and complex structure, in addition to the different resolutions and head positions during imaging, reveals a big challenge for the automated registration of the different image modalities. To obtain an automatic measurement of the cochlea length and the volume size, a segmentation method of cochlea medical images is needed.

The goal of this dissertation is to introduce new practical and automatic algorithms for the human cochlea multi-modal 3D image registration, fusion, segmentation and analysis. Two novel methods for automatic cochlea image registration (ACIR) and automatic cochlea analysis (ACA) are introduced. The proposed methods crop the input images to the cochlea part and then align the cropped images to obtain the optimal transformation. After that, this transformation is used to align the original images. ACIR and ACA use Mattes mutual information as similarity metric, the adaptive stochastic gradient descent (ASGD) or the stochastic limited memory Broyden–Fletcher–Goldfarb–Shanno (s-LBFGS) optimizer to estimate the parameters of 3D rigid transform. The second stage of non-rigid registration estimates B-spline coefficients that are used in an atlas-model-based segmentation to extract cochlea scalae and the relative measurements of the input image. The image which has segmentation is aligned to the input image to obtain the non-rigid transformation. After that the segmentation of the first image, in addition to point-models are transformed to the input image. The detailed transformed segmentation provides the scala volume size. Using the transformed point-models, the A-value, the central scala lengths, the lateral and the organ of corti scala tympani lengths are computed.

The methods have been tested using clinical 3D images of total 67 patients: from Germany (41 patients) and Egypt (26 patients). The patients are of different ages and gender. The number of images used in the experiments is 217, which are multi-modal 3D clinical images from CT, CBCT, and MRI scanners.

The proposed methods are compared to the state of the arts optimizers related medical image registration methods e.g. fast adaptive stochastic gradient descent (FASGD) and efficient preconditioned stochastic gradient descent (EPSGD). The comparison used the root mean squared distance (RMSE) between the ground truth landmarks and the resulted landmarks. The landmarks are located manually by two experts to represent the round window and the top of the cochlea. After obtaining the transformation using ACIR, the landmarks of the moving image are transformed using the resulted transformation and RMSE of the transformed

landmarks, and at the same time the fixed image landmarks are computed. I also used the active length of the cochlea implant electrodes to compute the error aroused by the image artifact, and I found out an error ranged from 0.5 mm to 1.12 mm.

ACIR method's RMSE average was 0.36 mm with a standard deviation (SD) of 0.17 mm. The total time average required for registration of an image pair using ACIR was 4.62 seconds with SD of 1.19 seconds. All experiments are repeated 3 times for justifications. Comparing the RMSE of ACIR2017 and ACIR2020 using paired T-test shows no significant difference (p-value = 0.17).

The total RMSE average of ACA method was 0.61 mm with a SD of 0.22 mm. The total time average required for analysing an image was 5.21 seconds with SD of 0.93 seconds.

The statistical tests show that there is no difference between the results from automatic A-value method and the manual A-value method (p-value = 0.42). There is no difference also between length's measurements of the left and the right ear sides (p-value $>$ 0.16).

Comparing the results from German and Egypt dataset shows there is no difference when using manual or automatic A-value methods (p-value $>$ 0.20). However, there is a significant difference when using ACA2000 method between the German and the Egyptian results (p-value $<$ 0.001).

The average time to obtain the segmentation and all measurements was 5.21 second per image. The cochlea scala tympani volume size ranged from 38.98 mm^3 to 57.67 mm^3. The combined scala media and scala vestibuli volume size ranged from 34.98 mm^3 to 49.3 mm^3. The overall volume size of the cochlea should range from 73.96 mm^3 to 106.97 mm^3. The lateral wall length of scala tympani ranged from 42.93 mm to 47.19 mm. The organ-of-Corti length of scala tympani ranged from 31.11 mm to 34.08 mm. Using the A-value method, the lateral length of scala tympani ranged from 36.69 mm to 45.91 mm. The organ-of-Corti length of scala tympani ranged from 29.12 mm to 39.05 mm.

The length from ACA2020 method can be visualised and has a well-defined endpoints. The ACA2020 method works on different modalities and different images despite the noise level or the resolution. In the other hand, the A-value method works neither on MRI nor noisy images. Hence, ACA2020 method may provide more reliable and accurate measurement than the A-value method.

The source-code and the datasets are made publicly available to help reproduction and validation of my result.

Zusammenfassung

Für eine effiziente Cochlea-Implantation wird Wissen über die Cochlea-Größe und Charakteristik vorausgesetzt. Dadurch können zum Patienten passende Implantate gefunden werden. Erfasste und zusammengeführte Bilder unterstützen Ärzte, durch informativere Bilder, die einen Vorteil aus verschiedenen Eigenschaften ziehen. Die kleine Größe und komplexe Struktur der Cochlea, ebenso wie die verschiedenen Auflösungen und Kopfpositionen während der Bildaufnahme, stellen eine große Herausforderung für die automatische Bilderfassung der verschiedenen Bildeigenschaften dar. Um eine automatische Ausmessung der Länge und des Volumens der Cochlea zu erhalten, wird eine Methode zur Segmentierung von medizinischen Cochlea-Bildern benötigt.

Das Ziel dieser Dissertation ist die Einführung eines neu anwendbaren und automatischen Algorithmus zur multimodalen 3D Bilderfassung, Zusammenführung, Segmentierung und Analyse der menschlichen Cochlea. Zwei neue Methoden zur automatischen Bilderfassung (ACIR) und Cochlea Analyse (ACA) werden eingeführt. Die vorgeschlagenen Methoden reduzieren das Eingangsbild auf die Cochlea und richten den Bildausschnitt aus, um eine optimale Transformation zu erhalten. Danach wird die Transformation genutzt, um die originalen Bilder auszurichten. ACIR und ACA nutzen Mattes Mutual Information als Ähnlichkeitsmetrik, den adaptiven stochastischen Gradientenabstieg (ASGD) oder den stochastisch speicherbegrenzenden Broyden-Flecher-Goldfarb-Shanno (s-LBFGS) Optimierer, um die Parameter der 3D starren Transformation abzuschätzen. Die zweite Phase der nicht-starren Bilderkennung schätzt den B-Spline Koeffizienten, der in einer Atlas-modellbasierten Segmentierung genutzt wird, um die Cochlea Größe und die relativen Werte des Eingangsbildes zu extrahieren. Das Bild mit Segmentierung wird nach dem Eingangsbild ausgerichtet, um die nicht-starre Transformation zu erhalten. Danach werden die Segmentierung des ersten Bildes und das Punktmodel in das Eingangsbild zurück transformiert. Die detailliertere Segmentierung stellt die Scala Volume Größe zur Verfügung. Mithilfe des transformierten Punktmodells, des A-Wertes und der mittigen Scala Länge, werden das Lateral und die Länge des Corti-Organs Scala-tympani berechnet.

Die Methode wurde mit klinischen 3D Bilder von insgesamt 67 Patienten aus Deutschland (41 Patienten) und Ägypten (26 Patienten) getestet. Die Patienten sind unterschiedlich alt und haben verschiedene Gender. In dem Versuch wurden 217 Bilder genutzt- dies sind multimodale klinische 3D Bilder aufgenommen von CT, CBCT und MRI Scannern.

Die vorgeschlagenen Methoden werden mit aktuellen Optimierern bezüglich medizinischer Bilderfassungsmethoden z.B. dem schnelle adaptive stochastische Gradientenabstieg (FASGD) und dem effizient bedingte stochastische Gradientenabstieg (EPSGD), verglichen. Der Vergleich nutzt die mittlere quadratische Abwe-

ichung der Wurzel des Abstands (RMSE) zwischen den tatsächlichen Merkmalen und den resultierenden Merkmalen. Die tatsächlichen Merkmale wurden manuell durch zwei Experten lokalisiert, um das runde Fenster und die Spitze der Cochlea zu erfassen. Nachdem die Transformation durch ACIR bestimmt wurde, werden die Merkmale des bewegten Bildes transformiert, indem resultierende Transformation und RMSE der transformierten Merkmale genutzt werden und gleichzeitig die fixen Bildmerkmale berechnet werden. Ebenso wird die aktive Länge der Cochlea Implantats Elektroden berechnet, um den Fehler, der durch Bildartifakte entsteht, zu berücksichtigen. Der Fehler beträgt zwischen 0.5 mm bis 1.12 mm.

Der RMSE-Mittelwert der ACIR Methode war 0.36 mm mit einer Standartabweichung (SD) von 0.17 mm. Die durchschnittliche Gesamtzeit, die für die Erfassung eines Bildpaare unter Verwendung von ACIR benötigt wird, beträgt 4.62 Sekunden mit einer SD von 1.19 Sekunden. Alle Versuche wurden dreimal wiederholt. Der Vergleich des RMSE von ACIR2017 und ACIR2020 unter Verwendung des gepaarten T-Tests zeigt keinen signifikanten Unterschied (p-Wert = 0.17).

Der Durschnitt des gesamten RMSE der ACA Methode betrug 0.61 mm mit einer SD von 0.22 mm. Die durchschnittliche Gesamtzeit für die Analyse eins Bildes war 5.21 Sekunden mit einer SD von 0.93 Sekunden.

Die statistischen Tests zeigen, dass es keinen Unterschied zwischen den Ergebnissen der automatischen A-Wert-Methode und der manuellen A-Wert-Methode gibt (p-Wert = 0.42). Es gibt auch keinen Unterschied zwischen den Längenmessungen der linken und der rechten Ohrseite (p- Wert $>$ 0.16). Der Vergleich der Ergebnisse aus deutschen und ägyptischen Datensätzen zeigt, dass es keinen Unterschied bei der Verwendung manueller oder automatischer A-Wert-Methoden gibt (p-Wert $>$ 0.20). Es gibt jedoch einen signifikanten Unterschied bei der Verwendung der ACA2000-Methode zwischen den deutschen und den ägyptischen Ergebnissen (p-Wert $<$ 0.001)

Die durchschnittliche Zeit zur Segmentierung und dem Berechnen aller Werte betrug 5.21 Sekunden pro Bild Die Cochlea Scala tympani Größe variiert zwischen 38.98 mm^3 und 57.67 mm^3. Das kombinierte Scala media und Scala vestibuli Volumen variiert zwischen 34.98 mm^3 und 49.3 mm^3. Die Laterallänge der Scala tympani ändert sich zwischen 42.93 mm und 47.19 mm. Die Länge des Gehörgangs der Scala tympani wechselt zwischen 31.11 mm und 34.08 mm. Mit der A-Wert Methode variieren die Lateralwerte der Scala tympani zwischen textbf36.69 mm und 45.91 mm. Die Länge des Corti Organs der Scala tympani variiert zwischen 29.12 mm und 39.05 mm.

Die Länge der ACA2020 Methode kann visualisiert werden und hat wohldefinierte Endpunkte. Die ACA2020 Methode arbeitet mit verschiedenen Eigenschaften und Bildtyen trotz Rauschen und unabhängig von der Auflösung. Hingegen arbeitet die A-Wert Methode weder mit MRI Bildern noch mit verrauschten

Bildern. Dadurch ergibt sich, dass die ACA2020 Methode möglicherweise zuverlässiger arbeitet und genauere Werte liefert als die A-Wert Methode.

Der Quellcode und das Datenset sind öffentlich zugänglich, um eine Nachbildung zu ermöglichen und die Arbeit zu bewerten.

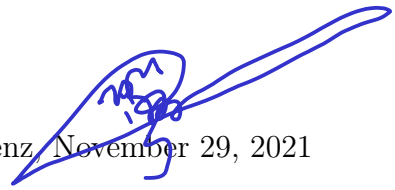
Declaration

I confirm that:

- this dissertation represents my own work.
- the contribution my supervisor and others to the research and to the dissertation was consistent with normal scientific practice.
- external contributions to the research are acknowledged.
- parts of this dissertation have been submitted and accepted for publication in advance of submission of the dissertation for examination. All references to these parts are provided in the reference section.
- I have neither submitted this or another dissertation/essay at another university as a dissertation.

I agree to publish this dissertation in the library. yes ☒ no ☐

I agree to publish this dissertation on the Internet. yes ☒ no ☐



Koblenz, November 29, 2021

List of Publications

1. Rania Helal, Roland Jacob, Marwa Elshinnawy, Amal Othman, Ibraheem Al-Dhamari, Dietrich Paulus, and Tougan Abdelaziz: "Cone-beam CT versus Multidetector CT in Postoperative Cochlear Implant Imaging: Evaluation of Image Quality and Radiation Dose". American Journal Of Neuroradiology (AJNR),42(2021), Nr. 2, S. 362–367.

2. (Abstract) Ibraheem Al-Dhamari, Sabine Bauer, Eva Keller, and Dietrich Paulus: "Automatic Detection of Cervical Spine Ligaments Origin and Insertion Points". In: Maier, Andreas (Hrsg.); Deserno, Thomas (Hrsg.); Handels, Heinz (Hrsg.); Maier-Hein, Klaus (Hrsg.); Palm, Christoph (Hrsg.). Bildverarbeitung für die Medizin 2020. Informatik aktuell. (2020), Springer Vieweg, Wiesbaden. S. 52-52.

3. Ibraheem Al-Dhamari, Sabine Bauer, Eva Keller, and Dietrich Paulus: "Automatic Detection of Cervical Spine Ligaments Origin and Insertion Points". IEEE 16^{th} International Symposium on Biomedical Imaging (ISBI 2019), 2019, S. 48–51.

4. (Abstract) Ibraheem Al-Dhamari, Sabine Bauer, Dietrich Paulus, Rania Helal, Freidrich Lissek, and Roland Jacob: "Automatic Estimation of Cochlear Duct Length and Volume Size". In: Maier, Andreas (Hrsg.); Deserno, Thomas (Hrsg.); Handels, Heinz (Hrsg.); Maier-Hein, Klaus (Hrsg.); Palm, Christoph (Hrsg.). Bildverarbeitung für die Medizin 2019. Informatik aktuell. (2019) Springer Vieweg, Wiesbaden. S. 55-55.

5. Ibraheem Al-Dhamari, Sabine Bauer, Dietrich Paulus, Rania Helal, Friedrich Lisseck, and Roland Jacob: "Automatic Cochlear Length and Volume Size Estimation". In: Stoyanov, Danail et. al. (Hrsg.). OR 2.0 Context Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis (MICCAI2018). Springer International Publishing, 2018. ISBN 978–3–030–01201–4, S. 54–61.

6. Ibraheem Al-Dhamari, Sabine Bauer, and Dietrich Paulus: "Automatic Multimodal Cervical Spine Image Atlas Segmentation". In: Maier, Andreas (Hrsg.); Deserno, Thomas (Hrsg.); Handels, Heinz (Hrsg.); Maier-Hein, Klaus (Hrsg.); Palm, Christoph (Hrsg.). Bildverarbeitung für die Medizin 2018. Informatik aktuell. (2018) Springer Vieweg, Berlin, Heidelberg. S. 303-308.

7. (Abstract) Ibraheem Al-Dhamari, Sabine Bauer, Dietrich Paulus, and Roland Jacob: "Automatic Cochlea Multi-modal Images Segmentation Using Adaptive Stochastic Gradient Descent". CI2018 DC Emerging Issues in Cochlear Implantation, March 7-10, (2018), Washington DC, USA.

8. (Abstract) Ibraheem Al-Dhamari, Eva Keller, Sabine Bauer, and Dietrich Paulus: "Cervical Spine Automatic Segmentation Using Adaptive Stochastic Gradient Descent". 12. Jahrestagung der Deutschen Wirbelsäulengesellschaft, 30.11-2.12.2017, Stuttgart, Germany.

9. (Abstract) Ibraheem Al-Dhamari, Sabine Bauer, Dietrich Paulus, Friedrich Lissek, and Roland Jacob: "ACIR: automatic cochlea image registration". In: Maier, Andreas (Hrsg.); Deserno, Thomas (Hrsg.); Handels, Heinz (Hrsg.); Maier-Hein, Klaus (Hrsg.); Palm, Christoph (Hrsg.). Bildverarbeitung für die Medizin 2017. Informatik aktuell. Springer Vieweg, Berlin, Heidelberg (2017). S. 140-140

10. (Abstract) Ibraheem Al-Dhamari, Sabine Bauer, Dietrich Paulus, and Roland Jacob: "Automatic Cochlea Segmentation Using Diffusion Snakes". 15^{th} International Symposium on Cochlear Implants in Children, July 26–29, (2017), San Francisco, USA.

11. Ibraheem Al-Dhamari, Sabine Bauer, Dietrich Paulus, Friedrich Lissek, and Roland Jacob: "ACIR: automatic cochlea image registration". Proceedings of SPIE Medical Imaging 2017, Image Processing, Orlando, USA. 10133 (2017), Nr. 10, S. 47-67.

12. (Abstract) Ibraheem Al-Dhamari, Sabine Bauer, Dietrich Paulus, Friedrich Lissek and, Roland Jacob: "Automatic Multimodal Registration and Fusion of 3D Human Cochlea Images". 14^{th} International Conference on Cochlear Implants, May 11-14, (2016), Toronto, Canada.

List of Other Publications

1. Azzam Sleit,Abdel Latif Abu Dalhoum, Ibraheem Al-Dhamari, and Aiman Awwad: "Efficient enhancement on cellular automata for data mining". (2009) Proceedings of the 13^{th} WSEAS international conference on Systems, S. 616-620.

2. Azzam Sleit, Abdel Latif Abu Dalhoum, Ibraheem Al-Dhamari, and Afaf Tareef: "An edge detection algorithm for online image analysis". (2010) Proceedings of the 2010 American conference on Applied mathematics, S. 250-254.

3. Azzam Sleit, Heba Saadeh, Ibraheem Al-Dhamari, and Afaf Tareef: "An enhanced sub image matching algorithm for binary images". (2010) American Conf. on Applied Mathematics, S. 565-569.

4. Abdel Latif Abu Dalhoum, and Ibraheem Al-Dhamari: "fMRI brain data classification using cellular automata". (2010) Proceedings of the 10^{th} international conference on applied informatics and communications. World Scientific and Engineering Academy and Society (WSEAS,) S. 348-352.

5. Abdel Latif Abu Dalhoum, Basil Mahafzah, Aiman Awwad, Ibraheem Al-Dhamari, Alfonso Ortega, Manuel Alfonseca: "Digital image scrambling using 2D cellular automata". (2012) IEEE Multimedia Journal. 19 (2012), Nr. 4, S. 28-36.

6. Omar Adwan, Ammar Huneiti, Aiman Ayyal Awwad, Ibraheem Al-Damari, Alfonso Ortega, Abdel Latif Abu Dalhoum, Manuel Alfonseca: "Utilizing an enhanced cellular automata model for data mining". (2013) International Review on Computers and Software (IRECOS). 8 (2013), Nr. 2. S. 435-443.

Contents

Prefcase

Improving the quality of life, reducing pain in the world, and saving life are important goals of medicine. As a computer science student who is looking to utilise his knowledge in a noble goal, helping doctors to provide better service is the best way to do it. That is why I decided to work on this field in my master thesis and PhD dissertation. By performing my PhD research in the medical imaging field, I hope someday my work can contribute positively to someone's life.

Another motivation point is that I already worked in this field before during my master study. Doing more research in medical image analysis field helps me to explore it deeply and learn more various problems and their solutions.

This work can be also used in commercial medical systems. As we all know, health is one of the top priorities of human being. One pays a lot to get high-quality health service for himself or the ones he cares. Nowadays, people around the world suffer from cochlear damage which affects their hearing ability. In many cases, a Cochlear Implant (CI) is commonly used as a solution which allows patients to communicate with other people and to enjoy a normal life. Providing an efficient way for cochlea registration and segmentation can be integrated in the current commercial medical imaging software.

Another motivation is the high level of challenging of images registration and segmentation problems. Image registration and segmentation are dominant problems that have many sufficient applications not just in medicine but also in other important fields, e.g. military, Geographical Information Systems (GISs), movie production and video games. Various subject-related papers are published every year which indicate the importance of these problems. Although there are already a great number of publications related to cochlea image registration, segmentation and analysis, no practical methods have been developed yet to solve these problems, so they remain open and active for researchers today. Solving these problems should automate the process of cochlea medical image analysis and provide surgeons and doctors with the needed information for successful decision making. Working in such topic satisfies my scientific curiosity and my interest for challenges.

Finally, obtaining a PhD degree is a big step to my life's dream of being a successful academic professor. There is a shortage in the number of experts in medical imaging academic field in the world, particularly in the Arabic countries. Due to this issue, I would like to spend my lifetime being busy with researches and teaching in many generations. I believe, I can provide the local and international society with well-educated and experienced experts in this field. This goal can be achieved if I succeed in getting a PhD degree under the supervision of an excellent professor in a well-established research environment such as Germany.

Acknowledgment

I am very grateful to everyone who was part of this - my family, my friends, and my teachers.

I could not complete this work without the help and the support of many people who were involved in my project. Especially my friends who stood by my side and supported me with starting my new life here, in Germany.

I would like to express my gratitude to all the people who were a positive part in my life. Starting with my family: my parents from whom I learned to be an independent and free person. My brothers and sisters who shared with me the hard and the good time during my childhood. My teachers who believed in me: Akram AbdulHai from the elementary school, Hussein Khaleel from the high school, Prof. Rauhi Elkhatib from Thamar University, Prof. Arwa Al-eryani from Saba University, Prof. Abdulateef Abu Dalhoum, Prof. Azzam Sleit from the University of Jordan, and especially to Prof Mohammed Al-Rawi from the University of Jordan who first introduced me to this field.

I would like to extend thanks to:

- DAAD for their support during my master study, my short research visit and the first year of my PhD study. Thank to Ms. Karla Barth for being always friendly and helpful.
- Prof. Scheuermann from Leipzig University who invited me for a short research visit where I learned more about fMRI at his laboratory and visited Germany for the first time.
- all my colleagues in AGAS,but especially Detlev Droege who always has the right answers to my questions.
- Ms. Olesia Morozova for helping in the English proofreading.
- Dr. Roland Jacob, Dr. Friedrich Lissek, Dr. Rania Helal, Dr. Horst Hessel, and the Cochlea company for their support during our COMBS project.
- all my excellent students from our internship at the University of Koblenz-Landau.
- the online ITK, VTK, and Slicer community from whom I learnt and I am still learning.
- eng. Hussein Khalil, Dr. Mohammed Adam, Dr. Yasser Jaffal, Dr. Samih Al-Ariki, Dr. Kamal Al-Bauani, Dr. Mohammed Al-Batal, Ms. Kathryna De Leon, Ms. Elfi Leitl, and Veikko Geiss for their help and support.

- everyone who puts effort and time to share his valuable knowledge online.

Finally, there are three persons that my words can not describe my gratitude. First, it is Prof. Dietrich Paulus who welcomed me at his laboratory, provided me with all the support I needed to start over in Koblenz. Second, it is my friend Fawsi Al-Ammari from Yemen, who supported me during the long months when DAAD decided to terminate my scholarship. Last, but not the least, it is Dr. Sabine Bauer who I consider as my German sister and not just my boss. Without them, I could give up even more in my life than just my PhD study.

Chapter 1

Introduction

1.1 Human Cochlea

Taste, sight, touch, smell, and hearing are the traditional five methods of perception. They allow us to communicate with the world around us. The human hearing allows us to communicate, express our feelings, learn, teach and enjoy life, e.g. listening to music.

It is a complicated process that has many steps, see figure 1.1. The sound waves, wave combinations of compression and rarefaction and movements are collected by the ear pinna and directed into the ear auditory canal. A vibration in the eardrum is created by these waves, then it is transmitted into the middle ear ossicles (small hearing bones). These ossicles are connected with the oval window which is a smaller membrane covers one end of the cochlea. After that the vibration causes the fluid inside the cochlea to make specific movements, which activate specific hair cells based on the vibration frequencies. These movements are converted into signals transmitted to the brain through the cochlear nerve (the hearing nerve) to be interpreted later as a sound. High frequencies activate hair cells at the base of the cochlea while low frequencies activate hair-cells at the top of the cochlea (apex or helicotrema), see figure 1.2, and figure 1.3.

An ear has three main parts: the outer ear, the middle ear and the inner ear. The inner ear is formed of interconnected bony chambers filled with fluid. It is responsible for hearing and balance. It consists of the cochlea, vestibule and semicircular canal, see figure 1.2. The cochlea has a crucial role in hearing. It filters and transfers the auditory signals, then sends them to the brain via the cochlear nerve, see figure 1.2. It has a spiral shape and it is divided into three champers called scalae, see figure 1.3. Scala tympani ends with a round window, scala vestibuli ends with an oval window and the scala media, which lies in between the other two scalae, contains the hair cells. The cochlea can be recognized easily because of its spiral shape, see figure 1.2, figure 1.3 and figure 1.4, but the cochlear

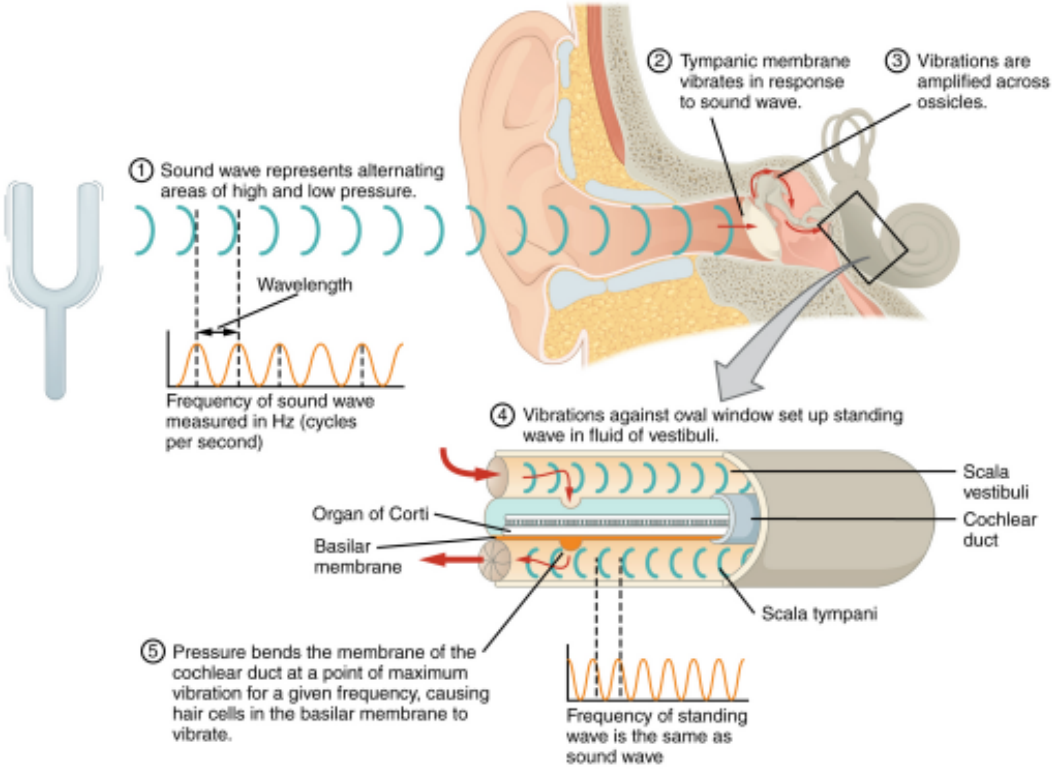


Figure 1.1: Hearing process. Image source is [2].

scalae and its detailed structure are not visible in clinical images, see figure 1.11.

Cochlea Implant (CI) is a relatively new method to restore hearing for patients with severe to profound hearing loss [VV14]. It differs from conventional hearing aids in that it stimulates the cochlear nerve directly from within the inner ear. It is formed of external and internal part, see figure 1.3. The external part constitutes a magnet, a microphone, a processor, and a transmitter, while the internal part constitutes a transducer, a receiver, cables and an array of several electrode contacts. The last one is situated on a silicone carrier implanted inside the scala tympani of the cochlea itself, such that the contacts are located in proximity to sensory nerve cells. With the help of this positioning, the electric signals can di-

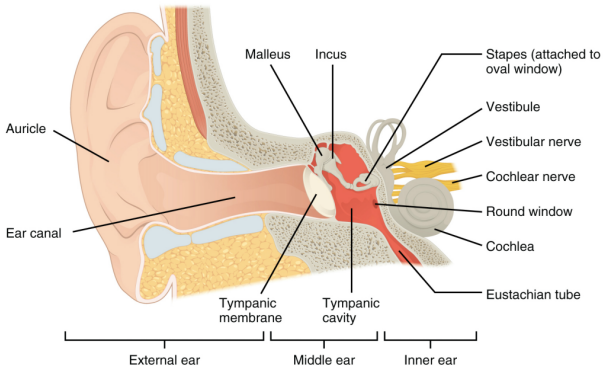


Figure 1.2: Ear anatomy. Image source is [2].

rectly stimulate the cochlear nerve and the neural signals can be transmitted to the brain and interpreted as a sound

These implants result in significant improvement in post-operative speech recognition. This is mostly underpinned by the adequate match between CI electrode frequency bands and their exact location inside the cochlea, as each audible frequency has a specific position inside the cochlea [MJ16] as shown in figure 1.2.

Cochlear Duct Length (CDL) can also have a significant impact on the process of pre-operative electrode selection. If the electrode has a length that is not appropriate to the cochlear length, this will be possible to result in incomplete insertion, cochlear trauma or poor cochlear coverage with poor matching between the electrodes and the cochlea [IELA18]. The surgeon has only one chance to insert CI inside the cochlea which makes the selection of the correct implant critical in successful CI surgery.

Micropump implant [FZA+19] is a promising solution that is more effective than oral drug delivery to treat patients with sensorineural hearing loss and to protect hearing against ototoxic insult, e.g. due to noise exposure or cancer treatments. Designing micropumps to deliver appropriate concentrations of drugs to the necessary cochlear compartments is of paramount importance. However, measuring local drug concentrations over time throughout the cochlea directly is not possible. In recent approaches, quantifying local drug concentrations indirectly using animal models capture a series of magnetic resonance (MR) or μCT images before and after infusion of a contrast agent into the cochlea.

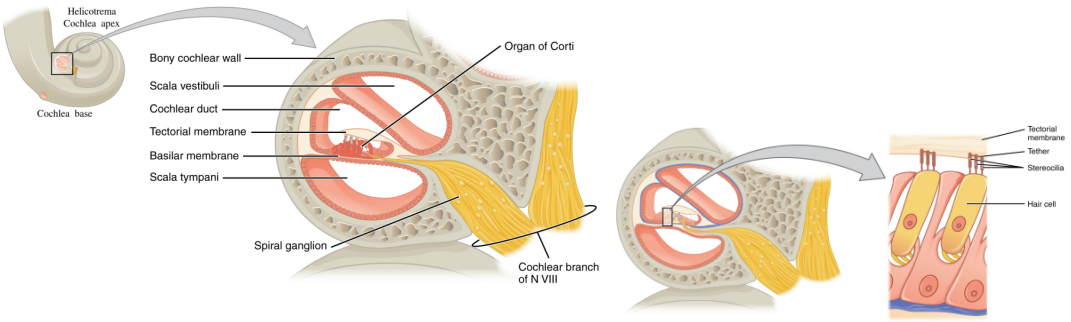


Figure 1.3: Detailed view of the cochlea. Image source is [2].

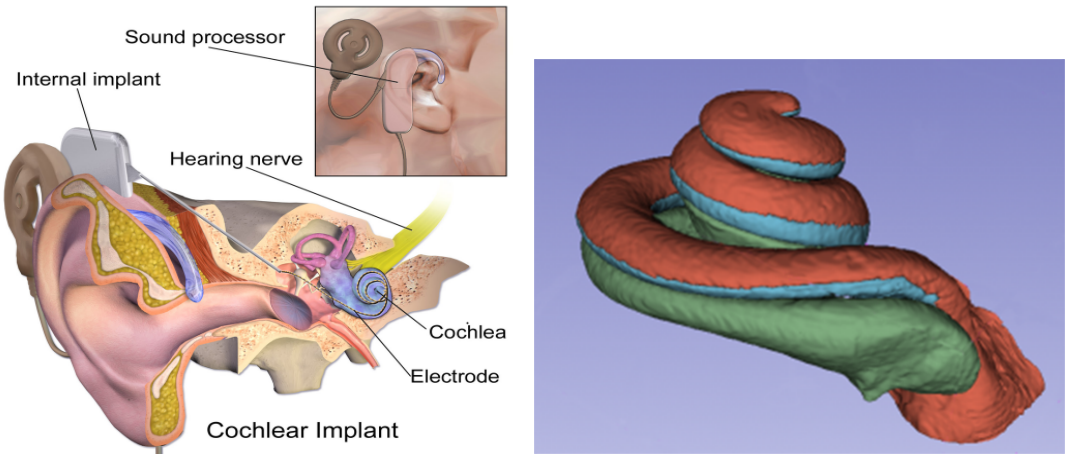


Figure 1.4: Left:Cochlea implant main components. Right: three-dimensional (3D) model of the cochlea generated from a Micro Computed Tomography (μCT) image. Scala tympani (green), scala media (blue and scala vestibuli (red)). Left image source is [14].

Surgical robots are gaining more and more popularity, primarily driven by improvements in nanotechnology and artificial intelligence. These robots need reliable real-time computer vision algorithms in order to detect and analyse the target organ. For instance, during a robotic cochlear surgery [WGW+17], a reliable real-time estimation of the length and the size of the cochlea is needed, e.g. to

decide a suitable CI for a specific patient.

1.2 Medical Imaging

Medical imaging is a field of science that study methods for creating visual representations of the interior body. This includes the physics and the engineering of the medical scanners such as Computed Tomography (CT) [Web88], Cone-beam Computed Tomography (CBCT) [Sha14] or Magnetic Resonance Imaging (MRI) [Web88]. It also includes the software development and the algorithms for processing and visualising these data.

It has many important applications in real life as the medical images are one of the most important tools for diagnostics. Medical images are also used for teaching as they provide clear visualisation of the interior anatomy of the human body.

Medical image data are available from different technologies and scanners. If the images are of the same type, we call them mono-modal e.g. CT and CT. If they are different, we call them multi-modal e.g. CT and MRI or MRI T1 and MRI T2 (more about them later). Each type of medical data has advantages and disadvantages over the other types. For this reason, there is a need for these different modalities. In CT one can view the bones but can not view soft tissue. In the other hand MRI shows the soft tissue, e.g. the brain, but can not show bones. CT is fast but dangerous while MRI is slow but more safe.

Medical image data could be two-dimensional (2D) as in X-ray, 3D as in MRI, or four-dimensional (4D) as in functional MRI (fMRI) [Bux02] [DAD10] or T2-mapping [WPRC$^+$04]. Each medical 3D image is a collection of a number of 2D images (called slices). Each 2D image represents a thin slice of a scanned body. The image pixels[1] are arranged on a two-dimensional grid with a known spacing, origin and orientation, see figure 1.5. This allows the calculation of the actual position of any voxel multiplying the respective spacing value with the respective voxel index i and adding the origin. The origin and the orientation define the location of the patient inside the scanner. The spacing defines the resolution of the image data, the smaller value of the spacing, the high resolution of the image [PB07]. It is very important to process the medical image carefully and wisely otherwise one could end up with an output image that does not represent the correspondence patient and misguide the treatment. For example, using the wrong orientation may cause a robot to cut the left leg instead of the right one.

Medical image data are usually stored using Digital Imaging and Communications in Medicine (DICOM) standard which is an international standard related

[1]Pixel: or a picture element, is a function of a point location in the image that returns the color or the intensity value at that location, e.g. $v = f(x, y)$. Voxel: is a pixel in a 3D image e.g. $v = f(x, y, z)$.

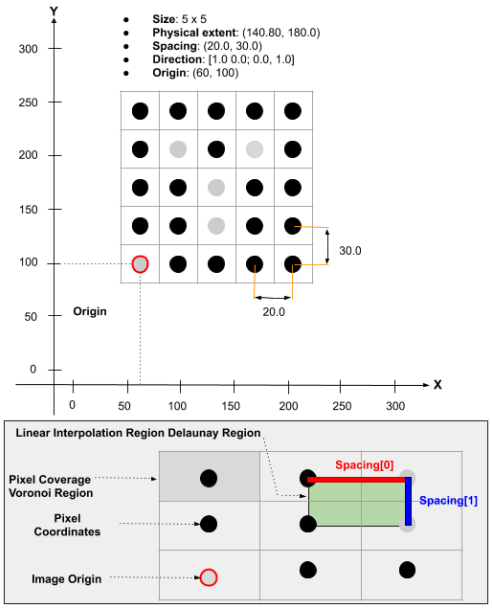


Figure 1.5: 2D pixel spacing and origin example. Image is inspired from [13].

to exchange, storage and communication of digital medical images and other related digital data. The standard covers both the formats and the protocols to be adopted for implementing several communication services. It allows compatibility support for exchanging information among different devices and medical information systems. Today, the vast majority of digital medical imaging systems of all major vendors support and comply with portions of the DICOM standard [PB07].

DICOM stores not just the image information but also all related information to the image, e.g. a scanner, a patient, a doctor and hospital information. DICOM stores this information using DICOM Tags.

Each tag row has three parts: a tag, a description and a value. The tag is hexadecimal numbers in the format (XXXX,XXXX) which may be divided further into DICOM Group Number and DICOM Element Number. The description, Value Representation (VR), describes the data type and the format of the attribute value. For example, the directions' matrix which stores the object orientation can be found at tag (0020,0037) and it may look like 0.977443, 0.017308, -0.210490, 0.017307, 0.986721, 0.161501.

It is very important to use an anonymous processed copy of the DICOM in research to avoid conflicts related to patient's privacy. Since the DICOM may have many irrelevant images as well, a preprocessing step is sometimes necessary.

To process or to view medical image data, one need a specialised medical image tool. These tools are based on cpp but now there are available wrappers to support

other popular programming languages such as Python and Java [2].

The Visualization Toolkit (VTK) [30] is a tool for visualising medical images or computer graphics objects. It is a free open-source Object-Oriented Programming (OOP) tool based on openGL [24]. Most of the computer graphics and rendering algorithms are implemented in a simple way. It is considered now one of the standard tools for medical image visualisation and used in popular software such as 3D Slicer, Medical Image Interaction Toolkit (MITK) [21], and itkSnap [26].

The Insight Toolkit (ITK) [17] is an image processing tool for medical image processing and analysis. It is something similar to openCV, but it adds the support of handling medical images in addition to "normal" images. ITK does not visualise the images that is why to view the images one needs a tool such as VTK. ITK has many images processing, image registration, and image segmentation algorithms that satisfy different needs and solve many problems.

The medical image registration tool Elastix [12] is a special tool for medical image registration based on ITK. It simplifies the use of ITK so one can select the registration method and all parameters by modifying a simple text file instead of writing and compiling the code. The tool has two main executable files, Elastix for image registration, and Transformix for image transformation. Unfortunately, the generated transform is not compatible with ITK and has a different format. Luckily, the deformation field can be used as a transform in 3D Slicer. The developers are active and they have implemented new optimisation[3] algorithms, but they are only available in Elastix[4]. The tool has some problems related to point transformation, and it seems the support for Python version is stopped. Similar to ITK, one needs an external tool, e.g. VTK, to view the transformed images.

3D Slicer [25] is a user-friendly, open-source, free application for medical image processing, visualisation, and analysis. It is based on ITK and VTK. One can extend it easily using simple a Python script. It has many ready-to-use modules for medical image processing, registration and segmentation. Moreover, many developers around the world contribute new extensions and tools regularly for solving different medical imaging field related problems. The application is easy to download and build in a few minutes. One can learn how to use it and extend it in a week. The community is very active and helpful. Questions are answered in a short time with enough details mostly by academic Professors with high expertise. There are many other freeware and commercial tools for medical images today but since I started using 3D Slicer, it is now my favourite tool. Other students who

[2]Some tools they have even simpler version in addition to the Python version, e.g. SimpleITK and SimpleElastix.

[3]Note: UK English is used in this dissertation.

[4]It seems, the developers want to be independent of ITK which I think is a bad idea, it would be nice if they implemented new components in ITK then imported it the same way they did with other ITK components. This would make it easier for maintenance and update.

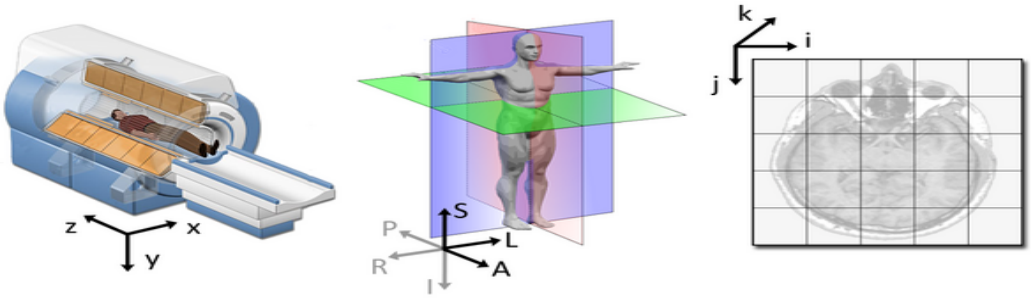


Figure 1.6: World, anatomical, and IJK coordinate systems. The letters in the middle figures are: P: posterior A: anterior, S: superior, I: inferior, L: left, R: right. Image source is [27].

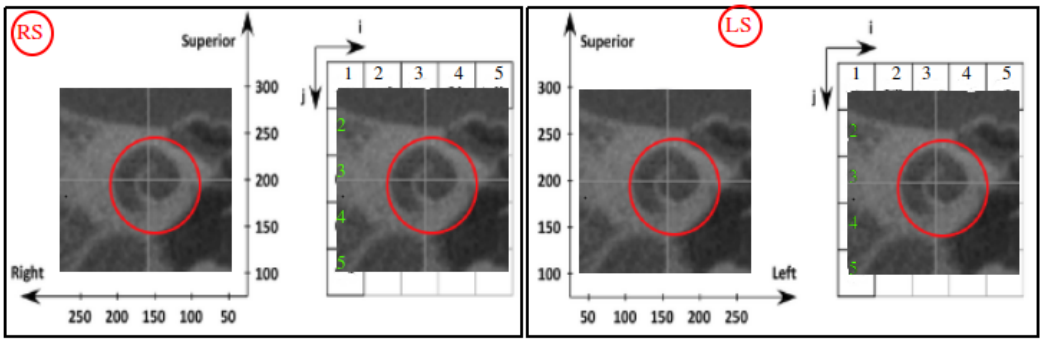


Figure 1.7: Coordinate systems in 2D. Left: RS coordinate system, the origin is at the point (250,300), right LS coordinate systems, the origin is at the point (50,300).

worked with me find it also easy to learn, they could start development in a few days.

Different medical applications use different coordinate systems. World, anatomical and image systems are the most common coordinate systems. The world coordinate system is the one found in the scanner, e.g. x, y, and z in the range of the scanner. The anatomical coordinate system is the one related to the patient, e.g. Right Anterior Superior (RAS). The image coordinate system (IJK) is the one found in the image space, an integer index describes the location of the pixels, see figure 1.5 [13], and figure 1.6 [27].

Sometimes, it is required to convert coordinate systems, e.g. RAS from or to IJK. This conversion uses a 3D affine transformation which includes a shear, a reflection, a rotation, a scaling and a translation. Using homogeneous coordinates, an affine transformation is written with a single matrix.

There are a high number of medical images technologies. I will brief here some of them which are used in this study. One should refer to the more technical book

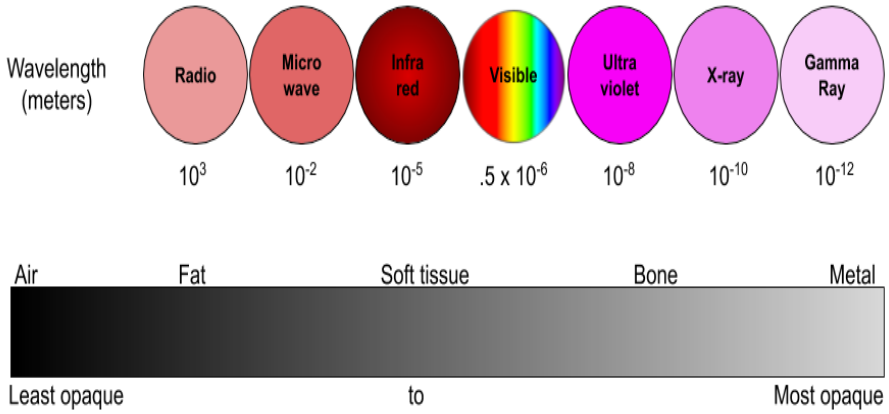


Figure 1.8: X-ray, top: wave length ranges, bottom: body characteristics.

for extra information such as [Web88].

X-ray is a penetrating form of high-energy electromagnetic radiation with a wavelength ranging from 10 picometers to 10 nano-meters figure 1.8 [10]. The German scientist Wilhelm Röntgen discovered X-rays in 1895 by accident while experimenting with electrical gas-filled tubes [32]. He named it X-radiation as it was an unknown type of rays. The photograph of Röntgen wife's hand was the first picture of a human body part using X-rays. When X-ray goes through a body, different types of tissues absorb it differently. X-ray photons have high energy that large calcium atoms are absorbed and result in an image of the bones on a plate of sensors sensitive to x-ray which is used behind the patient. Using contrast media (a liquid that absorbs X-ray photons),the x-ray produces images of soft tissues as well. An X-ray is a form of ionizing radiation that should not be used frequently. When X-ray hits an electron inside the body, it may form an ion. Collection of ions may result in a natural chemical reaction within the body that creates a DNA mutation. This mutation could form cancer or other extreme health issues.

Computed Tomography (CT) [Web88] (also formerly known as computed axial tomography or CAT) is a medical imaging technology that uses combinations of multiple X-ray measurements taken from different angles to produce a 3D image from (cross-sectional) slices of a body. This allows visualisation of the bony structures inside the body. The inventors of this technology, Cormack and Hounsfield, were awarded the 1979 Nobel Prize. An example of a CT scanner is shown in figure 1.9 left [8].

Cone-beam Computed Tomography (CBCT) is a variation of traditional CT systems [Sha14]. The CBCT system rotates around the patient, capturing data

Figure 1.9: Examples of CT, CBCT, and MRI scanners. Image sources are [8], [4], and [22].

using a cone-shaped X-ray beam. Using these data, a 3D image of specific patient's anatomy is reconstructed, e.g. dental (teeth); oral and maxillofacial region (mouth, jaw, and neck); and ears, nose and throat (ENT).

CBCT scanners produce 3D images, which are more informative than 2D images produced by the conventional X-ray technology, see figure 1.9 middle [4] . This may help for better diagnosis, treatment planning and evaluation. The radiation doses from CBCT devices are usually lower than from the other CT devices, but at the same time, they deliver more radiation than conventional dental X-ray devices.

Younger patients are more sensitive to radiation and more vulnerable to radiation exposure. The estimates of lifetime risk for cancer incidence and mortality per unit dose of ionizing radiation are high and the more they live the more probability for ill-effects to develop.

Micro Computed Tomography (μCT) [GZR+16] is a 3D imaging technique, which creates 3D images slice by slice by utilizing X-rays. It is similar to CT but on a small scale with very high resolution. It is usually used for small objects, e.g. under 25 cm. Since the scanner is too close to the object, this provides high-resolution images.

"In vivo"(a Latin word for "within life") means the scanning of live specimens, while "ex vivo" (a Latin for "out of living") refers to the things used to be alive or to samples excised from alive things. Ex vivo μCT instruments allow for higher spatial resolution, longer scan times, better signal-to-noise ratios, and therefore better images.

Most of the cochlea structure can be visualised using μCT with high resolution, e.g. 0.0004 mm. Hence, many studies uses ex-vivo cochlea μCT images.

Magnetic Resonance Imaging (MRI) [Web88] is a medical imaging technique that uses strong magnetic fields, e.g. 1.5 or 3.0 tesla, magnetic field gradients and radio waves to generate 3D images of the soft tissue inside the body, e.g. brian, and

heart. It was originally called nuclear magnetic resonance imaging. The scientists behind MRI, Mansfield and Lauterbur, were awarded the Nobel Prize in 2003. An example of MRI scanner is shown in figure 1.9 left.

MRI does not use X-rays or ionizing radiation, which makes it safer than CT and CBCT. MRI scanner is slower than CT scanner and louder. Because of the MRI nature, a patient with medical implants or other non-removable metal inside the body may be not allowed inside the MRI scanner.

The concept of MRI is described in figure 1.10. It is based on the fact that the human body consists of 60% water. A water molecule has two hydrogen atoms and an oxygen atom. These molecules are moving randomly. When they are affected by the magnetic field generated by the MRI scanner, the Hydrogen atoms response and rotate along. A few of these hydrogen atoms have low energy, so they are slow to gain or release the energy for the rotation.The MRI scanner uses the energy released from these atoms to generate the 3D image. T1 and T2 are two important types of images generated by MRI scanners. T1 image is generated during the T1-relaxation (spin-lattice) time, the duration of time required for molecules to align with the magnetic field. a T2 image is generated during the T2-relaxation (spin-spin) time, the time required for the molecule to restore their original status. Different tissues have different T1- and T2- relaxation times. In T1 images, the blood and the fat are bright, while in T2, they are less bright. Contrast agents, e.g. gadolinium, are used also to get more customised or enhanced MRI images.

There is a lot of physics and mathematics behind the medical imaging devices, one can learn more from more related technical books, e.g. "The Physics of Medical Imaging" by Webb [Web88].

1.3 Problem Definition

The dissertation problem is to find an automatic method for getting more accurate measurements of the human cochlea scalae lengths and sizes, i.e scala vestibuli, scala media and scala tympani, from standard cochlea clinical medical images. To achieve this, cochlea image registration and segmentation is needed.

Image registration and segmentation results are dependent on the image content. There is no general method that solves these problems for all type of images.

The current methods fail to produce satisfying results for the cochlea clinical images because there are lots of challenges. These scalae are not visible in such images, see figure 1.11.

This is due to the nature of the cochlea small size. A radiologist cannot obtain better quality cochlea images due to safety consideration of using CT scanners. Moreover, these clinical images have low resolution, different modalities and high

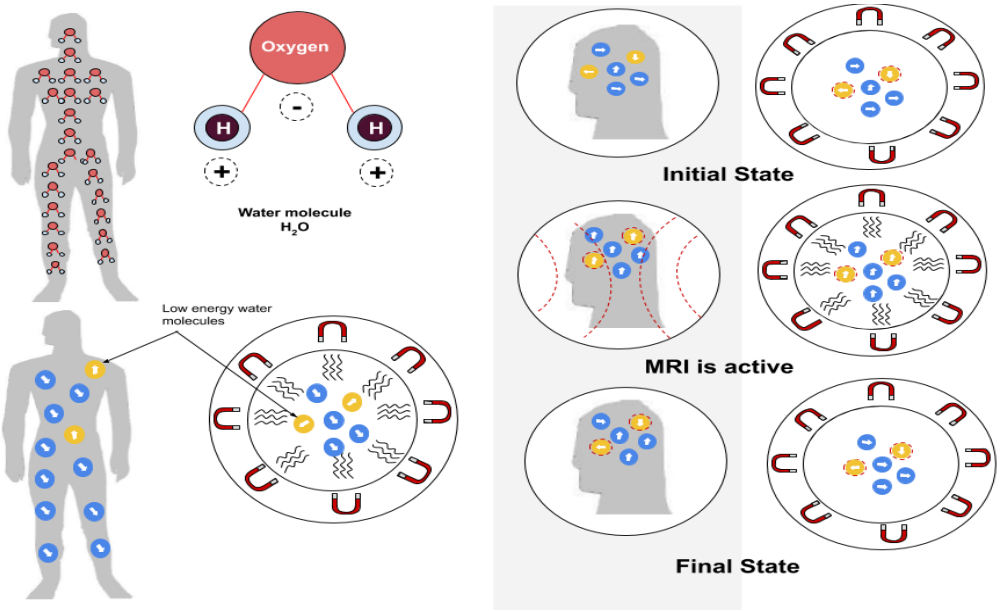


Figure 1.10: How MRI works. Up to 60% of Human body is water which contains of Hydrogen H and Oxygen O atoms. MRI targets water molecule with low energy. In the initial state before applying magnetic field, all H atoms rotate randomly. When applying the magnetic field, all atoms align with the field, some of them gain energy slowly. When the magnetic field stops, all atoms go to the initial state, some of them lose energy slowly. By applying a radio frequency during the gaining and losing energy periods, MRI images with different characteristics are generated. Image is inspired from [11].

noise level. All these challenges must be considered to solve the dissertation problem.

Clinicians and radiologists use cochlea medical images as an important tool that helps them to decide different factors about the cochlear implant surgery. The small size of the cochlea and the complicated bony structure make it difficult to be visualised in clinical images generated by today's technology, see figure 1.11.

Different image modalities provide different information, e.g. CT [Sue09] and CBCT [SFS06, Sha14] visualize the bony structures while MRI visualizes the soft tissues. Combining images from different modalities using image fusion techniques may provide a more informative image to help clinicians in getting more accurate and detailed information about the patient's condition.

The figure 1.12 shows two 3D clinical images from CT and MRI modalities. It shows one slice from each of three views axial, sagittal and coronal of the left side cochlea (the side is based on the direction of the patient's head, e.g. a top-to-bottom or bottom-to-top approach). The image voxel spacing sizes are 0.4 mm $\times$ 0.4 mm $\times$ 1.0 mm for CT and 0.4 mm $\times$ 0.4 mm $\times$ 0.7 mm for MR. With such a

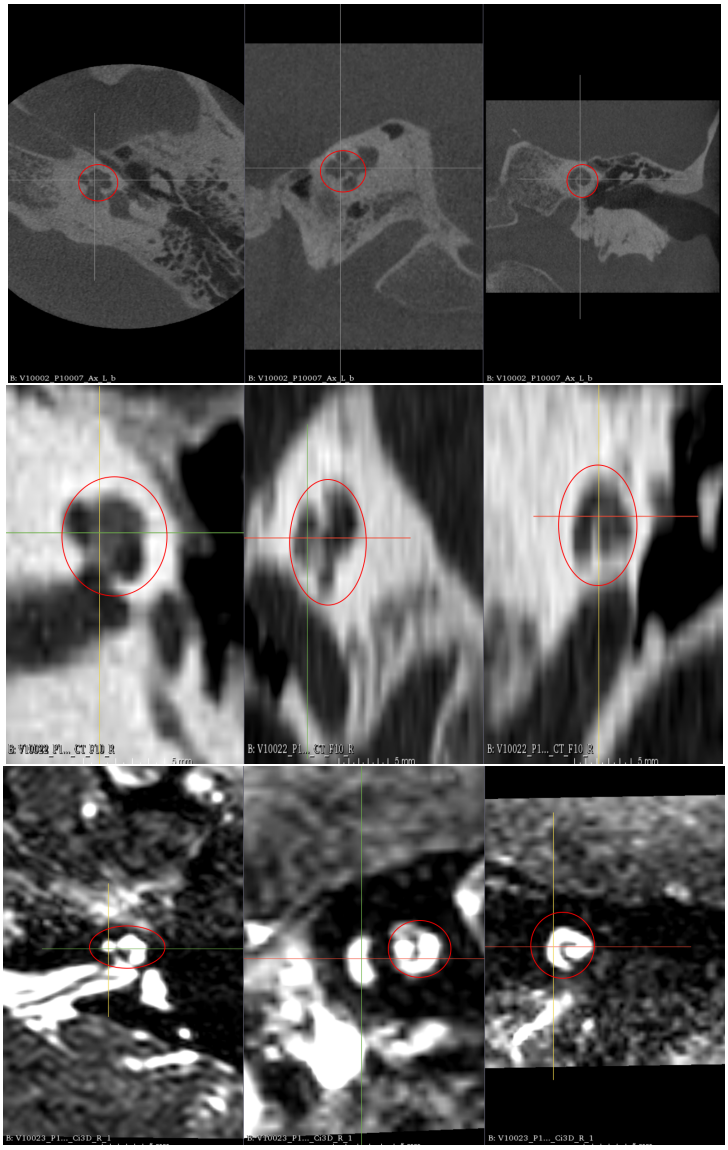


Figure 1.11: Cochlea clinical images. Top CBCT, middle: CT, and bottom: MRI. The red circles mark the cochlea area. Each image show the three 3D standard views i.e. axial(left), sagittal(middle) and coronal (right).

low resolution, one can not see the cochlea structure. Moreover, automatic analysis methods do not produce good results when they are used on such low-resolution images.

For an efficient fusion, images must be aligned correctly to the same physical space using image registration. Radiologists usually use a manual procedure that takes a long time and many efforts to align a pair of images. Automat-

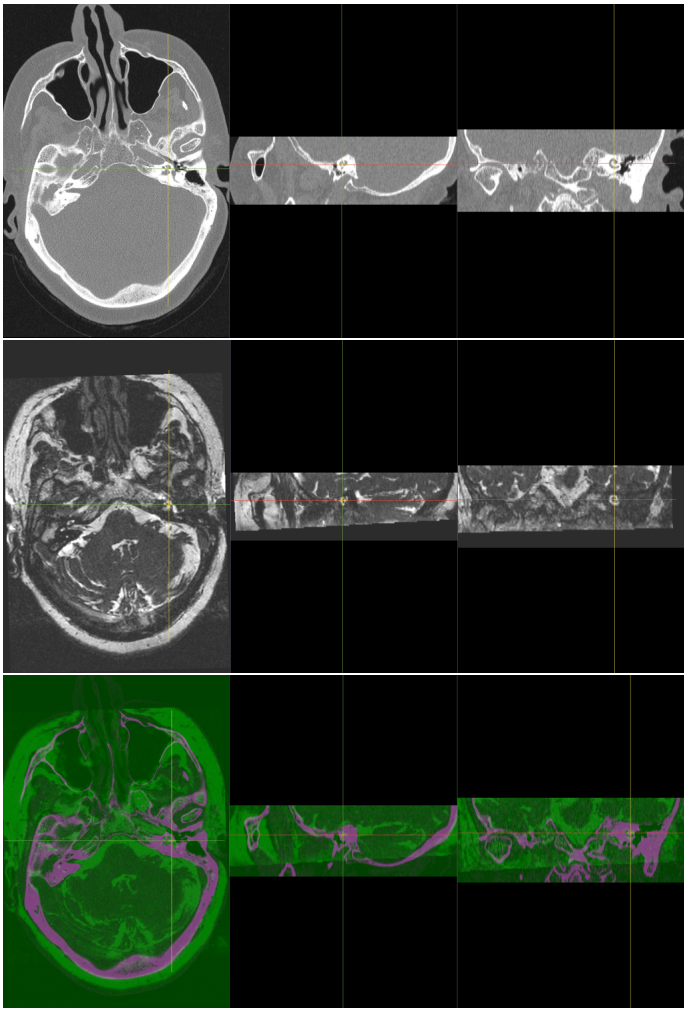


Figure 1.12: Image registration. Top to bottom: two input 3D cochlea images of the same patient, CT as a fixed image and MRI as a moving one. In the registered and fused 3D image, magenta colour represents the CT part and green colour represents the MR part. Each 3D image has 3 views, from left to right: axial, sagittal and coronal. The crossed lines represent the location of the left side cochlea.

ing this process using an image registration method saves both time and efforts. Another application for this automation is surgical robotics. Surgical robots are gaining popularity, primarily driven by current improvements in nanotechnology and artificial intelligence. These robots need reliable real-time computer-vision algorithms in order to detect and analyze the target organ. For instance, during a robotic cochlear surgery [WGW+17], a reliable real-time cochlea image registration is needed, e.g. to be used in cochlea image analysis.

1.4 Dissertation Hypotheses

The dissertation problem can be solved using image registration and segmentation techniques.

1. The cochlea multi-modal image registration problem can be solved using the Adaptive Stochastic Gradient Descent (ASGD) [KPSV09] optimizer to find parameters of 3D rigid transform that minimizes the negative value of Mattes's Mutual Information (MMI) metric [MHV+01]. The reasons for selecting these methods are, at first: they are state of the art that are published in the top scientific journals, and second: they successfully handled similar problems for large objects, e.g. brain, prostate and lunge scan.

   The novel contributed idea is to work on a small area in the clinical image that has less noise and clear shapes, then apply the transform on the original image. This solves the problem and required less computation time.

2. The cochlea multi-modal image segmentation problem can be solved using a combination of model-based and atlas-based segmentation approaches. Using a high-quality μCT cochlea image as a model helps to identify different scalae that are not visible in the clinical images. Using this model in an atlas-based segmentation approach [RBM+05] solves the segmentation problem practically, i.e. with more accuracy and less computation time. The novel contributed idea is to register a clinical image that serves as a model to the input image instead of registering the segmentation directly to the input image as in the classical atlas-segmentation. The multi-stages registration uses the rigid transform in the first stage and the B-spline transform in the second stage. The B-spline transform handles the different deformations between different patients.

Own contribution:
The following list shows my own contribution:

- Human Cochlea Dataset (HCD), a new public and standard cochlea dataset with landmarks for evaluation, see section 5.1. The dataset contains images of 66 patients, 41 from a German hospital and 25 from an Egyptian hospital. The patients are of different gender and age. The total number of images is 150 clinical images from three modalities 87 CBCT, 40 MR, and 23 CT.

- ACIR2016, ACIR2017 and ACIR2020 novel and practical methods for automatic multi-modal cochlea 3D image registration and fusion, see section 5.2. The main idea was cropping the image to a small clear part, e.g. the cochlea part. This idea was not used before in the literature. This removes extra

noise and make the registration problem simpler. Similar ideas exist but are not the same, e.g. masking or using Region Of Interest (ROI). My idea is different from masking as masking still using the full size image and the unmasked area could contribute to the similarity metric negatively. My idea is also different from ROI as it can be applied to any clear part of the image not just ROI. This works because the cochlea image registration problem in my case is a rigid registration problem so the same transformation can be applied on any part of the image.

- ACA2018, and ACA2020 novel and practical methods for automatic multi-modal cochlea 3D image segmentation and analysis using atlas-model-based, see section 5.3. The main idea is to register a pre-defined image to an input image then use the transformation to transform a segmentation or a points-model. This is not found in the literature, the atlas-based segmentation registers a segmentation directly to the input image. While the atlas-segmentation is faster, some times it just does not work. Registering images makes the problem is simpler and in my case, it works all the time. It also works for other cases and could solve many challenging problems e.g. feature or points detection, detailed multi-class segmentation, and getting different measurements from the image.

- Using ACA2018,ACA2020 for solving spine segmentation problem, see section 7.1. An example of using the atlas-model-based method above method to solve the spine segmentation.

- Using ACA2018,ACA2020 for solving spine points detection problem, section 7.2. An example of using the atlas-model-based method to solve points detection.

- Open-source user-friendly tools for cochlea and spine image registration, fusion, segmentation, and analysis.

1.5 Dissertation structure

The following list explains the dissertation structure:

- Chapter 1: Introduction. Problem definition and dissertation hypothesis.
- Chapter 2: Medical image registration and fusion. This covers the image registration theory, important components and terminologies.
- Chapter 3: Medical image segmentation and analysis. This covers image segmentation theory and atlas-based segmentation.
- Chapter 4: Related work. This covers recent publications that try to solve the dissertation problems and discussion on their results.
- Chapter 5: Materials and methods. The proposed methods and the implementation in addition to the dataset and the hardware.
- Chapter 6: The experimental design, results and discussion. This chapter describes the experimental design and the results.
- Chapter 7: Generalisation. This chapter shows that my proposed method can solve other related problems as well, e.g. spine segmentation, finding origin and insertion of ligament points.
- Chapter 8: Conclusion and future work. The dissertation conclusion and ideas for future work.

Chapter 2

Medical Image Fusion and Registration

Image Fusion is the process of combining two or more images into one image. It is used in medical field usually to combine images from different modalities, e.g. CT and MRI. The CT images shows bone structure while MRI image shows soft tissue. Combining them shows both bony structure and soft tissue at the same time. Cochlea fused images from different modalities can be obtained using image registration and fusion techniques. The fused cochlea image may provide a more informative image that helps surgeons making decisions about CI surgery. This process required the input images to be aligned, hence, solving the image registration problem is necessary.

The image registration problem [HH01, Yoo12, Gos04] is the problem of finding a transformation $T(P, \mu)$ that aligns one or more images, called the moving image $I_M(P)$, to a reference image, called the fixed image $I_F(P)$. Here P is the vector of all points coordinates belonged to the image and μ is a vector which contains the transformation's parameters.

This transformation transforms the points of an image to the same location of the points in the reference image. In medical image registration, the intra-subject medical image registration aligns images of the same patient. In the other hand, the inter-subject registration aligns images of different patients. The inter-subject registration is more challenging as different patients may differ in size and shape which requires finding parameters of an expensive non-rigid transformation, e.g. B-spline.

Mathematically, an image is a function of points where the output is the intensity (or colour) values of the point location. The transform is a multivariable function of two inputs The first input is the vector points P of the image. The second input is the vector μ which contains the transformation parameters. T

takes an image point (or a vector of points)[1] and outputs the image transformed point (or a vector of transformed points). We can write this transform as $T(P, \mu)$ where P is the image coordinates vector.

An example for a such vector is a vector of all points of a 2D image:

$$P = [(x_0, y_0), (x_1, y_1), ..., (x_{n-1}, y_{n-1})],$$

where n is the number of pixels in the image. An example is a vector of 2D translation transform parameters where μ includes a shift in x coordinate and a shift in y coordinate.

The 2D translation transform is a simple example of a transformation where $\mu = [t_x, t_y]$ is the vector contains the shifts in x and y coordinates. This transform is defined as:

$$T(p, \mu) = T((x, y), (t_x, t_y)) = (x + t_x, y + t_y),$$

Finding a transform parameter is a challenging task especially for complicated transforms with many parameters. The methods, which work for some images, may not work on a different set of images. Hence, the general registration problem is still unsolved and many papers are published every year trying to solve different specific image registration problems.

Optimisation techniques [NW06] are usually used to find these parameters. An optimiser, such as the Gradient Descent (GD)[Sny05], tries to find these parameters by minimising a similarity metric cost function of the input images.

Popular image registration cost functions use similarity metrics such as Mean Squared Error $S_{MSE}(I_F, I_M)$ [DCRB19] or Mutual Information $S_{MI}(I_F, I_M)$ [VW97]. The goal is to increase the similarity between the transformed moving image and the fixed image by minimising the similarity metric function. For some similarity metrics we may minimise the negative value of the similarity metric to maximise the similarity between the images, e.g. as in the case of the mutual information metric S_{MI}. figure 2.1 shows a diagram of the main components of the image registration pipeline.

The image registration optimisation problem is stated as follows:

$$\tilde{\mu} = \underset{\mu}{\text{argmin}}, S(I_F(P), I_M(T(P, \mu))) \quad , \tag{2.1}$$

where $\tilde{\mu}$ is the optimal transform parameters, μ is the transformation parameters' vector, I_F and I_M are the fixed and the moving images, P is the points

[1]Note: I am using vector to describe a list of elements as this is more related to implementation e.g. vector in c++, not a vector as a mathematical content.

coordinate vector, S is the similarity metric function, and $T(P, \mu)$ is the transformation function[2].

In the GD optimiser, the new parameters are updated using the derivative of the cost function with respect to the transformation parameters, e.g.:

$$\mu_{k+1} = \mu_k - \lambda \frac{\partial S(I_F(P), I_M(T(P, \mu_k)))}{\partial \mu} \quad , \tag{2.2}$$

where k is the optimisation iteration, λ represents the step-size of the GD.[3] The derivative of the similarity metric with respect to the transformation parameters is required for the equation (2.2). Using the chain rule, this derivative can be divided into two main parts:

$$\frac{\partial S}{\partial \mu} = \sum_{j=0}^{n-1} \left(\frac{\partial S(I_F(p_j), I_M(T(p_j, \mu_k)))}{\partial T(p_j)} \quad \frac{\partial T(p_j, \mu_k)}{\partial \mu} \right) \quad , \tag{2.3}$$

where $\frac{\partial S}{\partial \mu}$ is the first order derivative of the similarity metric S with respect to the transformation parameters μ at k optimisation iteration, p_j is an image pixel coordinate, and n is the total number of the image pixels. The first part of the right-hand side in the equation (2.3) requires the derivative of the similarity metric and the transformed moving image. The second part $\frac{\partial T(p_j, \mu_k)}{\partial \mu}$ is called J_μ and it is similar to the Jacobian matrix of the transformation as it represents the transformation partial derivatives at each image point with respect to the transformation parameters. The term J_μ is computed at each optimisation iteration. For a large image size and complex transformation, the computation requires a lot of time to complete.

2.1 Interpolation

In general, an image registration pipeline should have an interpolator. Almost every geometric transformation requires interpolation. The quality of the interpolation has a large influence on the result of the registration.

The interpolator is a method that predicts a function value $f(x)$ at a specific input value x. This prediction uses the known function values to predict the unknown function value. Since the transformation may map a point to an area outside the image space, interpolation is used to predict the intensity value of the transformed point when it is outside the image space based on the image values.

[2] $I_M(T(P, \mu))$ can be read as the transformed moving image, in some references it is written as $T_\mu(I_M)$.

[3] It is also called the learning rate in some references e.g. as in elastix and tensor flow optimisers, see `https://elastix.lumc.nl/doxygen/classitk_1_1GradientDescentOptimizer2.html` and `https://www.tensorflow.org/api_docs/python/tf/keras/optimizers/SGD`

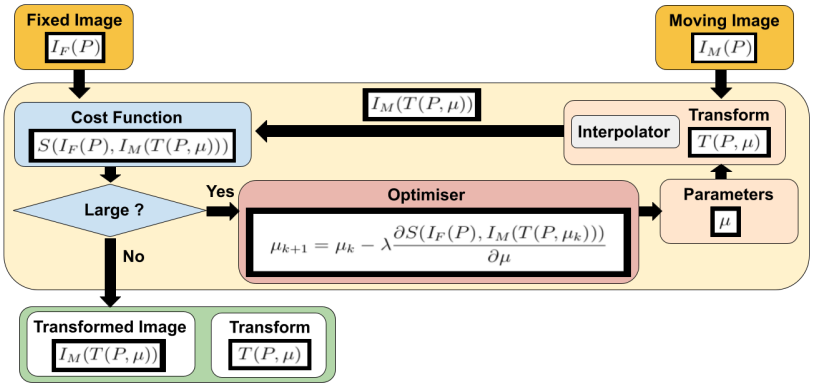


Figure 2.1: Main image registration components.

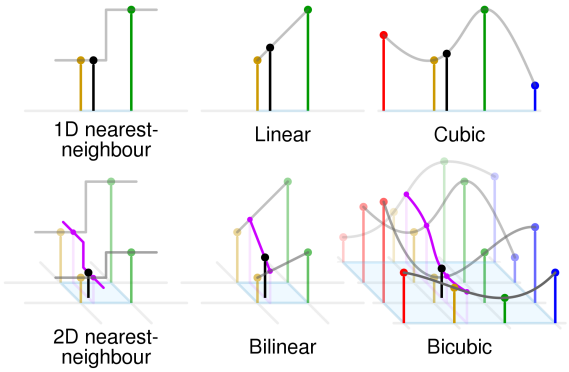


Figure 2.2: Comparison of some popular interpolation methods in 1D (top) and 2D (bottom). Image source is [20].

In this dissertation, some common interpolation methods used in my research will be covered briefly, e.g. K-Nearest Neighbour, Linear, Polynomial and B-splines interpolators.

The general interpolation form is $f(x) = \sum_{k \in Z^q} c_k . \phi(x - k)$, where x is the point that we need to interpolate, Z is the set of integer numbers, q is the dimension, e.g. for 2D q=2, c_k is a coefficient, e.g. some function of the sample value.

For implementation purposes we use this similar equation:

$$f(x) = \sum_{i \in |k|} c_i . \phi(x - k_i), \tag{2.4}$$

where k is a set of integer values from the domain of x and $|k|$ is the length of this set. The function $\phi(x)$ defines the interpolator as we will see in the next paragraphs.

K-Nearest Neighbour (KNN) interpolation is probably the simplest interpolator that selects a value from neighbour values. This gives no need to introduce new values. The function $\phi(x)$ in equation (2.4) is defined as follows:

$$\phi(x) = \begin{cases} 0, & x < \frac{-1}{2} \\ 1, & \frac{-1}{2} \leq x < \frac{1}{2} \\ 0, & \frac{1}{2} \leq x \end{cases} \tag{2.5}$$

This interpolator is important in case of segmentation or binary images to avoid introducing new intensity values. Note that the above definition is a simplified version of the original method. The original method involves sorting the distances from each point to all other points then select the k-nearest values to the point to be interpolated.

Linear interpolation is one of the simplest interpolators. The predicted value is a point in a line between the values of the function before and after the point to be interpolated. The $\phi(x)$ for linear interpolation in equation (2.4):

$$\phi(x) = \begin{cases} 1 - |x|, & |x| < 1 \\ 0, & 1 \leq |x| \end{cases} \tag{2.6}$$

where $|x|$ is the absolute value of x.

Polynomial interpolation tries to fit a polynomial function of degree n to the points. Usually cubic polynomial function is used, i.e. $n = 3$, see figure 2.2 and section 2.2 .

$$f(x_i) = \sum_{j=0}^{j=n} a_j x_i^j = a_0 + a_1 x_i + a_2 x_i^2 + ... + a_n x_i^n \quad . \tag{2.7}$$

Finding the values of a_j that satisfies $f(x_i) = y_i$ requires solving a linear equations system of the form $Xa = Y$, e.g. by using Gaussian elimination method.

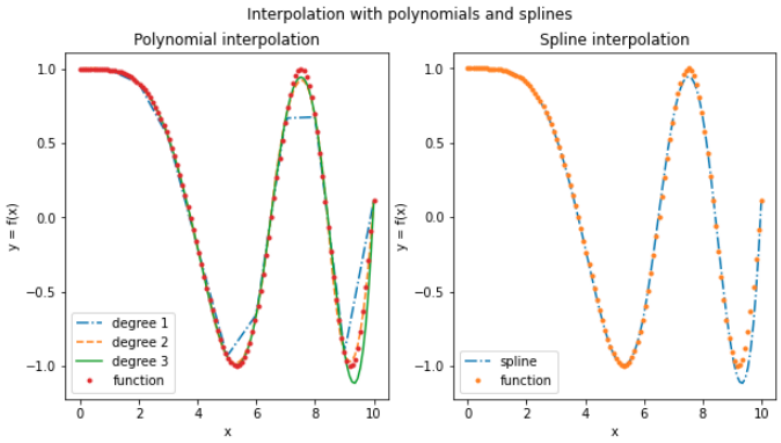


Figure 2.3: Comparison between polynomial and splines interpolation of a function $f(x) = 1/(1 + 25x^2)$.

B-splines interpolation tries to find a function that represents the point using the concept of the basis function. The word spline means a long, thin and flexible strip of wood or metal. It was used in the old days to draw smooth curves passing through fixed points. The points were marked with heavy nails and the spline was made to go around each one of them.

Splines describe a smooth curve that passed through all points. The Romanian-American mathematician I. J. Schoenberg used this idea to generate smooth functions passing through given points and decided to call them spline functions [SS73]. Splines are polynomials between two consecutive 'nails' but the polynomial between two consecutive nails is different from the one between the prior and the succeeding nails. That is why splines are said to be piece-wise polynomial.

In figure 2.3, the outcome of trying to fit polynomials of varying orders compared with spline interpolation is shown. I tried fitting polynomials of degrees 3, 5, and 7 to the points marked as red dots. The higher order polynomials fit the points better, but they also oscillate in a way the original function does not. This behaviour is called Runge's phenomenon, in honour of the German mathematician Carl David Tolmé Runge who first reported it [Run01].

B-spline can be used as a transformation, hence more about the B-spline theory will be explained in the next section.

2.2 Transformation

In image registration, the transformation is a spatial transform type not an intensity transform. This means we are not changing the pixel values rather we are moving them around. In the medical images processing field, one must be care-

ful using these transformations as it could easily generate misleading transformed images.

A transformation is a multi-variable function that takes one point p (or number of points P) and parameters μ as an input, and outputs a transformed point $\tilde{p}$ (or transformed points $\tilde{P}$):

$$T(P, \mu) = \tilde{P} \quad . \tag{2.8}$$

The derivative of the transformation with respect to its parameters is used in the image registration so it is important to compute this derivative.

$$J_{\mu} = \frac{\partial T(P, \mu)}{\partial \mu} \quad . \tag{2.9}$$

I will provide some simple examples of common transformations and their derivatives later. For now, some important terms related to transformation will be explained. The Identity transform is the transformation that does not change the input. It is sometimes used to simplify matrix operations. The rigid transform does not change the shape of the input object, e.g. identity transform, translation transform and Euler transform. These transforms may shift or rotate the object without applying scaling or deformation. On the other hand, the non-rigid transform includes rigid transform parameters and add more deformation parameters, e.g. scaling or shearing, see table 2.1 for some common transformations.

Table 2.1: Common transformations and their properties.

Transform	**Type**	**Number of parameters**		**properties**
		2D	**3D**	
Identity	rigid	0	0	output point =input point
Translation	rigid	2	3	shifting
Mirror	rigid	2	3	flipping
Euler	rigid	3	6	shifting, flipping, and rotation
Similarity	non-rigid	5	9	shifting, flipping,rotation, and scaling
Affine	non-rigid	7	12	shifting, flipping,rotation, scaling, and shear
B-splines	non-rigid	(H x W) grid	(H x W x D) grid	deformation of specific parts of the image

The letters H, W and D represent respectively the height, width and depth of the image. Note that the number of B-splines transform is variable. These parameters are the B-spline control points (also called B-spline coefficients) and usually they are represented as a 2D or 3D grid. This allows to transform one part or multiple parts of the image without affecting other parts by modifying only specific control-points in the grid. The size of the control-points grid is usually based on the images size so the number of parameters could reach thousands for some 3D medical images. This transform is also very useful to filter an image or

compute the image gradient in a more efficient way. More information about this transform with some simple examples will be added later.

In this chapter, I will explain only some basic transformations.

Translation transform shifts a point from one location to the other new location.

$$T(p,\mu) = T(x,t_x) = x + t_x \quad , \tag{2.10}$$

where p is one point belonging to the image. For n dimension:

$$T(p,\mu) = T((x_0,..,x_{n-1}),(t_{x_0},..,t_{x_{n-1}})) = (x_0 + t_{x_0},...,x_{n-1} + t_{x_{n-1}}) \quad , \tag{2.11}$$

where p is a multi-dimension one point belonging to the image. An example for a 2D point translation transform and its derivative is the following:

$$T(p,\mu) = T((x,y),(t_x,t_y)) = (x + t_x, y + t_y) \quad . \tag{2.12}$$

$$\frac{\partial T(p,\mu)}{\partial \mu} = \frac{(x + t_x, y + t_y)}{\partial (t_x, t_y)} = \left(\frac{(x + t_x, y + t_y)}{\partial (t_x)}, \frac{(x + t_x, y + t_y)}{\partial (t_y)} \right) = ((1,0),(0,1)) \quad . \tag{2.13}$$

Notice that the derivative is the square identity matrix of length 2 (the point dimension). In general, for the 2D translation transform we have:

$$J_\mu = \frac{\partial T(p,\mu)}{\partial \mu} = \left(\left(\frac{\partial T(\mu_0,x)}{\partial \mu_0}, \frac{\partial T(\mu_0,y)}{\partial \mu_0} \right), \left(\frac{\partial T(\mu_1,x)}{\partial \mu_1}, \frac{\partial T(\mu_1,y)}{\partial \mu_1} \right) \right) \quad , \tag{2.14}$$

Mirror transform requires only the flipping axis. Hence, it needs only 2 parameters in case of a 2D image and 3 for a 3D image. It is a special case of the rotation transform where the angle of the rotation is known (180 degrees) and we only need to know if the direction is affected or not. Note that we can also use one parameter, e.g. integer value that has a meaning as follows: 0 for a flip around x direction, 1 for a flip around y direction, 2 for a flip around both x and y directions, etc. After that, we define this mapping in the transform function.

Euler transform is a rigid transform that combines translation and rotation together. This composite transformation, or sequence of transformations, follows the matrix multiplication order. The order of the transformation is important, e.g. rotation followed by translation is not the same as translation followed by rotation. For 2D images, the transformation needs 3 parameters: the transformation angle θ

in degrees, the shift in x-coordinate and the shift in y-coordinate, e.g. $\mu = \theta, t_x, t_y$. The rotation is counterclockwise about the origin with respect to the x-axis. To rotate a point P around a specific point $P_c = (x_c, y_c)$, one first translates to -P_c then rotates then translates back to P_c.

The transformation matrix for 2D Euler transform:

$$T(p, \mu) = T((x, y), (\theta, t_x, t_y)) = \tag{2.15}$$
$$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & t_x \\ \sin(\theta) & \cos(\theta) & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

From the above equation, the transformed values $\tilde{x}$ and $\tilde{y}$ can be computed using:

$$\tilde{x} = x \ \cos(\theta) - y \ \sin(\theta) + t_x,$$
$$\tilde{y} = x \ \sin(\theta) + y \ \cos(\theta) + t_y \quad .$$

The above matrix applies a rotation followed by translation. For a translation followed by rotation we can write:

$$T(p, \mu) = T((x, y), (\theta, t_x, t_y)) = \tag{2.16}$$
$$\begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & (t_x cos(\theta) - t_y sin(\theta)) \\ \sin(\theta) & \cos(\theta) & (t_x sin(\theta) + t_y cos(\theta)) \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Here is an example: assuming a 2D point $p = (x, y) = (2, 3)$ and a 2D Euler transformation parameters $\mu = (\theta, t_x, t_y) = (30, 10, 20)$. Rotation followed by translation produces the output point $\tilde{p} = (10.232, 23.60)$ while translation followed by rotation produces the output point $\tilde{p} = (-1.11, 25.9)$. To get a translation followed by rotation:

$$\tilde{x} = (x + t_x) \ \cos(\theta) - (y + t_y) \ \sin(\theta),$$

$$\tilde{y} = (x + t_x) \ \sin(\theta) + (y + t_y) \ \cos(\theta),$$

The derivative of the 2D Euler transform with respect to it parameters J_μ is:

$$\begin{aligned} \frac{\partial T(P, \mu)}{\partial \mu} &= \frac{\partial T((x, y), (\theta, t_x, t_y))}{\partial \theta}, \frac{\partial T((x, y), (\theta, t_x, t_y))}{\partial t_x}, \frac{\partial T((x, y), (\theta, t_x, t_y))}{\partial t_y} \\ &= \left(\begin{bmatrix} -x \ \sin(\theta) - y \ \cos(\theta) \\ x \ \cos(\theta) - y \ \sin(\theta) \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right) \end{aligned} \tag{2.17}$$

Note that for 3D rotation we have 3 rotations matrices R_x, R_y, and R_Z with three angles α, β, and γ. These three matrices are combined based on the order of the rotations.

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{bmatrix} \tag{2.18}$$

$$R_y(\beta) = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \tag{2.19}$$

,

$$R_z(\gamma) = \begin{bmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.20}$$

B-spline transform[4] can be considered as a generalisation of the Bézier curves. The B-spline is a curve constructed from a number of smoothly connected Bézier curves. A Bézier curve [23] is a curve represented by a function which allows computing any curve point in the infinite real domain.

The Bézier curve is generated using something similar to the linear interpolation as shown in section 2.2. The figure shows a quadratic Bézier curve with 3 control points. To draw the curve points, first lines between the control points are generated, then a ratio point t is moved from 0.0 to 1.0 on both lines. The lines connecting these t points are generated. The curve points are now the t points on the generated lines.

The n degree Bézier curve is defined as:

$$C(t) = \sum_{i=0}^{n} B_{i,n}(t) p_i \quad t \in [0,1], \tag{2.21}$$

where B is a Bernstein polynomial [Ber12] of degree n and defined as:

$$B_{i,n}(t) = \binom{n}{i} (1-t)^{n-i} \quad t^i \quad 0 \leq i \leq n, \tag{2.22}$$

and $\binom{n}{i}$ is a binomial coefficient which is defined as:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \quad . \tag{2.23}$$

[4]Interactive online demo can be found at `http://jsxgraph.uni-bayreuth.de/wiki/index.php/B-splines`

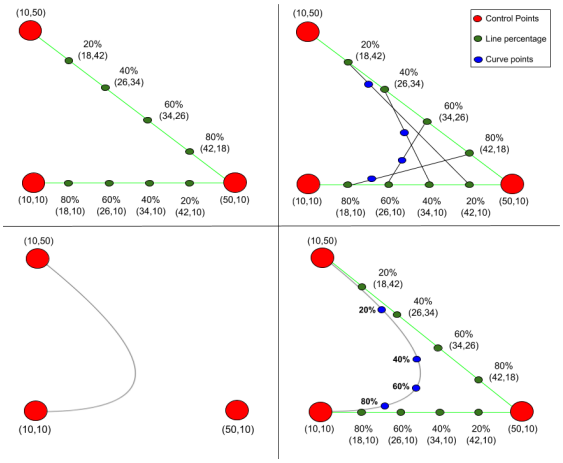


Figure 2.4: An example of a quadratic Bézier curve with 3 control points. Note: Image is generated using `www.ibiblio.org/e-notes` then modified in our lab.

Remember that as higher degree of a polynomial, as there more changes in the direction. Bernstein polynomial can be defined recursively , e.g.:

$$\begin{aligned} B_{i,n}(t) &= (1-t)B_{i,n-1}(t) + tB_{i-1,n-1}(t), \qquad n > 0 \\ B_{0,0}(t) &= 1, \\ B_{i,n-1}(t) &= 0, \qquad i = n, \\ B_{i-1,n-1}(t) &= 0, \qquad i = 0. \end{aligned} \tag{2.24}$$

The number of the Bernstein polynomials for a curve of degree n is $n+1$, e.g. for $n = 1$ there are two polynomials. Here are some examples:

- Linear, degree $n = 1$:

$$\begin{aligned} B_{0,1}(t) &= (1-t) \\ B_{1,1}(t) &= t \end{aligned} \tag{2.25}$$

- Cubic, degree $n = 3$

$$\begin{aligned} B_{0,2}(t) &= (1-t)^3, \\ B_{1,2}(t) &= 3(1-t)^2 t, \\ B_{2,2}(t) &= 3(1-t)t^2, \\ B_{3,2}(t) &= t^3 \end{aligned} \tag{2.26}$$

The Bernstein polynomials can be written as a matrix. This makes the implementation simpler. The cubic Bézier curve equation can be written using the equation above as:

$$C(t) = (-t^3 + 3t^2 - 3t + 1)p_0 + (3t^3 - 6t^2 + 3t)p_1 + (-3t^3 + 3t^2)p_2 + (t^3)p_3 \quad (2.27)$$

The matrix form:

$$C(t) = [p_0, p_1, p_2, p_3] \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} t^3 \\ t^2 \\ t \\ 1 \end{bmatrix} \quad (2.28)$$

The general form:

$$C = P \cdot B \cdot T \quad (2.29)$$

where C is the Bézier curve points vector, C is the control points vector, B is the Bernstein matrix, T is the vector containing t values, i.e. $[t^n, t^{n-1}, ..., t, 1]$.

The derivative of the Bernstein polynomial is defined recursively using equation (2.22):

$$\frac{\partial B_{i,n}(t)}{\partial t} = nB_{i-1,n-1}(t) - B_{i,n-1}(t). \quad (2.30)$$

The convolution of the Bernstein functions of degree n produces a smooth curve, see figure 2.5.

The result of two functions convolution f and g is a modified version of one of the original functions. The convolution is defined as:

$$(f * g)(t) \triangleq \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau \quad (2.31)$$

the symbol $\triangleq$ means "equal by definition", $*$ is the convolution operator. For implementation, the discrete form is required:

$$(f * g)[t] = \sum_{\tau=t}^{m} f[\tau].g[t - \tau] \quad (2.32)$$

where m is the range in which two signals are available at the same time[5]. The length of f signal is usually smaller than the g signal length. The Greek letter τ is an index for the location at f. Usually f is a large signal or an image while g as a small signal or a filter. When g is an even function , e.g. $g(t) = g(-t)$ convolution is equivalent to correlation.

[5]Note: [t] is a signal symbol

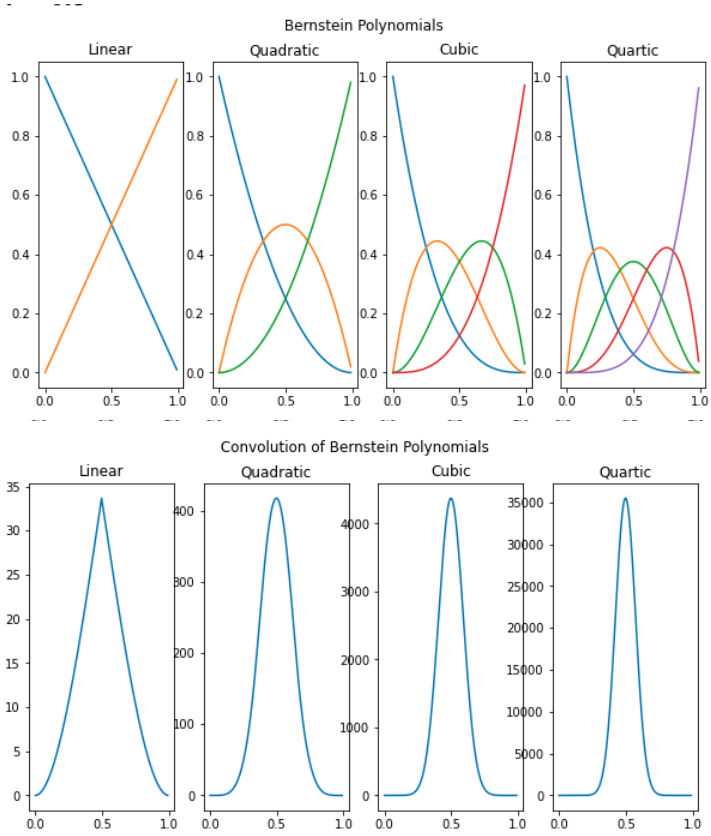


Figure 2.5: Bernstein functions of different degrees and their convolutions.

Here is an example: assuming $f = [1, 4, 2, 5]$, $g = [3, 4, 1]$, the result of convolution $c = f * g = [3, 16, 23, 27, 22, 5]$. Note that the relationship between the lengths i.e. $|c| = |f| + |g| - 1$.

The Bézier curve can be constructed using de Casteljau's algorithm[6] [FH00] based on the fact that the control point $p_{i,j}$ depends on $p_{i,j-1}$ and $p_{i+1,j-1}$ as shown in figure 2.6 [9].

In a Bézier curve, if one of the control points is changed, the whole curve changes. There is no local control. Another issue in the Bézier curve is that more control points are needed to represent complicated curves what means higher degree polynomials and more computation. The B-spline curve solves the two above issues and a better alternative for Bézier curves.

The B-spline curve is represented by $m-n$ Bézier segments, where n is the curve

[6]Note: Casteljau is the one who invented Bézier curves. Bézier is the one who made them popular.

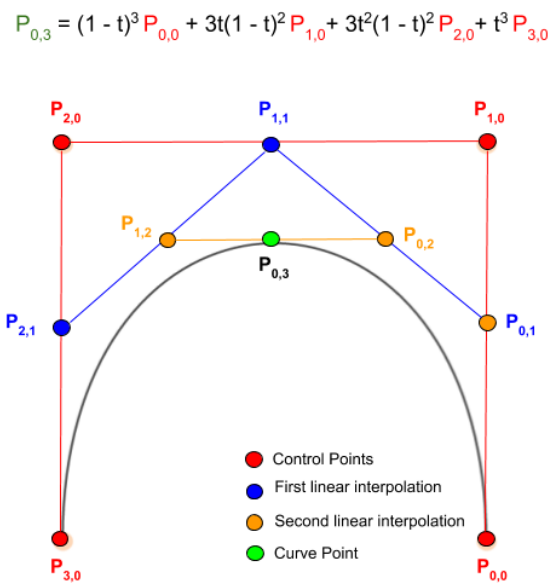


Figure 2.6: Cubic Bézier curve with 4 control points $p_{0,0}, p_{1,0}, p_{2,0}, p_{3,0}$ connected by the red line. Note: Image is generated using `www.ibiblio.org/e-notes` then modified in our lab.

degree and m is the total number of control points. Each segment is represented by a Bézier curve of degree n and $n + 1$ control points, see figure 2.7 top. The B-spline has $k = m - n + 1$ knots, where the first knot point is the first point in the curve $t_0 = C_0$ and the last knot point is the end point of the curve $t_k = C_{|C|-1}$. The knots are in a non-decreasing sequence. To make the end control-points are the curve-end points, more knots are added. We get that by making the first knots and the end knots are $n + 1$. This means the total number of knots is now $k = m - n + 1 + 2n = m + n + 1$ as shown in figure 2.7[31][31] bottom.

B-spline transform represents an image using its control-points. The dimension of the control-points grid is the same as the input image's dimension, e.g. if the input image is 3D, the grid is also 3D. The grid size is a variable, but it must satisfy that the number of control-points per each dimension to be larger than the B-spline degree. In this example, for a cubic B-spline it must be larger than 3.

By changing these control-point locations, one gets different transformed versions of the input image. One can also transform a specific part or parts of the image by changing only specific control-point locations.

The transformed images in figure 2.8 are generated by using random values for the 2D cubic B-spline transform control-points. The size of the control-points are varied from 16 to 100. The red arrows show the random displacement field. Notice that the smaller number of control-points means the less deformation of the

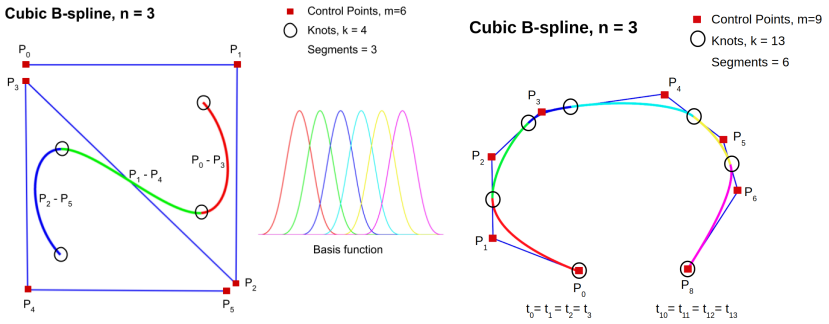


Figure 2.7: Cubic B-spline curve. Left: 6 control points and 4 knots, basis functions are shown as well. Each segment is controlled by 4 control points. Right: 9 control-points and 13 knots, the first and the end knot-points are $n+1$ knots each, this way the end points of the curves are control points. Note: Image is generated using `www.ibiblio.org/e-notes` then modified in our lab.

transformed image. The B-spline representation of the image has other desirable features like fast image filtering or derivative computation [Uns99].

The B-spline of degree n can be constructed using a generalisation of de Casteljau called Cox-de Boor recursion. To compute the B-spline curve points over an interval $X = x_0, x_1, ...$ we use:

$$C = \sum_{x_j \in X} \sum_{i=0}^{m-1} \beta_i^n(x_j, K) P_i \tag{2.33}$$

where P_i is a control-point, m is the number of control points, K is the knots vector of length k and β_i^n is the B-spline basis function of degree n defined as:

$$\begin{aligned}
\beta_0^n(x, K) &= \begin{cases} 1 & \text{if } t_i \le x < t_{i+1}, \quad t_i, t_{i+1} \in K \\ 0 & \text{otherwise} \end{cases} \\
\beta_i^n(x, K) &= C_1 + C_2 \\
C_1 &= \begin{cases} 0 & \text{if } t_{i+n} - t_i = 0 \\ \frac{x - t_i}{t_{i+n} - t_i} \beta_i^{n-1}(x, K) & \text{otherwise} \end{cases}, \\
C_2 &= \begin{cases} 0 & \text{if } t_{i+n+1} - t_{i+1} = 0 \\ \frac{t_{i+n+1} - x}{t_{i+n+1} - t_{i+1}} \beta_{i+1}^{n-1}(x, K) & \text{otherwise} \end{cases}.
\end{aligned} \tag{2.34}$$

The B-spline equation is modified to represent how actually it is implemented as the original equations found in the literature create confusion sometimes. The confusion comes from ignoring the division by zero and the knot point is used as the curve point while it is actually a point in the curve that connects two sub-curves.

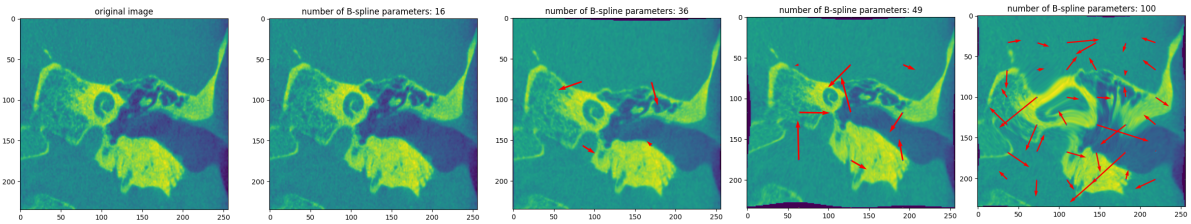


Figure 2.8: 2D B-spline transform of a 2D cochlea coronal view. When the number of control-points is small there is no deformation. More control-points may produce more deformation.

In image registration, we first compute the control-points of the input image and consider them as initial values of the B-spline transform parameters. During the optimisation process we keep changing them until the cost function is minimised.

As described by Unser et al. [UAE91], a different implementation can be achieved using the signal theory and the Z-transform. It is the implementation found in ITK package as it simplifies computing the derivative, mutual information, interpolation and filtering of medical images.

The B-spline basis function can be defined by self-convolution $n+1$ times with $\beta^0(x)$ which is a centered rectangle around origin[7].

$$\begin{aligned} \beta_+^n(x) &= \underbrace{\beta_+^0(x) * \beta_+^0(x) * ... * \beta_+^0(x)}_{\text{convolution n+1 times}}, \\ \beta_+^0(x) &= \begin{cases} 1, & x \in [0,1) \\ 0, & otherwise. \end{cases} \end{aligned} \tag{2.35}$$

The digital filter takes an input of a signal $f[k]$ and output of a signal $c[k]$ that is a convolution of an interpolation signal h_{int} and the input signal f, i.e. $c[k] = (h_{int} * f)[k]$.

$$H_{int}(z) = \frac{1}{B(z)} = \frac{1}{\sum_{k \in Z^q} \phi(k) z^{-k}} \tag{2.36}$$

An example of the cubic B-spline basis function is:

[7]Note that the number n in β^n does not mean "to the power n", it is just a convenient way to describe the degree of the B-spline basis function because of the use of recursion.

$$\beta^3(x) = \begin{cases} \frac{2}{3} - |x|^2 + \frac{1}{2}|x|^3, & 0 \leq |x| < 1 \\ \frac{1}{6}(2 - |x|)^3, & 1 \leq |x| < 2 \\ 0, & otherwise. \end{cases} \tag{2.37}$$

For $\beta^3(x)$, the discrete B-spline kernel is $B(z) = \frac{z+4+z^{-1}}{6}$ and the interpolation filter is

$$\frac{6}{z + 4 + z^{-1}} = \frac{(1-\alpha)^2}{(1-\alpha z)(1-\alpha z^{-1})} \overset{z}{\longleftrightarrow} h_{int}[k] = \left(\frac{1-\alpha}{1+\alpha}\right)\alpha^{|k|} \quad ,$$

where α is the pole value and $\alpha = -2 + \sqrt{3} = -0.171573$. The process has a cascade of first order recursive filters that contains a casual filter $\frac{1}{1-\alpha z^{-1}}$ and an anti-casual filter $\frac{1}{1-\alpha z}$.

2.3 Cost function

The optimisation cost function of the image registration is a similarity metric function that minimizes the similarity error between the input images. It is important to have the similarity metric and its derivative with respect to the transformation parameters in order to update the transformation parameters.

Mean Square Error (MSE) metric is useful in the case of mono-modal image registration, i.e. input images come from the same imaging modality, the image intensity at corresponding points between the two images should be similar. MSE is one of the simplest measures of the similarity. It is defined as:

$$S_{MSE}(I_F(P), I_M(T(P,\mu)) = \frac{1}{n}\sum_{i=0}^{n-1}(I_F(p_i) - I_M(T(p_i,\mu)))^2, \tag{2.38}$$

where:

- $I_F(P), I_M(P)$: the fixed and the moving images. P is the point coordinates of each pixel.
- $T(P,\mu)$: a transformation with μ parameters. It inputs a point (or points) and outputs a transformed point (or transformed points). For simplicity, I will use the 2D translation transform which is defined in equation (2.12) as an example.
- P is the image coordinates vector. An example for such vector is the vector of all points of a 2D image $P = [(x_0, y_0), (x_1, y_1), ..., (x_{n-1}, y_{n-1})]$, where n is the number of pixels in the image.
- μ is the vector of the transformation parameters. For example, the parameters of the 2D translation transform, where μ includes only two parameters - a shift in x coordinate and a shift in y coordinate.
- n is the number of pixels in the image.

The derivative of this metric is:

$$\frac{\partial S_{MSE}}{\partial \mu} = \frac{2}{n}\sum_{i=0}^{n-1}\left((I_F(p_i) - I_M(T(p_i,\mu)))\left(\frac{-\partial I_M(T(p,\mu))}{\partial \mu}\right)\right) \tag{2.39}$$

Using chain rule, we get:

$$\frac{\partial S_{MSE}}{\partial \mu} = \frac{2}{n}\sum_{i=0}^{n-1}\left(c_1\ c_2\ \frac{\partial (T(p_i,\mu))}{\partial \mu}\right), \tag{2.40}$$

where:

- $c_1 = (I_M(T(p_i,\mu) - I_F(p_i)))$ is constant during the optimisation.
- $c_2 = \frac{\partial I_M(T(p,\mu))}{\partial T(p,\mu)}$ is also constant during the optimisation. We compute the gradient of the moving image $\nabla I_M = \frac{\partial I_M(P)}{\partial P}$ (one time operation before the optimisation). During the optimisation we get a transform point $\tilde{p}_i = T(p_i,\mu)$, then we get the intensity value from the moving image gradient at a transformed point's location $v = \frac{\partial I_M(\tilde{p})}{\partial \tilde{p}}$.
- $\frac{\partial (T(p_i,\mu))}{\partial \mu}$ is a row from J_μ (see equation (2.41)) that represents the i_{th} transformed point derivative with respect to each transformation parameter $\mu_j \in \mu$. J_μ is the Jacobian of the transformation with respect to its parameters as shown in equation (2.41). J_μ is computed during each iteration of the optimisation. Note that $J_\mu = \frac{\partial T(P,\mu)}{\partial \mu}$ is different from the Jacobian of the transformation $J = \frac{\partial T(P,\mu)}{\partial P}$

$$J_\mu = \frac{\partial T(P,\mu)}{\partial \mu} = \begin{pmatrix} \frac{\partial T(p_1,\mu_1)}{\partial \mu_1} & \frac{\partial T(p_1,\mu_2)}{\partial \mu_2} & \frac{\partial T(p_1,\mu_3)}{\partial \mu_3} & . & . & \frac{\partial T(p_1,\mu_m)}{\partial \mu_m} \\ \frac{\partial T(p_2,\mu_1)}{\partial \mu_1} & \frac{\partial T(p_2,\mu_2)}{\partial \mu_2} & \frac{\partial T(p_2,\mu_3)}{\partial \mu_3} & . & . & \frac{\partial T(p_2,\mu_m)}{\partial \mu_m} \\ \frac{\partial T(p_3,\mu_1)}{\partial \mu_1} & \frac{\partial T(p_3,\mu_2)}{\partial \mu_2} & \frac{\partial T(p_3,\mu_3)}{\partial \mu_3} & . & . & \frac{\partial T(p_3,\mu_m)}{\partial \mu_m} \\ & & . & & & \\ & & . & & & \\ & & . & & & \\ \frac{\partial T(p_n,\mu_1)}{\partial \mu_1} & \frac{\partial T(p_n,\mu_2)}{\partial \mu_2} & \frac{\partial T(p_n,\mu_3)}{\partial \mu_3} & . & . & \frac{\partial T(p_n,\mu_m)}{\partial \mu_m} \end{pmatrix}, \tag{2.41}$$

where P is the image points coordinates vector, p_1 is a point belonging to this vector, T is the transformation, μ is the vector of the transformation's parameters, μ_m is a parameter element belonging to this vector, n is the number of points in

the image, m is the number of the parameters of the transform.

Since I could not find any example in the literature, I provide one here, let us assume:

- a 2D point $p = (x, y) = (2, 3)$.
- a 2D translation transform:

$$T(p, \mu) = T((x, y), (\mu_0, \mu_1)) = (x + \mu_0, y + \mu_1).$$

We have :

$$\frac{\partial T(p,\mu)}{\partial \mu} = \frac{\partial T((x,y),(\mu_0,\mu_1))}{\partial(\mu_0,\mu_1)} = \begin{pmatrix} \frac{\partial(x+\mu 0, y+\mu_1)}{\partial \mu_0} \\ \frac{\partial(x+\mu 0, y+\mu_1)}{\partial \mu_1} \end{pmatrix} = \begin{pmatrix} \frac{\partial(x+\mu 0)}{\partial \mu_0} & \frac{\partial(y+\mu_1)}{\partial \mu_0} \\ \frac{\partial(x+\mu 0)}{\partial \mu_1} & \frac{\partial(y+\mu_1)}{\partial \mu_1} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{2.42}$$

For our point we have:

$$\frac{\partial T((2,3),\mu_0)}{\partial \mu_0} = (1, 0), \frac{\partial T((2,3),\mu_1)}{\partial \mu_1} = (0, 1) \tag{2.43}$$

The final result should have a vector with the size of the point dimension. It seems that for updating the transformation parameters we just multiply the gradient of the transformed point with twice the error value,and then we divide it by the number of pixels.

Mutual Information (MI) metric has become standard for multi-modality problems. This technique does not require information about the surface properties of the object, besides its shape. It is robust with respect to variations of illumination in addition to the fact that it is quite general, i.e. it is foreseeable to be used in a wide variety of imaging situations. The idea is to use the mutual information between the object and the image, that is why it can be considered intensity approach rather than feature approach. As key-point detection and features are not easy to match between CBCT, CT and MRI images. This metric is defined using the entropy of the images:

$$S_{MI}(I_F, I_M(T(P, \mu))) = h(I_F) + h(I_M(T(P, \mu))) - h(I_F, I_M(T(P, \mu))), \tag{2.44}$$

where $h(I_F)$, $h(I_M(T(P, \mu)))$, $h(I_F, I_M(T(P, \mu)))$ are the estimations of the fixed image entropy, transformed moving image entropy and the joint entropy of both fixed and moving images.

$$h(X) = \begin{array}{ll} \sum_{x_i \in X}(p(x_i).log_2(p(x_i))), & p(x_i)0 \\ 0, & p(x_i) \leq 0 \end{array}$$

$$h(X,Y) = \begin{array}{ll} -\sum_{x_i \in X} \sum_{y_j \in Y} (p(x_i, y_j).log_2(p(x_i, y_j))), & p(x_i, y_j)0 \\ 0, & p(x_i, y_j) \leq 0 \end{array}$$

The cost function can be rewritten as:

$$S_{MI}(I_F, I_M(T(P, \mu))) = \sum_{x \in I_F} \sum_{y \in I_M(T(P,\mu))} \left(p(x,y) log_2 \left(\frac{p(x,y)}{p(x), p(y)} \right) \right) \tag{2.45}$$

This requires the estimation of Probability Density Functions (PDF) from the input images. This PDF estimation uses Parzen windows [Par62]. Viola and Wells [VW97] explains the approach in details. I give here a brief explanation for an implementation purposes.

Two groups A and B of random samples of lengths N_A and N_B are created. The samples contain intensity values taken randomly from the fixed and the transformed moving images as follows:

- Group A contains samples with N_A values from the fixed image $A_F(P_{r_1})$ and samples from the transformed moving image $A_M(T(P_{r_1}))$ at P_{r_1} random locations.

- Group B contains samples with N_B values from the fixed image $A_F(P_{r2})$ and samples from the transformed moving image $A_M(T(P_{r2}))$ at P_{r2} random locations.

Note that for each sample set we get the same locations from the fixed and the moving images, e.g. $\mathbf{A}_{F_i}$, and $\mathbf{A}_{M_i}$ are from the same location i, $\mathbf{A}_{F_i}$ from the fixed image and $\mathbf{A}_{M_i}$ from the moving image. We will use these samples to estimate the entropy of the fixed and the moving images. The estimated entropy can be defined as:

$$h(X) \approx E_X(-log_2(p(X))) = \frac{-\sum_{x_i \in B_X} (log_2(p(x_i)))}{N_B}, \tag{2.46}$$

where X is an input image, E_X is the estimated expected value of the image (the image mean), B_X is a sample set from the image X, $p(X)$ is the estimated image probability density function which can be computed using the Parzen window with the Gaussian kernel:

$$p(X) \approx \frac{1}{N_A} \sum_{x_j \in A_X} G(x - x_j, \psi), \tag{2.47}$$

where X is an input image, A_X is a sample set from image X with length N_A. G is the Gaussian function centered at x with a standard deviation ψ that represents the kernel bandwidth. The Gaussian G function is defined as:

$$G(x - \tilde{x}, \psi) = \frac{1}{\psi\sqrt{(2\pi)}} e^{-\frac{1}{2}(\frac{x-\tilde{x}}{\psi})^2} \quad , \tag{2.48}$$

where x is an intensity value from the sample set, x is the mean value of the image, ψ is the Gaussian kernel width.

By plugging the equation (2.47) in the equation (2.46) we get the estimation of the entropy of an image X.

$$h(X) \approx \frac{-\sum_{x_i \in B}(log_2(\sum_{x_j \in A} G(x_i - x_j, \psi)) - log_2(N_A))}{N_B} \quad . \tag{2.49}$$

From (2.44), the derivative of the mutual information metric is:

$$\frac{\partial S_{MI}(Y, X(T(P, \mu)))}{\partial \mu} \approx \frac{\partial h(X(T(P, \mu))))}{\partial \mu} - \frac{\partial h(X(T(P, \mu))), Y)}{\partial \mu} \quad , \tag{2.50}$$

where X is the fixed image I_F and X is the moving image I_M. By plugging (2.49) in (2.50) we get:

$$\frac{\partial S_{MI}}{\partial \mu} \approx \frac{\sum_{y_i, x_i \in B} \sum_{y_j, x_j \in A} \left(\frac{(x_i - x_j)}{\psi_x} (W_x - W_{yx}) \frac{\partial X(T(x_i - x_j, \mu))}{\partial x} \frac{\partial T(x_i - x_j, \mu)}{\partial \mu} \right)}{N_B} \quad , \tag{2.51}$$

where

$$W_x = \frac{G(x_i - x_j, \psi_x)}{\sum_{x_k \in A} G(x_i - x_k, \psi_x)},$$

and

$$W_{yx} = \frac{G((y_i - y_j, x_i - x_j), \psi_{y,x})}{\sum_{y_k, x_k \in A} G((y_i - y_k, x_i - x_k), \psi_{y,x})} \quad .$$

The term $\frac{\partial X(T(x_i - x_j, \mu))}{\partial x}$ is the gradient of the transformed moving image samples. The term $\frac{\partial T(x_i - x_j, \mu)}{\partial \mu}$ is the J_μ of the moving image samples which is similar to equation (2.41).

Finally, I should note that there is a typo in the original mutual information paper [VW97]. In equation 17, the weight function $W_{uv}(w_i, w_j)\psi_{vv}^{-1}$ should be actually $W_{uv}(w_i, w_j)\psi_v^{-1}$.

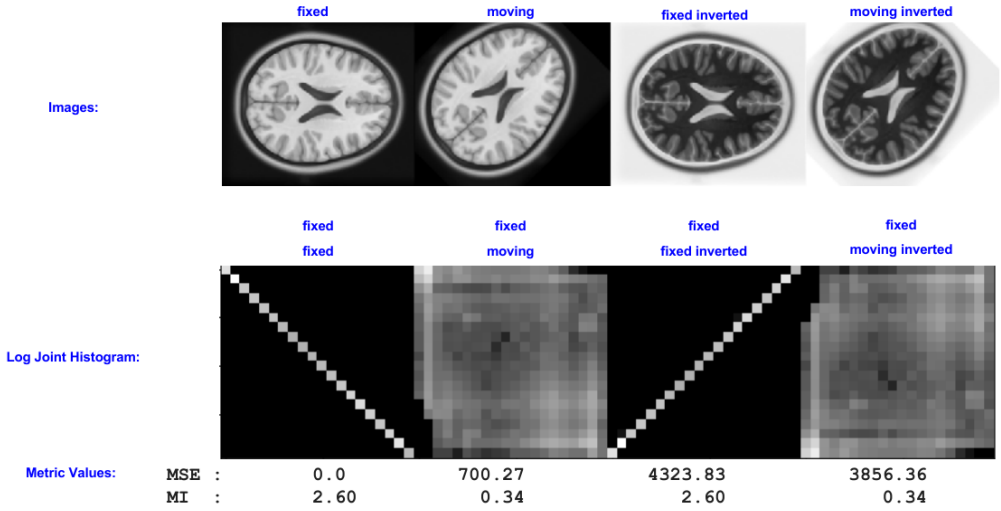


Figure 2.9: Cost functions comparison MSE vs MI. The brain image is taken from 3D Slicer sample dataset.

Mattes's Mutual Information (MMI) [MHV+01] is an enhancement of Viola and Well's Mutual Information. It is considered as the state of art that has been shown to be suitable for multi-modal and mono-modal medical image registration [KPSV09].

$$S_{MMI}(I_F(P), I_M(T(P,\mu))) = -\sum_{l_{H_F}}\sum_{l_{H_M}} p(l_{H_F}, l_{H_M}|\mu) log\left(\frac{p(l_{H_F}, l_{H_M}|\mu)}{p\mathrm{M}(l_{H_F}|\mu)p\mathrm{F}(l_{H_M}))}\right) \tag{2.52}$$

where l_{H_F} and l_{H_M} are the lengths of the histogram bins of the fixed image $I_F(P)$ and the transformed moving image $I_M(T(P,\mu))$. p, $p_M$and p_F are the joint marginal probability distributions of the fixed and the transformed moving images. These probabilities are estimated from B-spline representation of the images.

The figure 2.9 shows a comparison between the two popular cost functions MSE and MI. Notice that for a similar image with different colors, MI is more accurate.

2.4 Optimisers

The optimisation problem is stated in equation (2.1), gradient descent optimiser is already mentioned earlier in this chapter.

Stochastic Gradient Descent (SGD) approach [RM51, KSP07] uses samples instead of the whole images. Sometimes these samples are taken randomly. The

stochastic approach allows fewer computations and provides practical results in practice. The parameters updating rule of SGD is:

$$\mu_{k+1} = \mu_k - \lambda \frac{\partial S(\hat{I}_F(P), \hat{I}_M(T(P, \mu_k)))}{\partial \mu}, \tag{2.53}$$

where $\hat{I}_F$ and $\hat{I}_M$ represent samples from the input images I_F and I_M.

The step-size λ in the gradient descent methods, e.g. in equations (2.2) and (2.53), is an important optimisation factor that has to be set manually. It has a large influence on the gradient descent-based optimisation methods. This factor is data-dependent, so finding a suitable value for different problems is challenging.

Adaptive Stochastic Gradient Descent (ASGD) [KPSV09] is an optimiser that adapts the step-size factor λ automatically using an image-driven mechanism to predict its value. It replaces λ with a parameter δ that defines the maximum incremental displacement allowed between the optimisation's iterations[8]. The δ parameter is computed using the voxel[9] size of the input image. The mean voxel spacing (in mm) seems to be a good value as larger values lead to more aggressive optimisation [KPSV09]. This optimiser will be explained in more details in a later chapter.

Fast Adaptive Stochastic Gradient Descent (FASGD) [QvLS16] is another adaptive stochastic optimiser that estimates the step size automatically using the observed voxel displacement. A free parameter δ defines the maximum incremental displacement allowed between iterations. The δ parameter is computed using the voxel size (mm). During the computation of the step size, two parameters, i.e w and f_{min}, are fixed in FASGD. This is faster but less adaptive than ASGD.

Newton methods use an approximation of the function to find a function extremum. A function has its extremum when its derivative[10] is zero. Finding an extremum is thus equivalent to finding the root of the derivative (or gradient) function. That is the reason many root finding algorithms find their use in optimization algorithms. Perhaps the oldest algorithm to find roots it the Newton's method. If f is a function of x then to find an extremum of f we must find an x_0 such that $f'(x_0) = 0$. Thus, the problem of finding an extremum of f is identical to finding the root of its derivative. Taylor series [3] approximate a function as:

[8] In elastix, δ is the maximum step length parameter.

[9] The term pixel is short for picture element. Voxel is a pixel in a 3D image.

[10] gradient if it is a multi-variable function

$$f(x) \approx \sum_{n=0}^{\infty} (\frac{f^{(n)}(x_0)}{n!})(x-x_0)^n = f(x_0)+f'(x_0)(x-x_0)+\frac{f''(x_0)}{2}(x-x_0)^2+... \quad (2.54)$$

Now by replacing x_0 by x, and x by $x+h$:

$$f(x+h) \approx f(x)+f'(x)(x+h-x)+\frac{f''(x)}{2}(x+h-x)^2 \approx f(x)+h\,f'(x)+\frac{h^2}{2}f''(x) \quad (2.55)$$

The Taylor expansion of f up to the quadratic terms is:

$$f(x_k+h) \approx f(x_k)+h\,f'(x_k)+\frac{h^2}{2}f''(x_k). \quad (2.56)$$

The approximate form will have a minimum at:

$$h = \frac{-f'(x_k)}{f''(x_k)}. \quad (2.57)$$

Therefore, we can update our root to:

$$x_{k+1} = x_k + h = x_k - \frac{f'(x_k)}{f''(x_k)}. \quad (2.58)$$

Then we reach the extremum, $x_{k+1} = x_k$. This idea is generalized to the case when f is a function of more than one variable by replacing $f'(x_k)$ with the gradient $\nabla f(x_k)$ and the reciprocal of the second derivative by the inverse of the Hessian matrix H_k. Thus, equation (2.58) becomes:

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}. \quad (2.59)$$

By putting $f'(x_k) = \nabla f(x_k)$ and $\frac{1}{f''(x_k)} = H_k^{-1}$, the update rule for Newton optimiser is:

$$x_{k+1} = x_k - H_k^{-1}\nabla f(x_k). \quad (2.60)$$

Stochastic Quasi-Newton (S-LBFGS) [BHNS16] is the Newton optimiser method where the second-order derivative of the cost function is considered. GD assumes that the second-order derivative (Hessian) is the identity matrix I. In some cases

where the Hessian matrix is ill-conditioned, using GD produces inefficient optimisation result. The Newton approach is computationally very expensive as computation of the Hessian requires a lot of time. Stochastic quasi-Newton Methods [BHNS16] use an approximation of the Hessian of the cost function which reduces the computation time. The update function in quasi-Newton optimiser is:

$$\mu_{k+1} = \mu_k - \lambda.H^{-1}\frac{\partial S(I_F(P), I_M(T(P,\mu)))}{\partial \mu} \tag{2.61}$$

where k is the optimisation iteration, λ is the step size and H is an approximation of the second-order derivatives (Hessian matrix H). Hence,

$$H \simeq \mathbf{H} = \frac{\partial^2 S(I_F(P), I_M(T(P,\mu))}{\partial \mu^2}.$$

A popular quasi-Newton update rule is Broyden-Fletcher-Goldfarb-Shanno (BFGS) [NW06] which uses the first-order derivatives to update the inverse Hessian directly, see equation (2.60). This produces linear rate convergence.

$$H_{k+1} = V_k^T.H_k.V_k + \rho_k.s_k.s_k^T \tag{2.62}$$

where:
$s_k = \mu_{k+1} - \mu_k$, $y_k = g_{k+1} - g_k$, $g = \frac{\partial S(I_F(P), I_M(T(P,\mu)))}{\partial \mu}$, $\rho_k = \frac{1}{y_k^T.s_k}$, and $V_k = I - \rho_k.y_k.s_k^T$

However, this still requires a long computation time and a large memory. The memory issue is solved by the Limited memory BFGS (LBFGS) [SW17] method. It reduces the required memory by saving only the last Hessian approximations. The computation time issue can be solved by using a stochastic approach [QSLS15]. A stochastic approach uses only small parts of the data taken randomly at each optimisation iteration instead of all the data. However, it introduces a large amount of noise as well [QSLS15].

Fast Preconditioned Stochastic Gradient Descent (FPSGD) is an enhancement of [KSAP11] which was proposed to improve the rate of convergence of GD methods by adding a preconditioning strategy to Robbins-Monro Stochastic Gradient Descent (SGD_{RM}) [RM51] and ASGD methods. The updating rule of PSGD is:

$$\mu_{k+1} = \mu_k - \lambda M\frac{\partial S_k(\hat{I}_F(P), \hat{I}_M(T(P,\mu_k)))}{\partial \mu}. \tag{2.63}$$

The equation is similar to the equation (2.53). The only difference is adding the preconditioned matrix M which is a symmetric positive definite matrix with the size of number of the transformation parameters, i.e. $|\mu|$. When M is the identity matrix, we get the standard SGD method as in the equation (2.53). When M

is the inverse of the Hessian of the similarity metric, we get a Stochastic quasi-Newton method, see equation (2.61). In [KSAP11] the step-size is defined to be a non-increasing and non-zero sequence to guarantee convergence:

$$\lambda_k = \begin{cases} 1 & \text{if } k = 0 \\ \\ \frac{\eta}{((t_k+1)/A)+1} & \text{if } k > 0 \end{cases} \tag{2.64}$$

, where

$t_0 = t_1 = 0$,
$t_k = max(0, t_{k-1} + sigmoid(\tilde{g}_{k-1} M \tilde{g}_{k-2})$
$\tilde{g}$ is an approximation of $\frac{\partial S_k}{\partial \mu}$,
M is the preconditioner matrix,
$A = 20$ is a decay speed factor,
η is a noise factor and can be defined as:

$$\eta = \frac{E||g^T M g||}{E||\tilde{g}^T M \tilde{g}||} = \frac{E||g^T M g||}{E||g^T M g|| + E||\epsilon^T M \epsilon||}, \tag{2.65}$$

where g is the exact gradient and ϵ is a random noise added to the exact gradient.

One can think of M as a scaling factor for $\tilde{g}$. It is large when $\tilde{g}$ is smooth and small when $\tilde{g}$ has a curvature. PSGD in [KSAP11] was proposed to solve mono-modal image registration. This means the fixed and the moving images are the same type, e.g. CT. However, it is practically desirable for a medical image registration method to handle the problems both mono-modal and multi-modal at the same time.

FPSGD [QLS19] method proposed to solve both mono-modal and multi-modal image registration. It estimates the diagonal entries of a preconditioning matrix M on the distribution of voxel displacements. This rescales the registration similarity metric and produces more efficient optimisation. The authors mention that FPSGD is between 2 to 5 times faster than SGD methods while retaining the same accuracy.

Let M be a preconditioned matrix with the same length of the parameter vector μ, then MI is an element in the diagonal of M that can be computed using:

$$m_i = \frac{\delta}{E(||J_{\mu_i}||.||\tilde{g}_i||) + 2\sqrt{Var(||J_{\mu_i})||.||\tilde{g}_i||)} + \epsilon}, \tag{2.66}$$

where δ is a pre-defined value represents the maximum voxel displacement, $E||X||$ is the expectation of the l_2 norm, $Var||X||$ is the variance of the l_2 norm, J_{μ_i} is the i_{th} column in the Jacobian matrix $J_\mu = \frac{\partial T(P,\mu)}{\partial \mu}$ which is mentioned earlier in equation (2.41), ϵ is a small number to avoid dividing by zero.

2.5 Extra Image Registration Components

There are other important concepts I am covering in this section as they are used in this study.

Multi-registration approach is used when the image registration problem can not be solved by the one-stage optimisation. In this approach, the resulting output registered image is input to another stage optimisation process where one or more components of the optimisation may be changed. Usually, a rigid transformation is used in the first stage and non-rigid transformation is used in the second stage.

Multi-resolution approach is also helpful to be used. It is a hierarchical approach that solves some difficult image registration problems. There are two approaches of hierarchical image registration[11], the first approach is down-sampling and the second one is smoothing.

Down-sampling solves the registration problem at different levels. Suppose we have four levels of multi-resolutions, for a fixed image I_F and a moving image I_M of size 512x512, the first level has I_{F_1} and I_{M_1} of size 64x64, the second level has I_{F_2} and I_{M_2} of size 128x128, the third level has I_{F_3} and I_{M_3} of size 256x256 and the final level has the sizes of the original images, i.e. 512x512, see figure 2.10 top. Each level produces transform parameters that can be used as initial parameters for the next level.

In smoothing multi-resolution, the size does not change, only the details change, e.g. more smoothing at each level, see figure 2.10 bottom. Each approach has advantages and disadvantages and it is good to experiment with both to see which one fits the registration problem better.

[11] In Elastix RecursiveImagePyramid control the multi-resolution type.

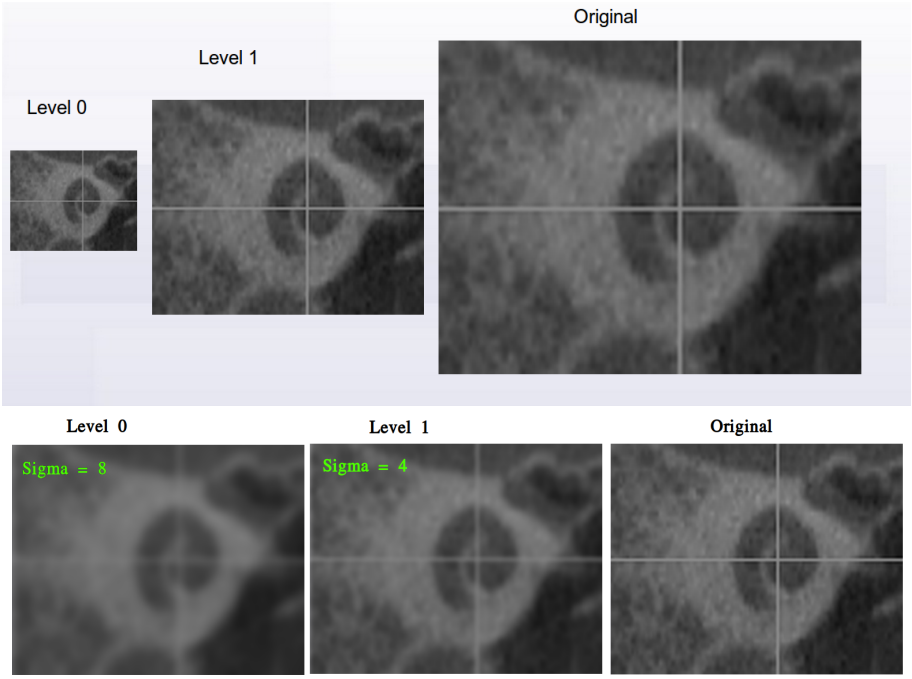


Figure 2.10: Coronal slices show a cochlea image pyramid with 3 multi-resolutions. Top: Spatial multi-resolution, bottom: smoothing multi-resolution with Gaussian smoothing with two different Sigma values 4.0 and 8.0.

Chapter 3

Medical Image Segmentation

Image segmentation is a process of extracting a desired object (or objects) from an input image. It has many real-world applications in different fields. In medicine, image segmentation can be used to extract a specific organ and measure its size, or retrieve other information from the segmentation result. Efficient automatic segmentation algorithms may help automating the cochlear image analysis. The segmentation is similar to the data classification problem where the pixels of the images are classified into different classes, see figure 3.1.

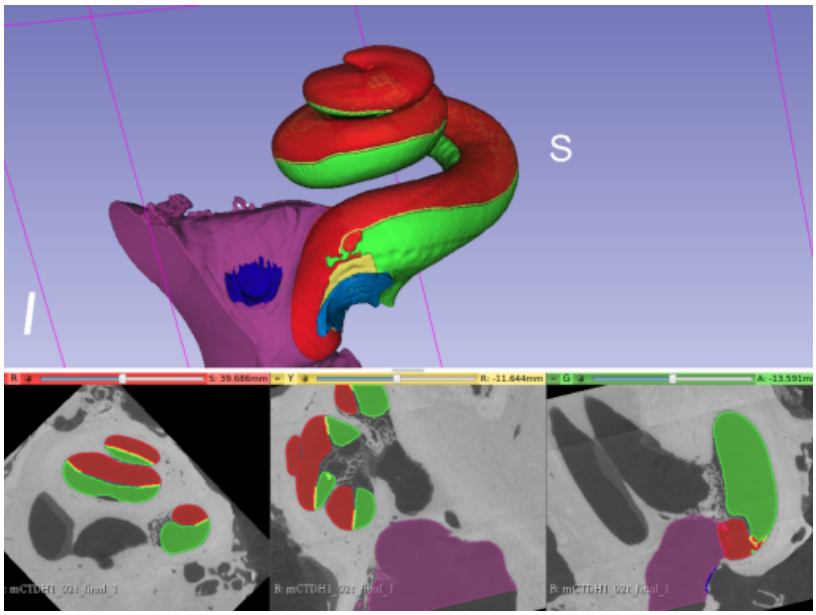


Figure 3.1: Cochlea Segmentation from μCT image in three 3D views: axial, sagittal and coronal. Green: scala tympani, red: scala vestibuli and scala media.

There are many different methods for image segmentation, e.g. edge-based,

region-based and atlas-based.

Threshold image segmentation is probably the simplest image segmentation method. It separates the background from the foreground based on a threshold value, see figure 3.2 right.

3.1 Edge-based image segmentation

Edge-based image segmentation is a result of convolution of an input image with an edge detector, e.g. Sobel, Prewitt, Canny or Lablacian of Gaussian (LoG) [GW06], see figure 3.2 middle. The Sobel edge detector kernels are defined as:

$$S_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}, S_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} \tag{3.1}$$

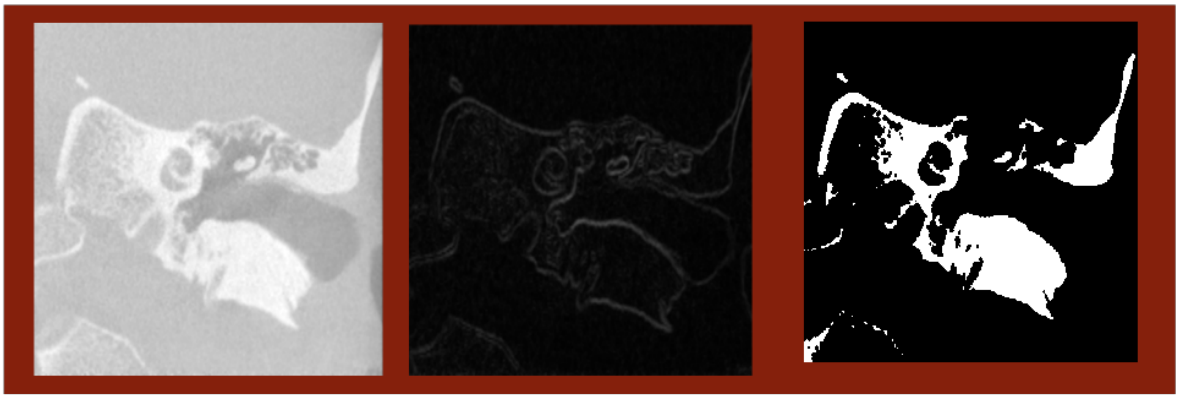


Figure 3.2: Left: input image. Middle: result of Sobel filter. Right: result of threshold segmentation.

3.2 Region-based segmentation

Region-based segmentation tries to classify the pixels based on the region they belongs to, e.g. region-grown[GW06], active contours (snakes)[KWT98] or graph-cuts [BJ01].

In region-grown class, an initialisation is one or multiple seeds. The seed check all of its neighbour pixels and decides if one or more of them belong to the segmentation or not, see figure 3.3.

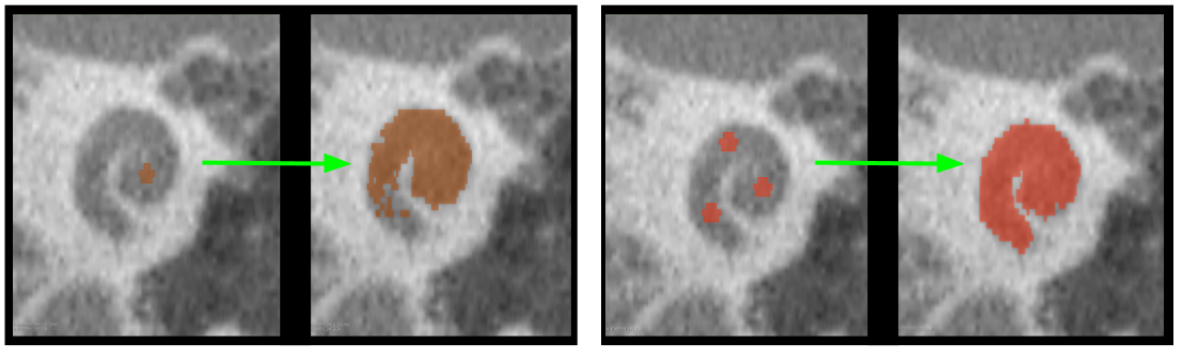


Figure 3.3: .
Region growing segmentation. Left: one seed, right: multiple seed. The green arrow show the input image (left of the arrow) and the output image (right of the arrow).

Graph-cuts uses method from graph theory to partition the image into segmentations. It construct a graph with edges and vertices. Each edge has a probability that decides if the edges belong to the same segmentation.

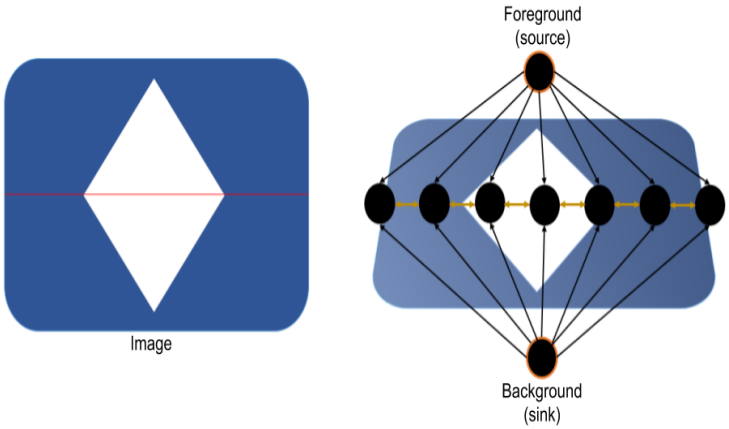


Figure 3.4: Graph cut segmentation concept. The right part represent the part from the image represented by the red line (the red line is not part of the image). The thickness in the arrow represent the probability.

Some image segmentation problems can be solved by using optimization e.g. minimizing a functional energy. An example of such energy is the snake energy.

Active Contours (Snake) method [KWT98], (in 3D, it is called Deformable Surfaces), is an edge based segmentation algorithm. The user provides an initial contour (snake) inside or outside the ROI. Then the algorithm moves the contour toward the boundaries of the object.

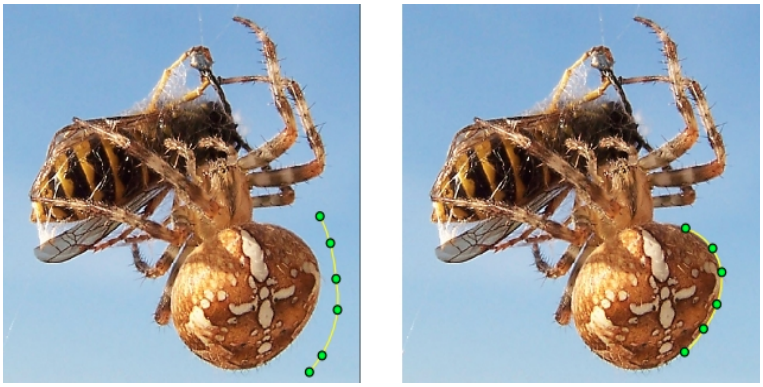


Figure 3.5: Active contour (snake) represented by a yellow line with green points. Left: initial position. Right: the contour moves to the edge and stopped. Image source is [7].

$$E_{Snake} = \alpha E_{internal} + \beta E_{external} \quad , \tag{3.2}$$

where the external energy is related to the input image and the internal energy related to the contour curve (the snake). The weights parameters α and β control how much each energy contributes to the total energy E_{Snake}. This energy can be minimized using optimization methods such as Gradient Descent [Sny05].

The update rule of the active contour:

$$Snake_{i+1} = Snake_i - \lambda \nabla E_{Snake_i} \quad , \tag{3.3}$$

where i is the iteration number and ∇ is the energy derivative. The step size λ controls how fast the control points move. Small λ produces slow movement, and a large λ produces fast movement but may produce a less accurate result.

Using Gradient Vector Flow (GVF) [XP97], an enhancement of the snake algorithm, the contour can evolve to reach a non-convex boundary e.g. such as the spiral shape of cochlea. The snake equation is changed by replacing the external energy with GVF energy E_{GVF}.

$$E_{Snake_{GVF}} = \alpha E_{internal} + \beta E_{GVF}. \tag{3.4}$$

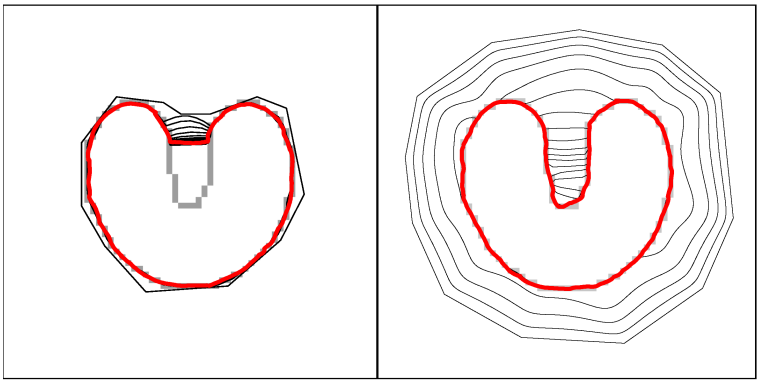


Figure 3.6: Convergence comparison: Snake (left) vs Snake with GVF. The red color shows the final segmentation.

Level-set image segmentation [Set99] represents the object as a zero level-set of a higher dimensional function Φ. In level-set we started usually with a Signed Distance Function (SDF) e.g. the signed distance transform of a random contour in the image. Then this function continues to change at each iteration until it reaches the boundary of the object, see figure 3.5.

This changing is based on an optimization process that computes the energy from the areas of the neighboring levels, then moves the function level either up or down. One important advantage of level-set method over the active contour method is segmenting separated objects.

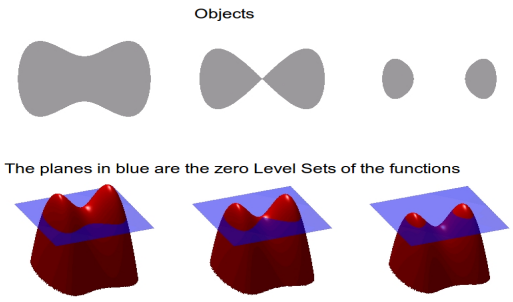


Figure 3.7: Level Set explanation[1]. Image source is [18].

There are many variations of Level Set methods e.g. Chan-Vese [CV01], Geodesic Active Contours (GAC) [CKS97] and GAC With Shapes [LGF00].

Chan-Vese method uses Mumford-Shah [MS89] method in a Level Set approach. The Mumford-Shah segmentation method segments the image by trying to achieve a balance between piece-wise smoothing and preserving the edges. It is one of the most popular region based segmentation algorithms. GAC uses active contours (snakes) in a Level Set approach. GAC with shapes uses a prior knowledge about

the shapes to get more accurate segmentation. In GAC with shape prior method, instead of providing a contour as input, the user provides number of shapes as a training input. The algorithm computes the Principal Component Analysis (PCA) and the mean shape of these training images. It uses the mean shape as an initial contour and as additional term in the optimization process.

3.3 Atlas-based Image Segmentation

Atlas-based segmentation [RBM+05] uses image registration concept to align an atlas or a pre-defined segmentation to an input image. The atlas usually is a well-defined histological image or a high resolution μCT image, as in see figure 3.1.

This method is very useful when a ground truth is missing, details of the segmentation is not visible, or when the segmentation boundaries are not clear.

This method reduces the segmentation problem to image registration problem and it work only if a registration method that produces the correct transformation is is available.

3.4 Model-based segmentation

Model-based segmentation e.g. Active Shapes Models (ASM) e.g. [CTCG95, ETC98], also called Smart Snake, is a very popular model based on a statistical knowledge about the object to be segmented. ASM tries to solve the problem of finding an alignment between a shape and an object by looking for strong edges or computing a distance e.g. Mahalanobis distance Mahalanobis. This method tries to fit a statistical shape model to the input image. The model is usually generated using many segmentation images. Figure 3.8 shows an example of a cochlea spiral shape.

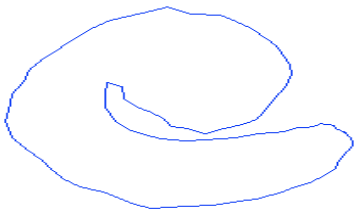


Figure 3.8: Example of a cochlea spiral shape. The cochlea scalae are not visible in this shape.

3.5 Deep learning

Medical image segmentation using deep learning, e.g. U-Net [RFB15], is currently considered the state of the arts in medical image semantic segmentation, see figure 3.9. In deep learning, semantic segmentation is a binary segmentation problem e.g. only two classes foreground and background. The multi-class segmentation is called instant segmentation.

U-Net uses convolution with different filters to learn different level of the image features through its layers starting from low to high. After that, it tries classifying the image pixels into background and foreground based on a provided example using a loss function that measure the error between the network outputs and the provided example.

This provides high accurate segmentation in a few seconds. However, deep learning based methods require many images to be segmented manually by experts which are not available all the time. They also require a special hardware, e.g. GPU with large memory, to perform fast.

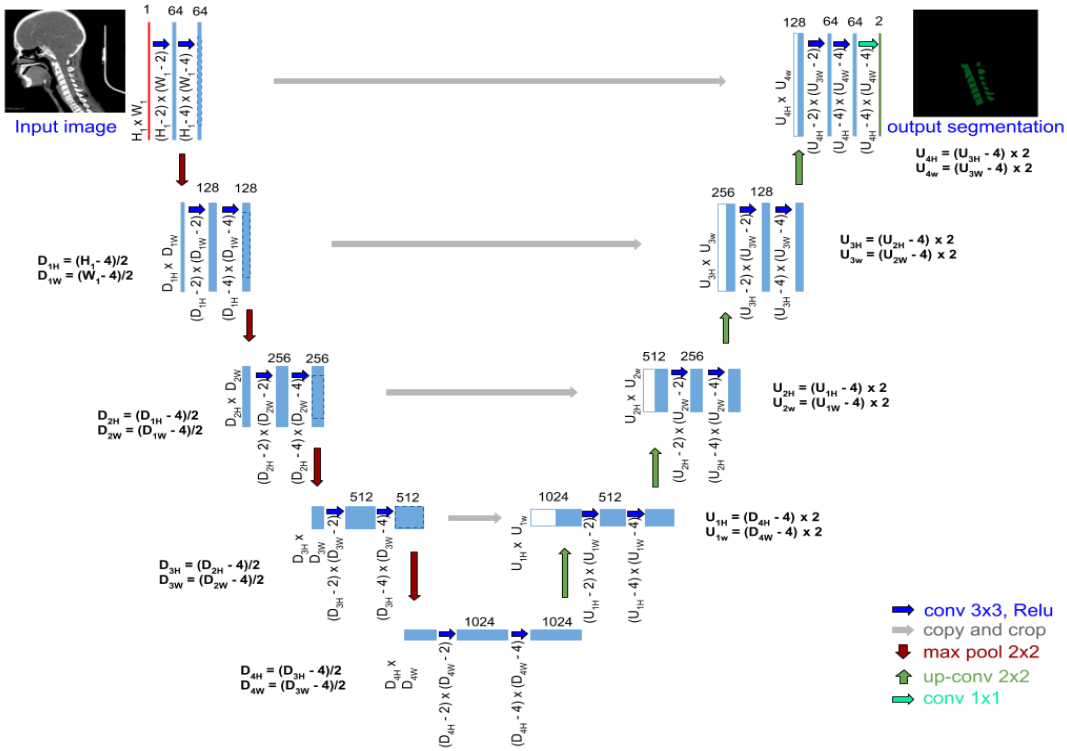


Figure 3.9: U-net structure, a deep learning method for image segmentation. Image is generalised version from the original structure in the U-Net paper.

Chapter 4

Literature and Related Work

There are a few scientific papers related to cochlea medical image processing. All cochlea related publications have neither a source-code nor dataset available what makes it very difficult to validate or reproduce their results.

I will describe here what I concluded from the papers I have read. In my conclusion, there are a few important points to be considered: the accuracy of the results,the time required for the processing, the type of the input images, e.g. mono-modal or multi-modal, and the reliability of the method.

A practical method must have high accuracy, required a few seconds to complete. It should work on different types of images, including on all input images, and it should not require special hardware.

For the scientific research sake, methods should be available freely, as an open-source with a sample dataset, e.g. one or two images. This helps other researchers to reproduce, validate and compare results of different methods.

4.1 Cochlea Image Registration

Manual image registration and fusion are usually done by doctors which requires much effort and time as in [BPR+05]. The manual procedure for image registration and fusion of CT and MR of the temporal bone is proposed using bony surgical landmarks. This procedure needs about 13 minutes per scan.

As described by Reda et al. [RNLD12], an automatic cochlea image registration for CT is proposed for Percutaneous Cochlear Implantation (PCI) surgery. They mentioned that a maximum of 0.19 mm error is achieved. The method is complicated as it involves using segmentation and two-stage registration. The computation time required to complete the registration is around 21 minutes which is larger than the time required for manual registration (about 10 minutes based on our experts). The authors investigated only complete head CT scans, so it is

not clear how the method works with multi-modal images or small images focused on the cochlea part.

In another research, high-resolution μCT data is used by Kjer et al. [KVF+15, KFV+16] where two methods were proposed for the cochlea image registration. The first method is based on the heat distribution similarity in a cubic B-splines registration model. The second method is based on skeleton similarity as an anatomical prior. Their methods require both cropped and segmented images, and it takes about 48 minutes per scan.

Finally, a proposed cochlea registration method by Dees et al. [DH16] where the fusion of sequential CBCT is compared to the gold standard fiducial in order to analyze clinical CI migration. BRAINSFit tool in 3D Slicer is used which makes the process require less than 2 minutes. Mean error of 0.16 mm is reported. However, this method does not support multi-modal as well and the images used had no implants. This makes the registration problem less challenging.

4.2 Cochlea Image Segmentation and Analysis

The goal of cochlea image segmentation is to extract the cochlea structure and measure its length and volume. This may help physicians to plan better the cochlea implant surgery and to select a suitable cochlea implant for each patient which will ideally provide better quality of hearing.

Many classical segmentation methods are already implemented in a well-known medical image software such as MITK [21] , 3D Slicer [25], ITK-Snap [26] and Matlab [19].

Since there is no general segmentation algorithm available, usually manual or semi-automatic segmentation is performed most of the time. It is not a surprise that this software fail to solve the cochlea segmentation problem even after trying many different parameters.

The segmentation results were either empty images, e.g. a black image or something different from the cochlea shape. Unfortunately, some important segmentation algorithms are not implemented in these tools, e.g. Chan-Vese [CV01] and ASM [CTCG95].

MITK provides a module for interactive 2D and 3D segmentation that supports Region Growing [Phi94] and Fast Marching [Set99] algorithms. Slicer3D has a module for interactive 2D and 3D segmentation that supports GrowCut [VK05], Watershed [BL79] and Fast Marching algorithms. ITK-Snap is a specialized segmentation tool based on ITK [17] and VTK [30], it supports Level Set[OS88, Set99] and Region Competition Snakes[ZLY96]. Matlab supports many segmentation algorithms such as Watershed, Texture Segmentation [MBLS01] and K-Means Clustering [HW79, Ste57].

I tested three algorithms provided in these tools on the cochlea datasets. The algorithms tested were Chan-Vese, GAC and ASM, see the previous chapter.

I re-grouped the images based on their modalities, i.e. CBCT, CT and MRI, and the cochlea side, i.e. left and right. After that, I applied some classical algorithms with different parameters. The result for each group is recorded using standard evaluation methods, e.g. Dice coefficient and Hausdorff distance standard segmentation metrics [Sø48, Dic45, Fel14].

Tables 1,2, and 3 show the selected parameters for each method. These parameters are selected based on many experiments and they should provide the best segmentation results to our cochlea images in terms of segmentation quality and speed.

Table 1: Chan-Vese Segmentation Method

Parameter	Value	Comment
Maximum number of iterations	100	
Step-size	1.0	
Stopping threshold	0.00001	
Internal energy weight	1.0	
External energy weight	1.0	

Table 2: Geodesic Active Contours (GAC)

Parameter	Value	Comment
Maximum number of iterations	500	
Step-size	1.0	
Stopping threshold	0.00001	
Internal energy weight	1.0	
External energy weight	1.0	

Table 3: Active Shapes Models (ASM)

Parameter	Value	Comment
Maximum number of iterations	40	
Step-size	1.0	
Number of resolutions	4	
Number of points	2	Number of interpolated contour points.
IP length	6	Length of landmark intensity profile
Search length	3	in pixels

In ASM, I only trained the model using CBCT shapes as it has higher resolution than the others. The ASM uses a multi-resolution approach which applies the same algorithm on different scales of the original image size defined by the number of resolutions parameter value. The result of each resolution is used in the next resolution as an initial location for the shape.

The charts in figure 4.1, figure 4.2 and figure 4.3 compare the results of the average Dice metric, Hausdorff distance and the speed of applying the previous algorithms on each modality, i.e. CBCT, CT and MRI. In the Dice metric, the higher the value, the better the result, whereas the smaller the value in Hausdorff distance and time, the better the result.

In CBCT group, the average Dice of Chan-Vese method was much higher 0.88 than GAC and ASM methods, 0.77 and 0.79 respectively. However, the Hausdorff of GAC was slightly lower than Chan-Vese. GAC required about double the time, 8.3 seconds, required by the other methods.

In CT group, the average Dice of the GAC method was higher, 0.92, than Chan-Vese and ASM methods, 0.84 and 0.78,. Hausdorff was also much lower, 2.53, than Chan-Vese and ASM, 4.42 and 4.14. GAC also required about double the time, 8.3 seconds, required by the other methods.

In MRI group, the average Dice of the Chan-Vese and GAC methods were much higher, 0.92 and 0.86, than ASM method, 0.21. It is clear that ASM produced bad results for the MRI images. This happens probably because ASM was trained only for CBCT images in this experiments which are similar to CT but not to MRI. In Hausdorff metric, Chan-Vese scored the best, 3.06, comparing to GAC and ASM, 4.41 and 5.42. ASM required more time, 5.4 seconds, than Chan-Vese and GAC, 3.1 and 4.4 seconds.

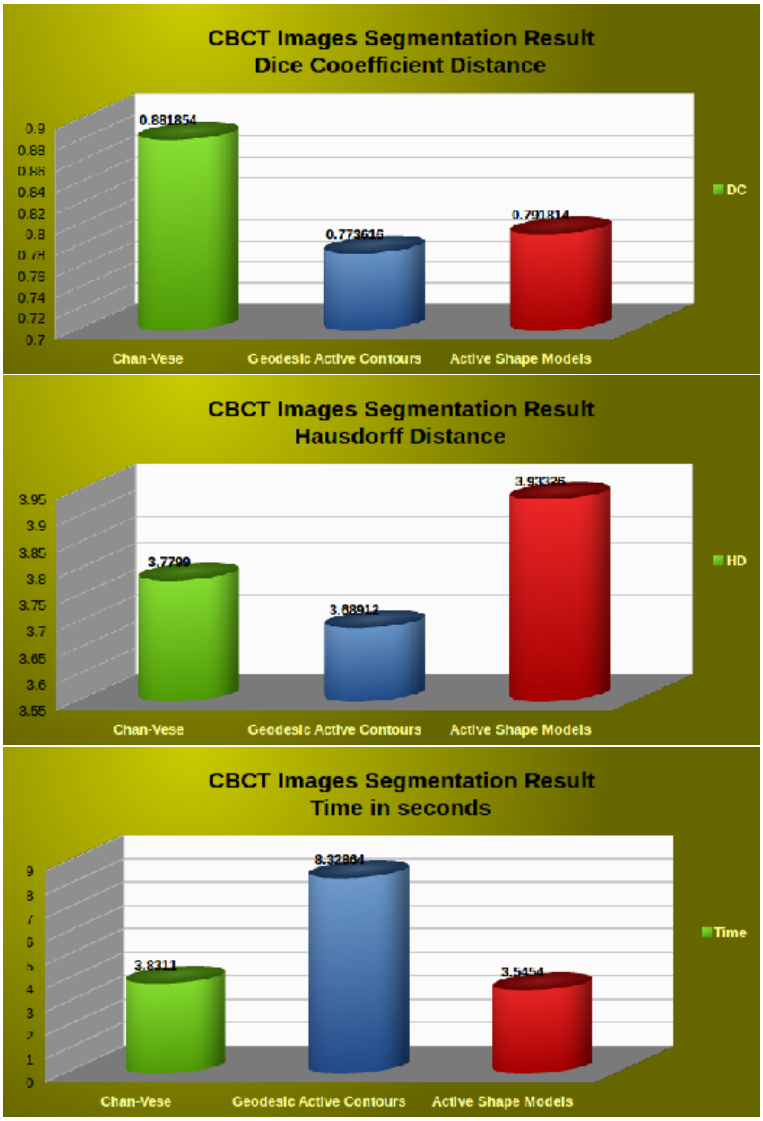


Figure 4.1: CBCT results, from top to bottom: Dice, Hausdorff and time.

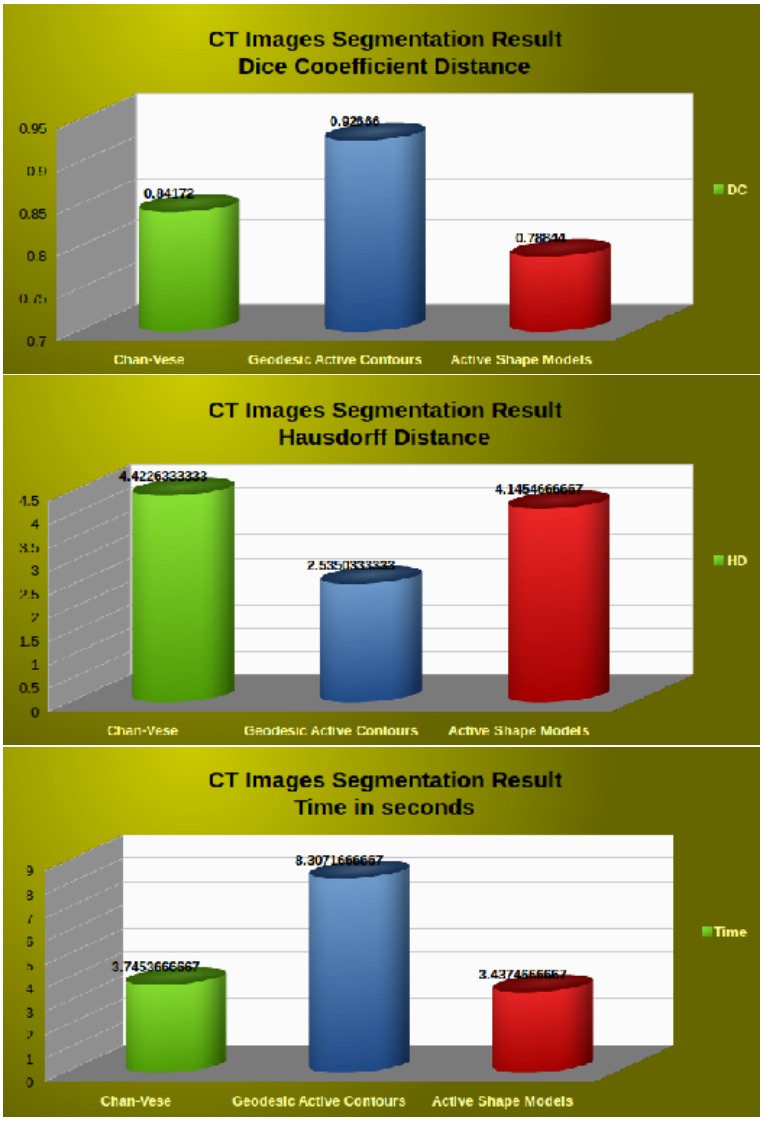


Figure 4.2: CT results, from top to bottom: Dice, Hausdorff and time.

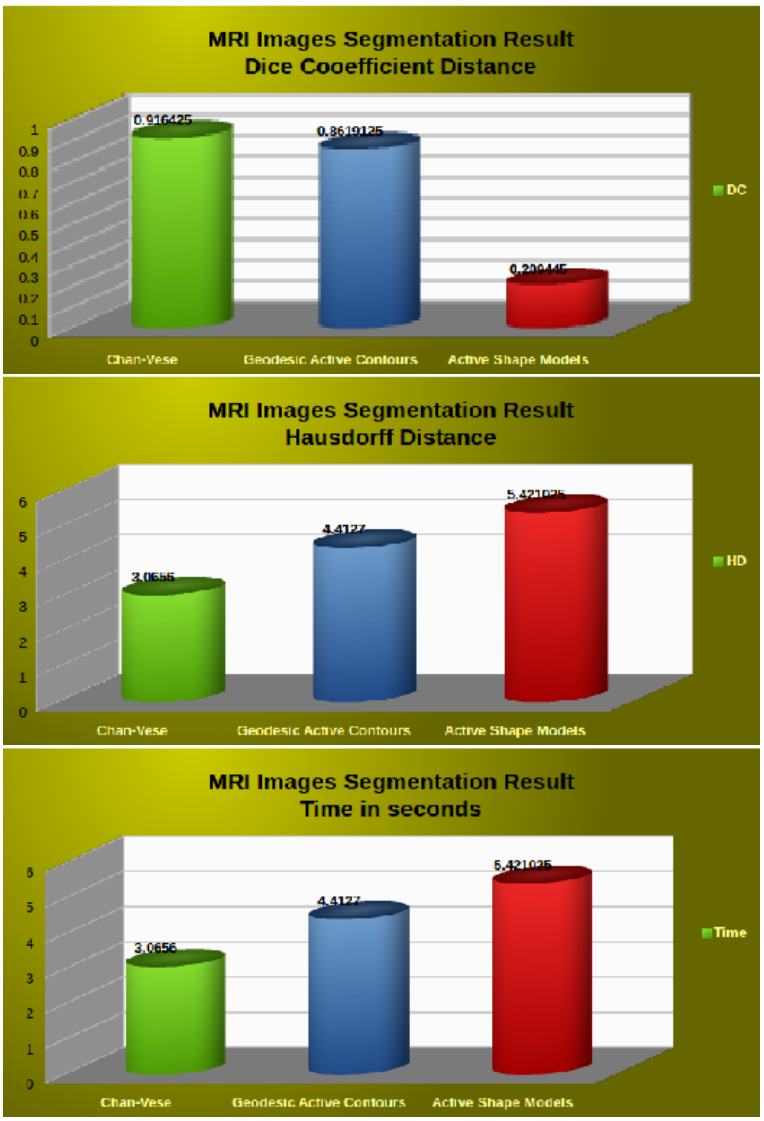


Figure 4.3: MRI results, from top to bottom: Dice, Hausdorff and time.

Visual samples of the coronal view from my experiment results are shown in figure 4.4, figure 4.5, and figure 4.6. They also show how the contours evolve during the time from the initial state until the boundary of the cochlea is found or the algorithm is failed.

Similarly, figure 4.7 shows visual samples of the coronal slice from the CT group. In the CBCT Images the results were better. The segmented cochlea by ASM provided false results even if the shape looks good. Figure 4.8 shows visual samples of the coronal slice from MRI group. In the CBCT and CT images, the results were better. Chan-Vese method provided much better segmentation while

the ASM produced a very bad segmentation.

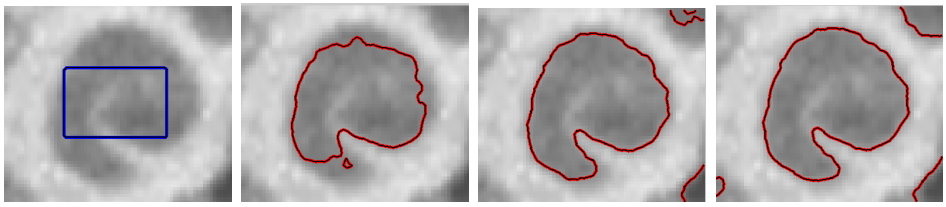


Figure 4.4: Chan-Vese method results, from left to right: initial contour, after 20 iterations, after 50 iterations, and the final result after 100 iterations.

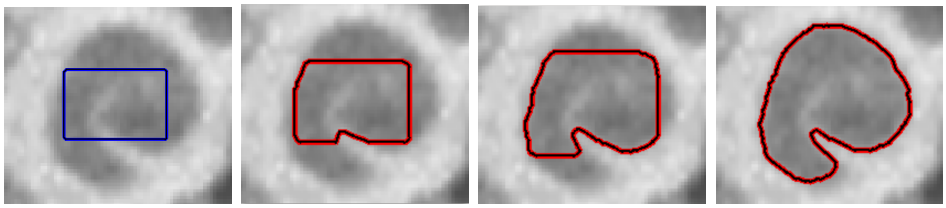


Figure 4.5: GAC method results, from left to right: initial contour, after 20 iterations, after 100 iterations and the final result after 300 iterations.

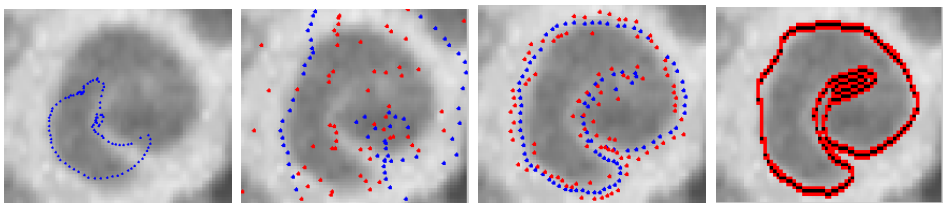


Figure 4.6: ASM method results, from left to right: initial shape, after 20 iterations in the fourth resolution, after 20 iterations in the first resolution and the final result after 40 iterations in the first resolution.

Figure 4.7: Sample of the results of the CT images, from left to right: Chan-Vese, GAC and ASM.

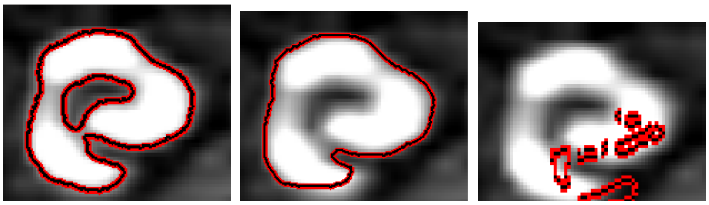


Figure 4.8: Sample of the results of the MRI images, from left to right: Chan-Vese, GAC and ASM.

As explained by Poznyakovskiy et al. [PZK+11], a multipart, model-based segmentation algorithm for the facial nerve and chorda tympani is proposed. Segmentation results are presented for ten test ears and compared to manually segmented surfaces. The results showed that the maximum error in the structure wall localization is 2 voxels for the facial nerve and the chorda. It is tested on only small and mono-modal dataset of 12 CT volumes. Moreover, the time reported the image segmenting was large, i.e. 5 minutes.

Noble et al [NLMD11] proposed using an active shape method where a shape model is created from μCT scans of ex vivo cochlea. The created model is used to segment conventional CT scans by fitting the model to clinical cochlea images. This helps to estimate the position of invisible cochlea structures in these images. Quantitative evaluation using Dice is used based on a set of μCTs. The results in Dice were 0.7 in average which is bad. Moreover, the time is not reported and the model is constructed from a small dataset of only 6 μCTs and tested on 5 of them.

A method for automatic segmentation of intra-cochlear anatomy in post implantation CT of unilateral recipients is proposed by Reda et al. [RML+14]. The method uses information extracted from the normal contralateral ear and exploits the intra-subject symmetry in the cochlear anatomy across the ears. It is validated on 30 ears for which both pre- and post-implantation CT images are available. The mean and the maximum segmentation errors were 0.224 and 0.734 mm, respectively. The method was not tested on other cochlea modalities and no time was reported.

A new framework for segmentation of μCT cochlear images is proposed by Pujadas et al. [PKV+16]. It uses a random walks algorithm where a region term is combined with a distance shape prior weighted by a confidence map. The map adjusts the influence of the distance shape prior according to the strength of the image contour. The region term takes advantage of the high contrast while the distance prior guides the segmentation to the exterior of the cochlea. A refinement is performed using a topological method and an error control map to prevent

leakage outside the boundary. The proposed framework is tested on only 10 μCT datasets. Moreover, it does not segment the cochlea structures, e.g. the scalae. The average Dice score was 0.94 with a standard deviation of 0.0082. No time is reported.

The literature shows major variations in the the measurements of human CDL. The main used methods were image processing and spiral coefficient equations.

As proposed by Escude et al. [EJD+06], an equation of a spiral coefficient is introduced which requires only one measurement called **A-value**. It is defined as a largest distance from the round window to the opposite cochlear lateral wall,i.e. the length of the line connects point x to point y in figure 4.9. This equation was improved by Alexiades et al. [ADJ15] and Koch et al. [KEZ+17] where a linear equation is used. The main disadvantage of the A-value based methods is that it needs a very clear image. It does not work in the case of the MRI images or the images affected by noise. It is also a manual procedure that requires time and effort of an expert.

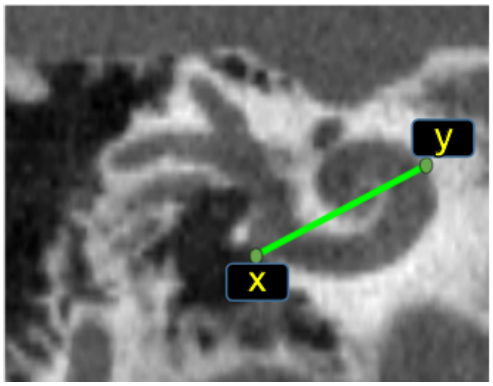


Figure 4.9: Cochlea A-value method. A-value is the distance between point x and y as it appears in CBCT data.

Iyaniwura et al. [IELA18] suggested a method to measure A-value automatically. It is tested using CT and μCT images of cadaveric cochlear specimens. They used the registration method provided in 3D Slicer software. An atlas with A-values is used as a moving image. They used 4 landmarks: cochlear apex, modiolus, round window and oval window. Then landmark registration was used as initialisation. Finally, they used cropping and two stages registration to align the atlas to the input image. An Affine image registration followed by B-spline. The Normalized Cross Correlation (NCC) was used as an image similarity metric.

They claimed, this method produces results similar to the manual method. No time is reported and the method is not tested against other image types, e.g. MRI.

As proposed by Weurfel et al. [WLLM14], a 3D reconstructions of the cross-sectional imaging is used to measure the cochlear length of temporal bones in the CBCT images. The cochlear length is measured from the distal bony rim of the round window until the helicotrema using a 3D curve set up from the outer edge of the bony cochlea.

Koch et al. [KEZ+17] computed the CDL and the relation between the basal turn lengths and CDL using the 3D multi-planar reconstructed CT images. They use 10 cadaveric temporal bones scanned by synchrotron radiation phase-contrast imaging. Points represent the organ of Corti, lateral wall and electrode location on the synchrotron radiation phase-contrast imaging slices along the length of the cochlea are calculated. Then the CDL is estimated from the points.

A comparison is made by Rivas et al. [RCH+17] between CDL computed from automatic A-value and 3D reconstruction to CDL from manual A-value and CDL. They mentioned that the automatic method result is reproducible and less time-consuming. The claimed that manual measurement of CDL by experts has high inter-observer variability, with mean absolute differences 1.15 mm. The range of CDL was from 29.54 mm to 37.66 mm.

Koch et al [KLEA17] reviewed the literature related to the CDL measurement. Part of this review can be found in table 4.4 and table 4.5. They analyzed results from different methods, e.g. direct measurement CDL from histology, reconstructing the shape of the cochlea, and determining CDL based on spiral coefficients. They concluded that the 3D reconstruction method is the most reliable method to measure CDL because it provides excellent visualization of the cochlea.

A deep learning was proposed by Moudgalya et al. [MCB20] for automatic segmentation to segment cochlear compartments from mice μCT volumes. They used a modified V-Net structure which is a convolutional neural network (CNN) architecture for 3D segmentation. They decreased the numbers of encoder and decoder blocks, and used dilated convolutions that enable extracting local estimates of drug concentration that are comparable to those extracted using atlas-based segmentation (3.37%, 4.81% and 19.65% average relative error in scala tympani, scala media and scala vestibuli), but much faster. They also tested the feasibility of training their network on a larger MRI dataset, and then by using transfer learning to perform segmentation on a smaller number of μCT volumes. This

Table 4.4: Survey of cochlea's length (in mm) literature.

idx	Ref	Modality	NumberOfImages	MinimumLength	MaximumLength	AverageLength
1	[Ret84]	Histology	5	32	34	33.5
2	[Har38]	Histology	68	25.6	35.46	31.6
3	[AP.85]	Histology	20	30.1	36.4	32.6
4	[PFS87]	Histology	9	24	33.5	28.4
5	[UVJ87]	Histology	50	28	40.1	34.2
6	[TS89]	Histology	1	NA	NA	36.4
7	[SST91]	Histology	18	32.7	43.2	38.6
8	[KSC96]	Histology	8	37.9	43.8	40.8
9	[KSC96]	Histology	8	34.15	37.9	35.6
10	[KSW$^+$98]	Spiral CT	20	29.5	37.5	33.01
11	[EJD$^+$06]	CT	42	30.8	37.4	34.4
12	[SSBL07]	Histology	9	30.5	36.87	33.11
13	[SLY$^+$09]	MR	6	17	26.5	NA
14	[CBOFO09]	CT MR	19	8.3	10.4	9.36
15	[ELRA11]	Plastic molds	325	38.6	45.6	42
16	[RALE$^+$12]	Plastic molds	51	37.6	44.9	41.2
17	[WLLM14]	CBCT	218	30.8	43.2	37.9
18	[JKB$^+$17]	CT	242	30.3	41.5	35.1
19	[RCH$^+$17]	CT	275	32.7	34.1	NA

Table 4.5: Survey of cochlea's size (in mm^3) literature.

idx	Ref.	Modality	NumberOfImages	Minimum Size	Maximum Size	Average Size
1	[Gal54]	NA	3	NA	NA	98.1
2	[Mag66]	Histology	NA	NA	NA	82
3	[IOI86]	MR	4	78.2	86.6	83.6
4	[HVW$^+$15]	MR	11	55	81	68

would enable the previous technique to be used in the future to characterize drug delivery in the cochlea of larger mammals. They didn't apply their method on the human clinical dataset as μCT segmentation is less challenging. They also didn't compare their method with the state of arts related methods, e.g. U-Net.

A quick survey from 23 papers within 233 years from 1884 to 2017 shows there is no agreement on the ranges of the size and the length of the cochlea in the literature, see table 4.4, and table 4.5.

Most of the previous researches measured the cochlea manually. Few researchers measured the cochlea automatically using proposed mathematical equations or 3D reconstruction.

In the previous studies, there are different modalities and a different number of patients. Some of them used high-quality histological images while others used CT, MRI or CBCT. There is no accurate information of the measurement start or end points. The ranges of the cochlea's length are from 8.3 mm to 45.6 mm and the cochlea's size range is from 55 mm^3 to 98.1 mm^3.

The histology image is a very high-resolution images generated from a microscope, and usually they are used as an atlas. The plastic mold means injecting the cochlea with a liquid or pliable material using a rigid frame called a mold or matrix. This allows the polymer to take the cochlea shape.

From the literature above, the development of a practical and fast method for multi-modal cochlea image registration, segmentation is needed. The method must be more consistent, and the results must be reproducible. In the next chapter, I will describe my proposed method that satisfies the above criteria.

Chapter 5

Materials and Methods

In the previous chapters, a brief introduction is given about the registration component used in this study. In this chapter, I will introduce my new proposed ideas for efficient automatic cochlea multi-modal image registration, fusion, segmentation, and analysis. I will also introduce a new public standard multi-modal 3D human cochlea images dataset that can be used for reproduction of my experiments and results or in future research.

5.1 Human Cochlea Datasets (HCD)

Different cochlea-related scientific research requires a standard public dataset. Unfortunately, such dataset does not exist. Most of the cochlea researchers used a private dataset which makes it difficult to reproduce and validate their results. A standard dataset HCD is proposed and described in [ADBP+17]. HCD contains 3D human cochlea images from three different modalities, i.e. MR, CT and CBCT of different patients.

About 150 anonymous datasets are collected from our partner hospitals in Germany and Egypt. The collected DICOM datasets are processed manually as follows:

- Assigning an ID to each DICOM dataset to be used as a reference. This allows getting more information in the future that are not available in the DICOM metadata, e.g. the type of the cochlea implant. Only the radiologist who provides the dataset can access the original patient information.

- Extracting 3D images with a high resolution by visualising all the images in the DICOM datasets, images with very low resolution are ignored.

- Converting DICOM format to a single compressed NRRD file format. This removes extra patient information if found[1].

- Renaming the NRRD files to represent the content of the image. This allows easy filtering and automating the process of these images during the experiments. The file name has the following format:

  P<Patient ID>_<Modality>_<Side>_<Implant>,

  where Patient ID is 6 digit integer number. Modality is two letters represent the image type, e.g. DV for CBCT, MR for MRI, and CT. The side identification is an optional one-letter "L" for left or "R" for right, and no letter means both ear sides are available in the image. One-letter "a" means after surgery and implant is in the image, and "b" means before the surgery and the image does not show an implant. If two or more files have the same patient ID, it means all images in these files belong to the same patient. Here are some naming examples:

  - P100003_DV_R_b: right side cochlea CBCT image before the implant surgery of patient 100003.
  - P100007_CT: both sides cochlea CT image before the implant surgery of patient 100007.
  - P100015_MR: both sides cochlea MRI image before the implant surgery of patient 100015.

- Locating two points landmarks per cochlea for all images. These points are located by experts at the round window and the cochlea apex. Each image file has a landmark file saved in 3D Slicer RAS orientation and .fcsv format.

The dataset includes 150 human cochlea 3D volumes of 67 patients of different gender and age. All images come from two hospitals of two different geographical locations, i.e. Germany (41 patients) and Egypt (26 patients). The dataset currently contains 87 CBCT, 38 MR and 22 CT cochlea images[2]

All the MRI images in HCD are T2 images that represent the patient status before CI surgery. Each image has a size of 384 voxels $\times$ 512 voxels $\times$ 64 voxels with 0.39 mm $\times$ 0.39 mm $\times$ 0.7 mm spacing. All belong to the German datasets group and obtained using "Siemens Skyra 3 Tesla" and "Siemens Avanto 1.5 Tesla" scanners.

[1]There were some cases that radiologist forgot to remove the patient's private data.

[2]Recall that some images have two ears while others not.

CBCT images from German datasets are obtained using "Morita 3D Accuitomo 170" scanner. These images represent one side only, either left ear or right ear. CBCT images are taken after CI surgery and they have a size of 485 voxels × 485 voxels × 121 voxels with 0.12 mm × 0.12 mm × 0.5 mm spacing. CBCT images are taken before CI surgery and they have a size of 483 voxels × 483 voxels × 161 voxels with 0.12 mm × 0.12 mm × 0.3 mm spacing. These CBCT images probably are cropped by the radiologist before handing them to me as they don't have isotropic spacing.

CT images from German datasets are obtained using "Siemens Sensation Cardiac 64" scanner. All CT images are taken before CI surgery, like MR, due to the high risk of doing this procedure after CI surgery. Each CT image has a size of 512 voxels × 512 voxels × 58 voxels with 0.12 mm × 0.12 mm × 1.0 mm spacing.

The CBCT images from Egyptian datasets were obtained using "i-CAT Next Generation" scanner with the following protocol 120 kvp, 5 mAs, voxel 0.2, matrix= 0.2.2.2 mm. FOV 2.5 cm, with 7 seconds exposure time and 14.7 seconds scanning time.

The CT images from Egyptian dataset are obtained using "Siemens Somatom Definition flash SD dual source 64 row" scanner with the following parameters KV, 100-140 mA 100-800, with pitch factor 0.55-1.5, ratio using one X-ray source and a single collimation width of 0.6 mm.

HCD is a proposed standard dataset for multi-modal 3D cochlea images. Many researchers in the field of CI can contribute to it and benefit from it. Hopefully, in the future, this dataset can be used in research and advanced deep learning techniques to analyse the cochlea images.

All patients' information is removed to protect their privacy. All images have two 3D landmark points, one for the round window and one for the cochlea apex (helicotrema), see figure 5.1. These landmarks are located by two experts and can be used for the results validation.

Samples from the dataset are already shown in figure 1.11 and figure 1.12. In the next chapter, I will show many samples from the dataset and their registration and segmentation results.

5.2 ACIR: Automatic Cochlea Image Registration

The objective of this study is to propose and to evaluate new, relatively easy and fast methodologies utilizing simple computer hardware and software to align and fuse cochlea images from different modalities.

In [ADBP+16][3], I proposed the multi-registration Automatic Cochlea Image

[3]Only poster is available.

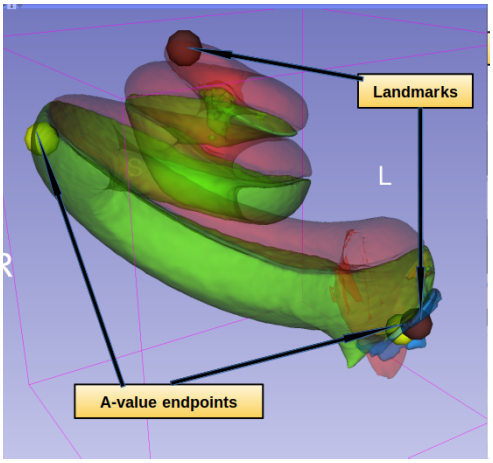


Figure 5.1: Cochlea landmarks in brown color and A-value endpoints in yellow color.

Registration (ACIR2016) which is a novel method that provides practical cochlea image registration. It combines different standard techniques and tunes parameters in a customized way that works for cochlea images. It combines different registration elements in a hierarchical approach of two stages: a rigid image registration followed by a non-rigid B-spline image registration. Moreover, the metric is based on the images types: S_{MSE} for mono-modal images (CBCT, CBCT) and S_{MMI} for multi-modal images,e.g. (CT, MR) or (CT, CBCT). Even though CBCT and CT are very similar, in practice, (CBCT-CT) image registration has better results when S_{MMI} is used. The method steps are listed in algorithm 1.

More details about important components of ACIR2016, e.g. S_{MMI} and ASGD, is explained in the next paragraphs.

Mattes' Mutual Information (S_{MMI}) [MHV+01] is a similarity metric used in ACIR2016. It is an improved mutual information metric that uses B-spline representation of the images. The image deformations is modelled using cubic B-splines. This make it possible to benefit from their computational efficiency, smoothness and local control[Uns99].

Using a sparse regular grid of control points over the volume, one can define a variable deformation by moving the control points. Cubic splines distribute this coarse grid over the entire image. Using B-spline to interpolate the deformation values between the control points produces a local and smooth deformation. It allows the simple derivative computation of both the image and the metric. The author's of [MHV+01] suggested the number of deformation coefficients to be around 100-1000 control-points.

The spacing of the grid defines the resolution of the deformation, e.g. for 3D

Algorithm 1: ACIR2016

Input: two cochlea images $I_F(P)$, and $I_M(P)$;
Output: a registered and fused image $I_{result}(P)$;
Set number of multi-resolution levels = 4;
Set optimiser = ASGD;
Set transform T = 3D rigid transform;
Set similarity metric = S_{MI} ;
if *same modality* **then**
    Set similarity metric = S_{MSE};
    μ = register(I_F, I_M) ;
    Set transform T = B-spline transform;
    μ = register($I_F(P)$, $(I_M(T(P, \mu)))$;
else
    $\tilde{I}_F$, $\tilde{I}_M$ = Crop(I_F,I_M);
    μ = register($\tilde{I}_F$, $\tilde{I}_M$) ;
    Set transform T = B-spline transform;
    μ = register($\tilde{I}_F(P)$,$\tilde{I}_M(T(P, \mu))$) ;
Transform the moving image $I_{result}(P) = I_M(T(P, \mu))$;
Fuse the result $I_{result} = I_{result} + I_F$

we have the vector $n_D = [n_x, n_y, n_z]^T$ that contains tuples of x,y and z coordinates of the control points.

In order to use S_{MMI} in the image registration minimization problem, one must compute the negative value of the metric and use its derivative with respect to the transformation parameters μ.

$$S_{MMI}(I_F(P), I_M(T(P, \mu))) = -\sum_{x \in I_F(P)} \sum_{y \in I_M(T(P,\mu))} \left(Pr(x, y) log_2 \left(\frac{Pr(x, y)}{Pr(x)Pr(y)} \right) \right) \tag{5.1}$$

where x and y are discrete sets of intensities associated to the fixed and transformed moving images, $Pr(x, y)$ is the joint probability, $Pr(x)$ and $Pr(y)$ are the marginal probabilities. The probability distributions in equation (5.1) are based on marginal and joint histograms of the two input images.

In order to form continuous estimates of image histograms, the Parzen windowing is used. This is useful also for reducing the effects of quantization and to make the joint distribution an explicitly differentiable function.

The Parzen window estimates a probability $Pr(x)$ of a sample x using N samples from the random variable X as follows:

$$\tilde{Pr}_N(x) = \frac{1}{hN} \sum_{i=1}^{N} w\left(\frac{x - x_i}{h}\right) \quad , \tag{5.2}$$

where:

- $w(\frac{x-x_i}{h})$, a Parzen window of width h, is a function with a unit integral e.g. $\int_{-\infty}^{\infty} w(x)dx = 1$,
- x_i is a set of N pixel values samples from a random variable X with probability $Pr(x)$,
- $h = \epsilon N$ is the Parzen window width, ϵ is a positive scaling factor that controls the width of the Parzen window w. When there are many samples available e.g. N is large, ϵ is small, and $w(\frac{x-x_i}{h})$ is a Dirac-like function. When N is small, ϵ is increased which increases the width of the Parzen window.

Note that $\tilde{Pr}(x)$ is large when the samples x_i are closed to x. When these samples are not dense around x, a few Parzen windows overlap and $\tilde{Pr}(x)$ is small.

The 2D joint discrete Parzen histogram is defined as:

$$h_j = \sum_{m \in L_M, f \in L_F} \frac{1}{\epsilon_m \epsilon_f} \sum_{x_i \in I_F} w\left(\frac{m}{\epsilon_M} - \frac{I_M(T(x_i, \mu))}{\epsilon_m}\right) . w\left(\frac{f}{\epsilon_f} - \frac{I_F(x_i)}{\epsilon_f}\right) \quad , \tag{5.3}$$

where

- h_j is the 2D joint histogram of all color pairs m, f.
- $I_M(x)$ is a moving image, I_F is a fixed image, x_i is a pixel location $\in I_F$,
- $T(x_i, \mu)$ is a transform with parameters μ, it outputs a new pixel location $\tilde{x}_i$.
- L_M, and L_F are pixel values samples from I_M and I_F, w is a Parzen window,
- m is a pixel value from L_M, and f is a pixel value from L_F,
- ϵ_m is related to $card(L_M)$, ϵ_f is related to $card(L_F)$.

Notice that $\sum h_j = card(V)$ is the number of pixels in one of the input samples. The contribution to the joint histogram of a single pair of pixels $I_M(x_i), I_F(x_i)$ is distributed over several discrete bins f and m by the window function w.

The joint histogram is proportional to the Parzen probability Pr:

$$Pr(f, m, \mu) = \alpha(\mu)h(f, m) \tag{5.4}$$

where α is a normalisation factor to avoid having $\sum w \neq 1$

$$\alpha(\mu) = \frac{1}{\sum_{m \in L_M} \sum_{f \in L_F} h(m, f, \mu)} \quad . \tag{5.5}$$

The joint histogram value means how many times the pixel value (or the colour) were found at the same locations, e.g. $h(m, f) = 22$ means when scanning image M and image F the pixel values m, and the pixel value f are found at the same locations 22 times.

In figure 5.2, an example from [TU00] is shown using only two colours m_0 and m_1 for image M, and two colors f_0 and f_1 for image F. Both images have size of $6 = 36$ pixels, so $card(V) = 36$ for this example. Image M is shifted by 1 pixel at each direction, so our transform T is a 2D translation transform with parameters $\mu = [1, 1]$. For simplicity, the Parzen window as a centered square pulse $w(m, f) = 1$ for $|m|, |f| < \frac{1}{2}$ and zero else where. By using $\epsilon_M = \epsilon_F = 1$, the histogram function is similar to the traditional one.

From figure 5.2, we have $Pr(m_0, f_0) = \frac{5}{36}$, $Pr(m_0, f_1) = \frac{4}{36}$, $Pr(m_1, f_0) = \frac{22}{36}$, $Pr(m_1, f_1) = \frac{5}{36}$, $Pr(m_0) = \frac{9}{36}$, $Pr(m_1) = \frac{27}{36}$, $Pr(f_0) = \frac{27}{36}$, and $Pr(f_1) = \frac{9}{36}$.

Using equation (5.1) and equation (5.4):

$$S_{MMI}(M, F) = -\sum_{m \in M} \sum_{f \in F} \left(Pr(m, f) log_2 \left(\frac{Pr(m, f)}{Pr(m)Pr(f)} \right) \right) \tag{5.6}$$

$$= -\frac{5}{36} log_2 \left(\frac{(5)(36)}{(9)(27)} \right) - \frac{4}{36} log_2 \left(\frac{(4)(36)}{(9)(9)} \right) - \frac{22}{36} log_2 \left(\frac{(22)(36)}{(27)(27)} \right) - \frac{5}{36} log_2 \left(\frac{(5)(36)}{(27)(9)} \right)$$

which is equal -0.045. When the images are aligned correctly the negative mutual information will be at minimum, e.g. $S_{MMI} = -0.865$ for the above example.

The derivative of the Mutual Information defined in equation (5.1) is:

$$\frac{\partial S_{MMI}(I_M(T(P, \mu)), I_F(P))}{\partial \mu} = -\sum_{m \in L_M} \sum_{f \in L_F} \frac{(m, f)}{\partial \mu} log_2 \left(\frac{Pr(m, f)}{Pr(m)Pr(f)} \right) \quad , \tag{5.7}$$

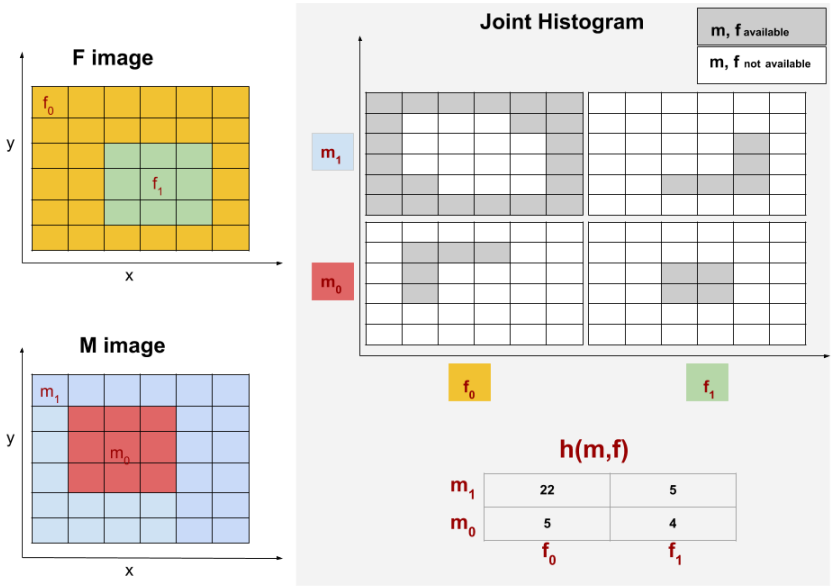


Figure 5.2: Example of joint histogram of two images from different color distributions.

where L_M, and L_F are pixel values samples from the fixed image and the transformed moving image at the same locations. This equation requires solving the derivative $\frac{\partial Pr(m,f)}{\partial \mu}$. This can be done using n degree B-spline, e.g. $\beta^n(x)$ [4] [TU00].

$$\frac{\partial Pr(m,f)}{\partial \mu} = \frac{1}{c(V)} \sum_{x_i \in V} \beta^n \left(\frac{f}{\epsilon_F} - \frac{I_F(x_i)}{\epsilon_F} \right) \frac{\partial \beta^n(\zeta)}{\partial \zeta} \frac{1}{\epsilon_M} \left(\frac{-\partial I_M(T(x_i,\mu))}{\partial x} \right) \frac{\partial T(x_i,\mu)}{\partial \mu}, \tag{5.8}$$

where $c(V)$ is the size of the sample V, $\zeta = \frac{m}{\epsilon_M} - \frac{I_M(T(x_i,\mu))}{\epsilon_M}$. There are two more derivatives yet to be solved, the B-spline function $\frac{\partial \beta^n(\zeta)}{\partial \zeta}$ and the transformed moving image $\frac{\partial I_M(T(x_i,\mu)}{\partial x}$:

$$\frac{\partial \beta^n(\zeta)}{\partial \zeta} = \beta^{n-1}(\zeta + \frac{1}{2}) - \beta^{n-1}(\zeta - \frac{1}{2}) \quad . \tag{5.9}$$

$$\frac{\partial I_M(T(x_i,\mu)}{\partial x} = \sum_{x_i \in V} c(x_i) \frac{\partial \beta^n(T(x_i, mu) - x_i)}{\partial x} \quad . \tag{5.10}$$

The marginal joint density derivative is:

[4] Remember that n in β^n does not mean "to the power n", it means the degree of the B-spline basis function.

$$\frac{\partial Pr(m)}{\partial \mu} = \sum_{f \in L_F} \frac{\partial Pr(m,f)}{\partial \mu} = \frac{1}{card(V)} \sum_{x_i \in V} \frac{\partial \beta^n(\zeta)}{\partial \zeta} \frac{1}{\epsilon_M} \frac{-\partial I_M(T(x_i,\mu))}{\partial x} \frac{\partial T(x_i,\mu)}{\partial \mu} \quad . \tag{5.11}$$

ACIR2016 uses ASGD optimiser which is based on Stochastic Gradient Method, RM-SGD, proposed in [RM51] and the adaptive step-size proposed in [PC04]. The update rule for RM-SGD is defined in equation (5.12).

$$\mu_{k+1} = \mu_k - \lambda_k \frac{\partial S(\tilde{I}_F(P), \tilde{I}_M(T(P,\mu)))}{\partial \mu} \quad , \tag{5.12}$$

where the approximation of the cost function derivative is used and λ_k is defined as:

$$\lambda_k = \frac{a}{(k+A)^\alpha} \quad , \tag{5.13}$$

where k is the current optimisation iteration, $a > 0$, $A \geq 1$ and $0 < \alpha \leq 1$. For large k, $\alpha = 1$ is a good choice. Setting $\alpha < 1$ causes the step-size to decay slower. The factor a has no unit, and depends on the cost function. If a is set too small, the optimisation is slow. However, when a is set too large, the optimisation may become unstable.

ASGD [KPSV09] adapt the step-size λ_k based on the inner product of the current approximation of the cost function derivative $\frac{\partial S_k}{\partial \mu_k}$ and the previous one $\frac{\partial S_{k-1}}{\partial \mu_{k-1}}$. A larger steps can be taken if the approximated derivatives in two consecutive iterations point in (almost) the same direction and the step size is reduced otherwise.

In ASGD, λ_k is defined as a function of t_k:

$$\lambda_k = \lambda(t_k) = \frac{a}{t_k + A} \quad , \tag{5.14}$$

where

$$t_{k+1} = max(0, [t_k + f(-\tilde{g}_k^T \tilde{g}_{k-1})]) \quad , \tag{5.15}$$

where $\tilde{g}_k = \frac{\partial S(\tilde{I}_F(P), \tilde{I}_M(T(P,\mu_k)))}{\partial \mu}$, and f is a sigmoid function defined in equation (5.16). The initial values of μ_0, t_0, and t_1 are defined by the user before the optimisation. The computation starts from t_2 as we need two cost function derivatives, i.e. $\tilde{g}_0$, and $\tilde{g}_1$. When the inner product of $\tilde{g}_{k-1}$ and $\tilde{g}_k$ are positive, the

time t is reduced and we get a large step-size λ. When $f = 1$ we get the original RM-SGD method.

$$f(x) = f_{MIN} + \frac{f_{MAX} - f_{MIN}}{1 - \frac{f_{MAX}}{f_{MIN}} e^{-x/\omega}} \quad , \tag{5.16}$$

where $f_{MAX} > 0$, $f_{MIN} < 0$ and $\omega > 0$. When ω approaches zero, f approaches a step function.

ASGD Algorithm steps:

1. Set the required user-defined variables:
   - Set the number of iterations K, e.g. $k = 300$.
   - Set the initial conditions t_0, t_1, e.g. $t < tK$.
   - Set $0 < \eta \leq 1$.
   - set δ = voxel size in mm, e.g. 1.0 mm, $N = 10$, $A = 20$, $\zeta = 0.1$.
2. Compute S using:
   $$S = \frac{1}{|P|} \sum_{p_i \in P} J_\mu^T J_\mu \quad , \tag{5.17}$$
   where P is the points samples from the image, J_{μ_i} is the Jacobian $J_\mu = \frac{\partial T(p,\mu)}{\partial \mu}$, the transformation derivative with respect to the transformation parameters which is defined in equation (2.41).
3. Compute σ_4 using:
   $$\sigma_4^2 = min_{p_j \in P} \frac{\delta^2}{||J_{\mu_j}||_F^2 + 2\sqrt{2}||J_{\mu_j} J_{\mu_j}^T||_F} \quad , \tag{5.18}$$
   where $||J||_F = \sqrt{\sum_{x \in J} |x|^2}$ is the Frobenius norm [GV13] of the Jacobian.
4. Generate N=10 random instances of μ using:
   $$\mu \sim \mathcal{N}(\mu_0, \sigma_4^2 I) \quad , \tag{5.19}$$

   $$\mu = \frac{1}{\sqrt{(2\pi)|\Sigma|}} e^{\frac{-1}{2}(x-\mu_0)^T \frac{1}{\Sigma}(x-\mu_0)} \quad , \tag{5.20}$$
   where x is sequence of values in the parameters' domain, Σ is the covariance matrix of $\sigma_4^2 I$, and $|\Sigma|$ is the determinant of Σ.

5. Compute σ_1, σ_2 and σ_3 using:

$$\sigma^2 = \frac{\frac{1}{N}\sum_{n=1}^{N}||g||^2}{tr(S)} \quad , \tag{5.21}$$

where tr(S) is the trace of the matrix [GV13] which is defined as the sum of the main diagonal elements of the square matrix, and g is the exact metric derivative.

6. Compute the exact gradient using:

$$\begin{aligned} g_i &= \frac{\partial S(I_F(P), I_M(T(P, \mu_i)))}{\partial \mu_i}, \quad i \in 1, 2, .., N. \\ g_k &\sim \mathcal{N}(0, \sigma_1^2 S) \quad . \end{aligned} \tag{5.22}$$

7. Compute the approximated gradient using:

$$\begin{aligned} \tilde{g}_i &= \frac{\partial S(\tilde{I}_F(P_i), \tilde{I}_M(T(P_i, \mu_i)))}{\partial \mu_i}, \quad i \in 1, 2, .., N. \\ \tilde{g}_k &\sim \mathcal{N}(0, \sigma_2^2 S) \quad . \end{aligned} \tag{5.23}$$

Note that a new voxel set P_i is selected each time.

8. Compute the difference between the approximated gradient and the exact gradient using:

$$\begin{aligned} \epsilon_i &= g_i - \tilde{g}_i, \quad i \in 1, 2, .., N. \\ \epsilon &\sim \mathcal{N}(0, \sigma_3^2 S) \quad . \end{aligned} \tag{5.24}$$

9. Compute a_{MAX} using:

$$a_{MAX} \approx \frac{2A}{\lambda} = \frac{A\delta}{\sigma_1} min_{_{p_j \in I_F(\tilde{P})}} \frac{1}{\sqrt{tr(J_j S J_j^T) + 2\sqrt{2}||J_j S J_j^T||_F}} \quad . \tag{5.25}$$

10. Compute η and a using:

$$a = a_{MAX} = \frac{E||g||^2}{E||g||^2 + E||\epsilon_k||^2} \equiv (a_{MAX})(\eta) \quad , \tag{5.26}$$

where $0 < \eta \leq 1$. For $E||g||^2$ and $E||\epsilon||^2$ their empirical estimates can be used directly. The original parameter a is replaced by a new user-defined parameter, δ that has a unit (mm) and an intuitive meaning.

11. Compute f_{MIN}, using:

$$f_{MIN} = \eta - f_{MAX} = \eta - 1 \quad , \tag{5.27}$$

where $0 < \eta \leq 1$ and $f_{MAX} = 1$.

12. Compute ω using:

$$\omega = \zeta \sqrt{Var(\epsilon_k^T \epsilon_{k-1})} \quad , \tag{5.28}$$

where $\zeta = 0.1$. $Var(\epsilon_k^T \epsilon_{k-1}) = \sigma_3^4 ||S||_F^2$.

13. Start optimisation and stop after k iterations. Note that the steps 1-12 are all done before the optimisation one time only.

ASGD method requires the computation of stochastic approximations of the cost function derivatives at each iteration. It takes into consideration the free choice of the similarity measure, the transformation model, and the image content. I used ASGD in my registration methods as it was already shown it works with other medical images problems successfully. It is also already implemented in Elastix tool as open-source.

The goal is to register all three modalities, i.e CT, MR and CBCT, to the same physical space. For having this done, at first CBCT to CT are aligned, then MR to CT are aligned. All three modalities can be fused to one image that represents information from CT,CBCT and MRI. Registering from image modality to another sometimes works only in one direction, e.g. registering CT to MR does not work but MR to CT works. To solve this issue, the inverse transform of the working registration situation is used in some cases. ACIR2016 has problems with some cochlea images, a major issue is the registration of (CBCT-MR) when a patient has no CT image. Moreover, it fails to provide accurate registration in some cases. Finally, it takes a longer time than other methods to complete the registration of two mono-modal images, e.g. (CBCT-CBCT). This happens due to the expensive multi-registration and B-spline transformation approach. However, this time is still much better than the time in the other existing methods at that time.

The cochlea image registration problem in our case is intra-subject image registration. Hence, it belongs to rigid image registration problems. For such images, all pixels in the image is affected by the transform. Finding the correct transform for any part of the moving image may solve the problem for other parts in the image.

However, the image registration result is highly dependent on image contents so finding a method that works on cochlea images is a challenging task. Some areas in these images have clear structures and less noise. Cropping the original images to

one of these small areas and registering them is a faster and better way to produce a transform that registers the original images. Benefiting from prior knowledge of the images' characteristics, I proposed an improved Automatic Cochlea Image Registration method, ACIR2017 [ADBP$^+$17]. This method uses ASGD optimiser to minimise negative S_{MMI} similarity metric of a cropped version of the two input images by modifying 3D rigid transformation parameters, see algorithm 2.

Algorithm 2: ACIR2017

1 **Input:** two cochlea images $I_F(P)$, and $I_M(P)$;
2 **Output:** a registered and fused image $I_{result}(P)$;
3 Locate the cochlea locations in input images;
4 $\tilde{I}_F$, $\tilde{I}_M$ = Crop(I_F,I_M);
5 Set transform T = 3D rigid transform;
6 Set $k = 0$.;
7 **While** -$S_{MMI}(\tilde{I}_F(P), \tilde{I}_M(T(P,\mu))$ is large and $k < 100$ **do**;
8 μ = update the old μ using ASGD ;
9 Set $k = k + 1$;
10 Transform the moving image, $I_{result}(P) = I_M(T(P,\mu))$;
11 Fuse the result, $I_{result} = I_{result} + I_F$

ACIR2017 is less complicated than ACIR2016. Neither multi-resolution nor B-spline transformation is involved. The cropping part allows the optimisation process to complete faster as it involves small size images. The localization in step 3 is the only manual step in this method, the registration process is fully automated as the localization step is used only for cropping in step 4. Optimisation finds the 6 parameters of the 3D rigid transform, i.e. the Euler angle of rotation for each axis r_x, r_y, r_z, in addition to translations in each dimension t_x, t_y and t_z in a few seconds.

ACIR2017 is more flexible than the ROI methods because it takes any clear part of the input images, not just the cochlea part ROIs. It has the advantage in comparison to the masking techniques because it does not consider the non-mask pixels in the original images during the optimisation.

The cost function is computed based on the cropped cochlea areas only, which makes it faster and more stable as there is always a voxel mapping available during the optimisation.

ACIR2016 and ACIR2017 are gradient descent methods. As mentioned in the previous chapters, GD methods do not perform well when the Hessian is ill-conditioned. Based on the Stochastic Quasi-Newton with Limited memory Broy-

den Fletcher Goldfarb Shanno (S-LBFGS) [SW17] updating rule, I proposed another Automatic Cochlea Image Registration (ACIR2020) method.

The proposed method is inspired by ACIR2017 and S-LBFGS [QSLS15]. I proposed replacing the ASGD optimiser in ACIR2017 by the S-LBFGS optimiser. This way, I took advantages of both the ACIR2017 and S-LBFGS methods. This may allow faster convergence and more robust results.

Strict implementation of Newton's method involves finding the inverse of the Hessian matrix at every iteration. If the number of variables on which f depends on is large then this is prohibitively expensive. In the case of the cost function of a 512×512 image, the number of variables will be $n = 2^{18}$.

This is where the quasi-Newton methods come into picture. In these methods, an approximation of the actual Hessian matrix is used. There is a large number of choices of the approximation form of the Hessian matrix, each one gives a rise to a variant of quasi-Newton method. One of the most popular variations is (BFGS) [NW06]. If $H_k = \boldsymbol{H}_k^{-1}$ then the BFGS approximation is defined as:

$$H_{k+1} = (I - ((y_k^T d\mu)^{-1}) y_k d\mu^T)^T . H_k . (I - ((y_k^T d\mu)^{-1}) y_k d\mu^T) + ((y_k^T d\mu)^{-1}) d\mu d\mu^T \tag{5.29}$$

where $d\mu = \mu_{k+1} - \mu_k$.

$$H_{k+1} = V_k^T H_k V_k + \rho_k s_k s_k^T, \tag{5.30}$$

where $\rho_k = (y_k^T s_k)^{-1}, V_k = I - \rho_k y_k s_k^T, s_k = \mu_{k+1} - \mu_k, y_k = g_{k+1} - g_k$ and g_k being the same as $\nabla f(x_k)$.

If f is a function of n variables then the Hessian matrix and its approximation have size $n \times n$.

For a large n storing the Hessian itself will take up a lot of memory. The LBFGS [SW17] algorithm approximates the Hessian further by storing only a few vectors that can reconstruct the Hessian. Thus, the storage demand drops from $O(n^2)$ to $O(n)$.

If $g_k = \nabla f(x_k)$ and $H_k = \boldsymbol{H}_k^{-1}$, then one can write:

$$x_{k+1} = x_k - \lambda H_k g_k, \tag{5.31}$$

It is observed that storing a $n \times n$ matrix H_k is not needed if there is a way to estimate the n-vector $H_k g_k$. One way to estimate the product $H_k g_k$ is the two-step recursion explained in LBFGS.

The previous equations suggest that all the quantities can be readily computed from the curvature points $\{s_k, y_k\}$. Computing these at each iteration involves all points of the image. One way to reduce this effort is to work with only a subset of these. This is similar to how the stochastic gradient descent algorithm works.

However, the stochastic gradient descent algorithm is inherently noisy and the problem is exacerbated in the case of the medical images by the fact that each computation of a stochastic gradient potential involves a different set of voxels.

To solve this issue one can de-noise the optimization parameters $\boldsymbol{\mu}$ by averaging them after every L iterations and use a small random subset $\mathcal{S}_1$ in each of these iterations. After that, the average is used to compute the full Hessian using a random subset $\mathcal{S}_2$ of the image.

This can be adapted to medical images by using more samples in the L iterations used for updating $\{s_k, y_k\}$ and computing the difference in gradients instead of the Hessian. Thus,

$$y_k = g(\bar{\boldsymbol{\mu}}_I; \mathcal{S}_2) - g(\bar{\boldsymbol{\mu}}_J; \mathcal{S}_2) \tag{5.32}$$

Note that the same random subset $\mathcal{S}_2$ of the image is used for the computation of both gradients.

The opitmiser S-LBFGS uses an adaptive step-size estimation based on the image information. It uses the idea proposed by [BHNS16] where the authors suggest eliminating the stochastic gradient noise by averaging the optimisation parameters each L iteration and calculating H on a random subset of the data. This allows for a much more efficient way of obtaining the approximation of H and stochastic gradients.

The authors of [QSLS15] also used a restarting mechanism to recompute the step-size for every L iteration and used the gradient difference to estimate H. Using S-LBFGS optimiser in ACA2020 solves the noise problem as the original image is cropped and most of the noise should be removed. My proposed method steps are listed in algorithm 3.

Algorithm 3: ACIR2020

Input: two cochlea images $I_F(P)$, and $I_M(P)$;
Output: a registered and fused image $I_{result}(P)$ Locate the cochlea locations in input images;
$\tilde{I}_F$, $\tilde{I}_M$ = Crop(I_F,I_M);
Set transform T = 3D rigid transform;
Set $k = 0$.;
While -$S_{MMI}(\tilde{I}_F(P), \tilde{I}_M(T(P, \mu))$ is large and $k < 100$ **do**;
    μ = update the old μ using S-LBFGS ;
    Set $k = k + 1$;
Transform the moving image, $I_{result}(P) = I_M(T(P, \mu))$;
Fuse the result, $I_{result} = I_{result} + I_F$

The only difference between ACIR2017 and ACIR2020 is the optimisation method. ACIR2020 uses S-LBFGS which is a Stochastic quasi-Newton method based on LBFGS. It uses the adaptive step-size estimation based on the image information. The S-LBFGS uses the idea proposed by Byrd et al [BHNS16] where the authors suggested eliminating the stochastic gradient noise by averaging the optimisation parameters for each L iteration and calculating H on a random subset of the data.

This allows for a much more efficient way of obtaining an approximation of H and stochastic gradients. Using the S-LBFGS optimiser should provide more enhancement in terms of time and accuracy.

5.3 ACA: Automatic Cochlea Analysis

I proposed a novel method, ACA2018 [ADBP$^+$18b], for the automatic cochlea image segmentation and analysis. It uses an atlas-model-based segmentation to align a predefined segmentation and point models to an input image, see figure 5.3. The predefined segmentation serves as an atlas. The atlas was manually segmented using a high resolution μCT image obtained from a public and standard μCT cochlear dataset [GRB$^+$17].

The original μCT was too large to be processed in a standard computer so it is resampled from 0.008 $mm\times$ 0.008 $mm\times$ 0.008 mm spacing to 0.032 $mm\times$ 0.032 $mm\times$ 0.008 mm spacing. This reduced the image size from 13.4 GB to 806 MB. ITK and 3D Slicer software was used for this resampling. Next, the image was cropped to the cochlear part only. This allows for a smaller image size of 103.2 MB with 243 voxels $\times$ 202 voxels $\times$ 1191 voxels[5] instead of 437 voxels $times$ 412 voxels $times$ 2349 voxels. After that, the two main cochlear scalae, i.e. scala tympani and a combination of both scala media and scala vestibuli were segmented manually[6], see figure 5.4 left.

The model and its segmentation were cloned and transformed manually to represent left and right cochlear sides. The transformed models were automatically aligned to one of the clinical CBCT images using ACIR2017 [ADBP$^+$17]. The atlases were aligned the same way. A user-friendly interface for the atlas-based segmentation method was developed as a Slicer plug-in [6, 28].

Getting the size of the cochlea scala automatically from the segmentation[7] is done by counting the number of voxels belonging to this segment area then multiplying the resulted number by the voxel size in mm^3:

[5]Voxel: a volume element

[6]Even with this high-resolution image it was difficult to see the separation between scala media and scala vestibule.

[7]Tip: in 3D Slicer, the "Segment Statistics" module can be used for this purpose.

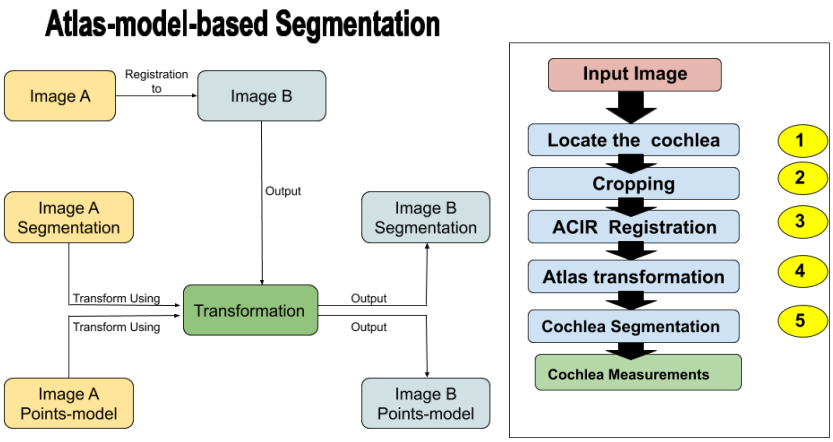


Figure 5.3: Left: atlas-model-based concept. Right: cochlea analysis pipeline

$$V = \prod_{i \in d} S_i \quad , \tag{5.33}$$

where V is the volume size in mm^d, S is the spacing in mm, d is the image dimensions, e.g. 3 for 3D.

To get the length of a cochlea scala, usually, the Skeletonization method is needed [GW06]. The skeleton is an image processing thinning process that produces a line or curve from an object. This makes it easy to measure the length of a curve by measuring the distance of each two consecutive points. Unfortunately, this arises a new challenge as the standard skeleton methods did not work due to the non-standard shape of the scalae.

Even for a simpler problem i.e. getting the length of the cochlea implant, skeleton does not work because sometimes there is a gap between the segments or there is a large noise, see figure 5.4 left and figure 5.5. Notice, in figure 5.4 that the first electrode location is more deeper inside the scala tympani than it should be, this shows how these images are misleading sometimes.

In figure 6.5, the first image in the right side was segmented using a threshold of [2173-32446], notice that there is a small jab in the electrodes curve. The middle image was segmented using a threshold of [1800-32446], notice how the jab is closed but the curve still has irregular shape. If the threshold range is larger, more noise is introduced in the image as in the image in the right where the threshold was [1500-32446].

To solve this issue, I proposed using a points-set mode. It contains 55 points representing the center of the scala tympani, see figure 5.6. Using the transformed

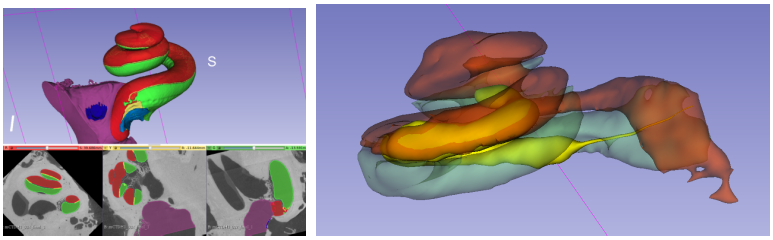


Figure 5.4: Left: Cochlea atlas is generated from high resolution μCT image, scala tympani is in green color and scala media and scala vestibule are combined in red color. Right: CBCT CI visualization example, scala tympani (green), scala vestibuli (red) and CI (yellow).

Figure 5.5: Segmentation of CI with different thresholds, large threshold range produces more noise.

points-set, the length of the scala tympani can be calculated by computing the distance between each two consecutive points by equation (5.34).

$$L_{ST} = \sum_{i=1}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2} \tag{5.34}$$

where L_{ST} is the length of the scala tympani in mm, n is the number of the points, x, y and z are the 3D point coordinates. Using this approach has two benefits, first, it is faster than the skeleton as it includes only a matrix multiplication. Second, the points can be corrected or modified later using a friendly user-interface I developed as a plug-in for 3D Slicer tool [6, 28], e.g. to produce different useful measurements like measuring the inner length or the outer length of a scala, see figure 5.7.

A user friendly interface is developed as a 3D Slicer plugin for the cochlea analysis[8], see figure 5.7. The user only inputs the cochlea location point. After that, everything is done automatically.

The algorithm finds the segmentation, generates a 3D cochlea detailed model then computes the size and the length of the scala tympani. Since the plugin is based on 3D Slicer, it runs in different operating systems and devices.

[8]https://github.com/MedicalImageAnalysisTutorials/SlicerCochlea

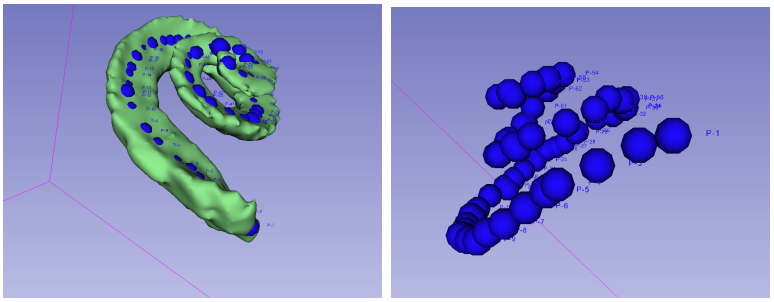


Figure 5.6: Different visualizations of the scala tympani points sets.

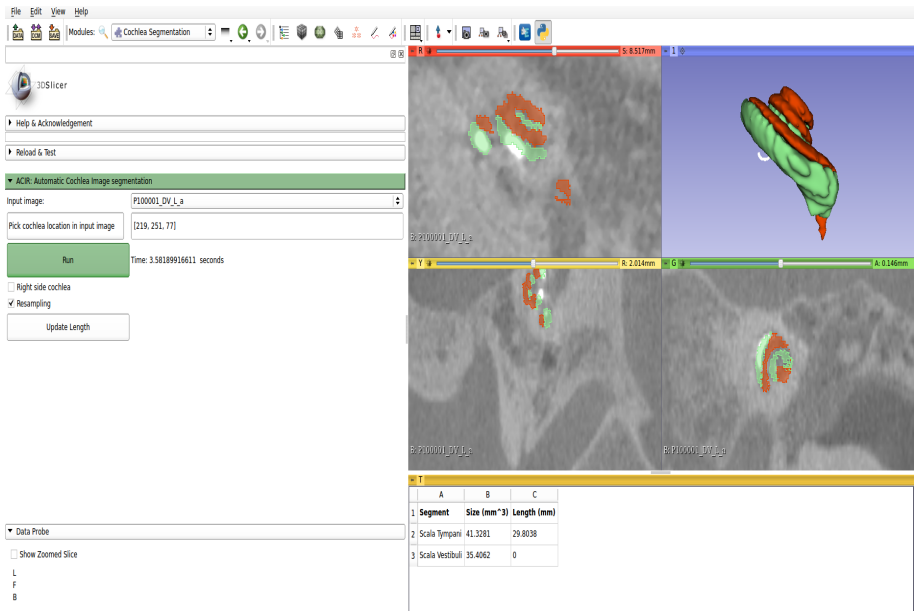


Figure 5.7: My 3D slicer cochlea plugin's user friendly interface.

ACA2018 aligns an image with a pre-defined detailed segmentation and a pre-defined points model to the input image. After that, it uses the resulted transformation to transform the pre-defined segmentation and the points model to the input image, see figure 5.3. In this method, there was only one points-model for estimating the length of the scala tympani which is of more interesting due to its rule in CI surgery. The method uses Elastix tool which has a problem with points transformation. To solve this issue, the point-model is transformed to the image. Then the transformation found by ACIR2017 is applied to the points-image, and finally the points are extracted from the transformed points-image. Due to this transformation, some points are missing after the transformation. These missing points resulted from the interpolator used with Elastix tool. One may try using Nearest Neighbour (NN) interpolator but I found out that while using 3D Slicer,

Algorithm 4: ACA2018

Input: cochlea image I(P) ;
Output: cochlea segmentation and different measurements ;
Locate the cochlea locations in input images;
$\tilde{I}(P)$= Crop($I(P)$);
Apply ACIR2017 on the cropped input image and the predefined image ;
    $I_{res}, T = (\tilde{I}(P), I_M(P))$;
Transform segmentation and points-models using T and its parameters μ;
    $I_{seg}(P) = I_{M_{seg}}(T(P), \mu)$;
    $I_{points}(P) = I_{M_{points}}(T(P), \mu)$;
Output the segmentation, $I_{seg}(P)$, the segmentation sizes, and length of points model $I_{points}(P)$

I can apply the transform directly to the points.

The image registration part of ACA2018 is based on ACIR2017 which produces a rigid transformation. This rigid transformation does not handle the difference in shape and size among different patients. I proposed an improved method, ACA2020, by adding a second stage non-rigid registration that uses B-spline transformation. This addresses the difference between the input images of different patients and the model. Using the non-rigid registration gives more accurate deformation and more realistic measurements. The steps of the method are listed in algorithm 5.

Furthermore, ACA2020 uses the stochastic quasi-Newton optimiser proposed by Qiao et al. [QSLS15] instead of ASGD. This handles the situation when a Hessian matrix is ill-conditioned as the Newton optimiser method takes in consideration the second-order partial derivatives of the cost function. Moreover, a higher-resolution cochlea atlas is used which produces more accurate results. The figure 5.8 shows how the proposed method result looks like comparing to ACA2018 [ADBP$^+$18b].

Finally, the previous points-model is enhanced to have 75 instead of 55 points and I added three more points-models to automate the process of measuring different lengths. Currently, ACA2020 includes these point models:, and figure 5.3:

- Scala vestibuli points model (SvPt): a set of 74 points that passes through the center of the scala vestibule, see figure 5.9, points are in magenta color.
- Scala tympani points model (StPt): a set of 75 points that passes through the center of the scala tympani, see figure 5.9, points are in yellow color.
- Scala tympani lateral points model (StLt): a set of 118 points that passes

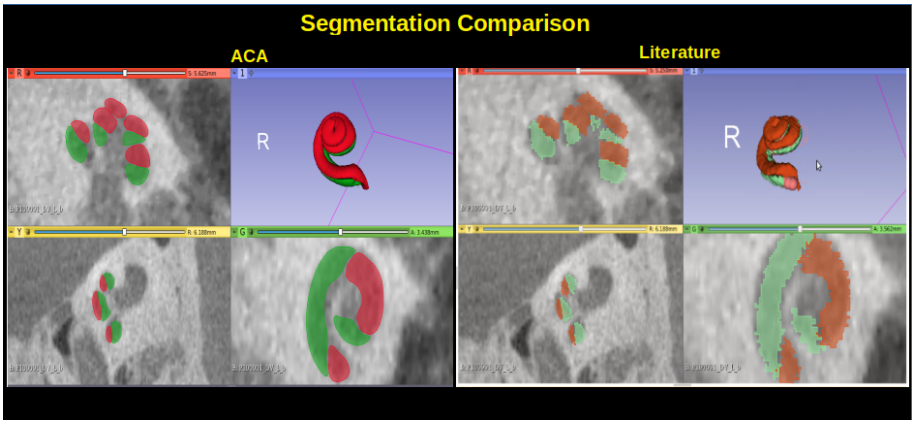


Figure 5.8: Cochlea Atlas-Model comparison. ACA2018 has low resolution comparing to ACA2020.

through the outer wall of the scala tympani, see figure 5.10, points are in green color.

- Scala tympani organ of Corti points model (StOC): a set of 100 points that passes through the inner wall of the scala tympani (the organ of Corti), see figure 5.10, points are in red color.

- A-value endpoints model: the end points of the A-value method. This helps to automate the process of measuring A-value and get the scala tympani lateral and organ of Corti lengths using the A-value method automatically, see figure 5.1.

Using these point models one can get all standard measurements from different methods, e.g. A-value which helps doctors deciding the best implant type for a specific patient. ACA2020 has the advantage because it works on any type of images, even when high noise is present. Moreover, the points-to-image and image-to-points conversion processes in ACA2018 are removed in ACA2020. The transformation found by ACIR2020 is applied directly to the points using 3D Slicer[9]. This solves the issue with missing points in ACA2018.

[9]Tip: use Elastix to generate the transformation file, after that use Transformix to generate a deformation field image, e.g. in ".nrrd" format. This deformation-field image can be loaded to 3D Slicer as a transform. In 3D Slicer, transform can be applied on any node, e.g. points, segmentation, ...etc.

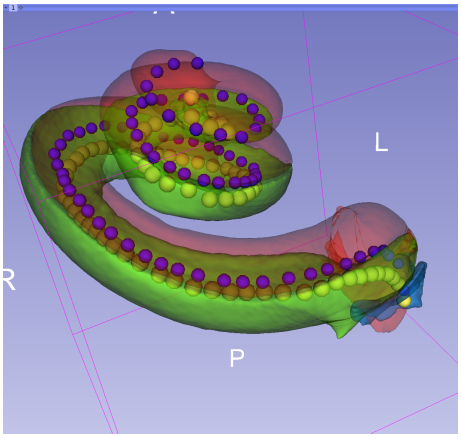


Figure 5.9: The point-models of scala tympani (yellow color) and scala vestibuli (blue color). Each points-model measures a length of a curve passes by the center of the scala.

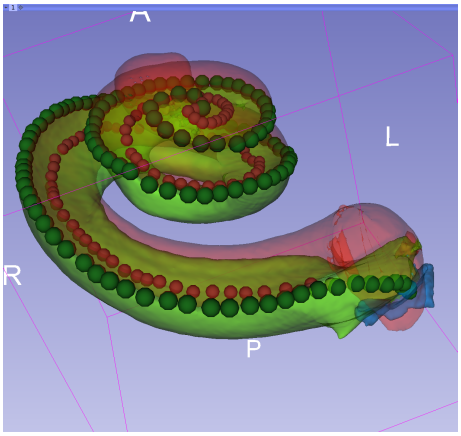


Figure 5.10: The point-models of scala tympani. The outer points-model is for measuring the lateral length (dark green color). The inner points-model is for measuring the organ of Corti length (dark red color).

Algorithm 5: ACA2020

Input: cochlea image I(P) ;
Output: cochlea segmentation and different measurements ;
Locate the cochlea locations in input images;
$\tilde{I}(P)$= Crop($I(P)$);
Apply ACIR2020 on the cropped input image and the predefined image ;
    $I_{res,T} = (\tilde{I}(P), I_M(P))$;
Register the resulted image with the model image using B-spline transform ;
    T = register $(\tilde{I}_{res0}, I_M(P))$;
Transform the segmentation and the point models using T_{final} and its parameters μ;
    $I_{seg}(P) = I_{M_{seg}}(T(P), \mu)$;
    $I_{points}(P) = I_{M_{points}}(T(P), \mu)$;
Output the segmentation, $I_{seg}(P)$, the segmentation sizes, and length of points model $I_{points}(P)$

Chapter 6

Results and Discussion

The results of my experiments are divided into two parts: image registration results, and image segmentation and analysis results. The following sections describe the design of the experiments in details and list samples from the results. The complete results of statistical files with all related scripts will be attached to the digital version of this dissertation.

6.1 Cochlea Image Registration Results

The goal is to compare the accuracy and the time required by the methods I proposed: ACIR2016, ACIR2017 and ACIR2020, against the related state of the arts optimisers, i.e. ASGD, FASGD and FPSGD, using the same metric, transformation and other parameters.

I created four types of experiments based on image modalities. This gives the detailed results and describes the behaviour of different image registration methods on different types of images. The groups are:

1. CBCT to CBCT.
2. CBCT to CT.
3. CBCT to MR.
4. CT to MR.

I wrote a python script that runs an image registration of patient images, the procedure is repeated 3 times for justification purposes. The average Root Mean Squared Error (RMSE) [DCRB19] of the fixed image landmarks and the transformed moving image landmarks is computed from the 3 versions of the results. Similarly, the average required time to register a pair of images is computed. After

that, the total RMSE and time from all results are computed along with the Standard Deviation (STD). This should give a good estimation of the accuracy and the required time of each method. The total number of the image registration experiments were 744 experiments represented 214 images of 41 patients from Germany datasets[1], as Egypt datasets have only one image per patient. The section 6.1 shows sample of the result's table.

The time recorded is the total time which includes reading the input images, writing the output results and the preprocessing steps, e.g. the cropping.

The robustness of the method is computed based on the availability of valid output. The Invalid output may happen if the optimisation process stopped for some reason without producing an output transform, in this case, the result is missing.

$$r = 100(\frac{v}{v+i}) \quad , \tag{6.1}$$

where r is the robustness percentage, v is a number of valid results, and i is the number of invalid results. If the method works successfully on all the tested images and produces valid results all the time, the method robustness is 100% despite if the accuracy is bad.

Designing the experiments this way gives local view of each image registration method with-in different cochlea image types in addition to a global view of the performance of these methods on all images.

Using ASGD2009 without the cropping, the total RMSE median[2] was 17.55 mm with a range of [0.29-36.86] mm. The total time average required for registration of an image pair using this method was 6.44 seconds with STD of 0.82 seconds. The method without modification produced large error in some cases, hence it is not recommended to be used. The improved method FASGD2015 total[3] RMSE median was 0.53 mm with range of [0.08-102.35] mm. The total time average required for registration of an image pair using this method was 4.80 seconds with STD of 1.27 seconds. One can see from the large error in some cases that the method is not recommended as well. The recently published method FPSGD2019 total RMSE median was 0.44 mm with a range of [0.08-88.74] mm. The total time average required for registration of an image pair using this method was 5.02 seconds with STD of 1.47 seconds. This method also has a large error in some images.

[1]In more details: 124 images, 75 CBCT, 9 CT, 40 MR, 77 left, 88 right, 81 before surgery, and 43 after surgery.

[2]The median and the range are used instead of the mean and the standard deviation when the data is not normally distributed.

[3]Note: The missing values are replaced by the maximum error to penalized the method and produces fair comparison with the other methods.

	A	B	C	D	E	F	G
1	idx	Moving Image	Fixed Image	Method	MSE (mm)	Time (seconds)	Note
2	1	P100001_DV_L_b	P100001_DV_L_a	T00_Opt_ACIR2016	2.9515	14.3576710224	
3	2	P100001_DV_L_b	P100001_DV_L_a	T01_Opt_ACIR2017	0.4475	5.20184898376	
4	3	P100001_DV_L_b	P100001_DV_L_a	T02_Opt_ACIR2020	0.449166666667	5.21927380562	
5	4	P100001_DV_L_b	P100001_DV_L_a	T03_Opt_ASGD2009	3.41816666667	5.35579490662	
6	5	P100001_DV_L_b	P100001_DV_L_a	T04_Opt_FASGD2015	0.543166666667	5.33399200439	
7	6	P100001_DV_L_b	P100001_DV_L_a	T05_Opt_FPSGD2019	0.472666666667	5.18830394745	
8	7	P100001_DV_L_a	P100001_MR	T00_Opt_ACIR2016	7.79	14.999063015	
9	8	P100001_DV_L_a	P100001_MR	T01_Opt_ACIR2017	0.615	4.1475379467	
10	9	P100001_DV_L_a	P100001_MR	T02_Opt_ACIR2020	0.668166666667	3.86424899101	
11	10	P100001_DV_L_a	P100001_MR	T03_Opt_ASGD2009	17.2905	3.85375404358	
12	11	P100001_DV_L_a	P100001_MR	T04_Opt_FASGD2015	0.656666666667	4.06173682213	
13	12	P100001_DV_L_a	P100001_MR	T05_Opt_FPSGD2019	1.7625	3.79101395607	
14	13	P100001_DV_L_b	P100001_MR	T00_Opt_ACIR2016	6.82916666667	15.2931041718	
15	14	P100001_DV_L_b	P100001_MR	T01_Opt_ACIR2017	0.529166666667	4.77348899841	
16	15	P100001_DV_L_b	P100001_MR	T02_Opt_ACIR2020	0.550333333333	4.12127494812	
17	16	P100001_DV_L_b	P100001_MR	T03_Opt_ASGD2009	15.7018333333	4.36567211151	
18	17	P100001_DV_L_b	P100001_MR	T04_Opt_FASGD2015	0.542666666667	4.44771289825	
19	18	P100001_DV_L_b	P100001_MR	T05_Opt_FPSGD2019	0.5355	4.12250614166	
20	19	P100002_DV_R_b	P100002_DV_R_a	T00_Opt_ACIR2016	1.2965	14.127076149	
21	20	P100002_DV_R_b	P100002_DV_R_a	T01_Opt_ACIR2017	0.335333333333	5.06379008293	
22	21	P100002_DV_R_b	P100002_DV_R_a	T02_Opt_ACIR2020	0.340666666667	5.21635198593	
23	22	P100002_DV_R_b	P100002_DV_R_a	T03_Opt_ASGD2009	1.75683333333	5.25991201401	
24	23	P100002_DV_R_b	P100002_DV_R_a	T04_Opt_FASGD2015	0.2855	5.07623004913	
25	24	P100002_DV_R_b	P100002_DV_R_a	T05_Opt_FPSGD2019	0.3375	4.95180606842	
26	25	P100002_DV_R_a	P100002_MR	T00_Opt_ACIR2016	21.0956666667	12.9863569736	
27	26	P100002_DV_R_a	P100002_MR	T01_Opt_ACIR2017	0.499166666667	2.47333192825	
28	27	P100002_DV_R_a	P100002_MR	T02_Opt_ACIR2020	0.584333333333	2.56497216225	
29	28	P100002_DV_R_a	P100002_MR	T03_Opt_ASGD2009	14.156	2.38788199425	
30	29	P100002_DV_R_a	P100002_MR	T04_Opt_FASGD2015	0.502166666667	2.44550895691	
31	30	P100002_DV_R_a	P100002_MR	T05_Opt_FPSGD2019	2.87066666667	2.4946539402	
32	31	P100002_DV_R_b	P100002_MR	T00_Opt_ACIR2016	10.433	12.4684598446	
33	32	P100002_DV_R_b	P100002_MR	T01_Opt_ACIR2017	0.3735	2.89469790459	
34	33	P100002_DV_R_b	P100002_MR	T02_Opt_ACIR2020	0.3545	3.0549030304	
35	34	P100002_DV_R_b	P100002_MR	T03_Opt_ASGD2009	12.6111666667	2.95450115204	
36	35	P100002_DV_R_b	P100002_MR	T04_Opt_FASGD2015	0.558	3.41547417641	
37	36	P100002_DV_R_b	P100002_MR	T05_Opt_FPSGD2019	0.408833333333	2.94891691208	

Figure 6.1: A sample from a result table that shows all experiments of two patients.

Total RMSE median of ACIR2016 was 9.52 mm with a range of [0.21-49.78] mm. The total time average required for registration of an image pair using this method was 15.96 seconds with STD of 3.37 seconds. Comparing to ASGD, ACIR2016 may produce better results but it still required larger time and the accuracy is not good enough in practice. The total RMSE average of ACIR2017 was 0.36 mm with a STD of 0.17 mm. The total time average required for registration of an image pair using this method was 4.64 seconds with STD of 1.19 seconds. Total RMSE median of ACIR2020 was 0.34 mm with a range of [0.07-2.12] mm. The total time average required for registration of an image pair using this method was 4.62 seconds with STD of 1.19 seconds. Both ACIR2017 and ACIR2020 results have high accuracy and required a few seconds to complete a registration of a pair of images.

To confirm the results, I compared ACIR2020 to the other methods using statistical tests[4] with 1000 boot-strapping, I found out that the resulted means were significant (p-value < 0.001 for RMSE and p-value <0.003 for time) except for ACIR2017 (p-value $= 0.168$ for RMSE and p-value 0.277 for time). This means

[4]RMSE: Wilcoxon test. Time: Two-samples paired t-test.

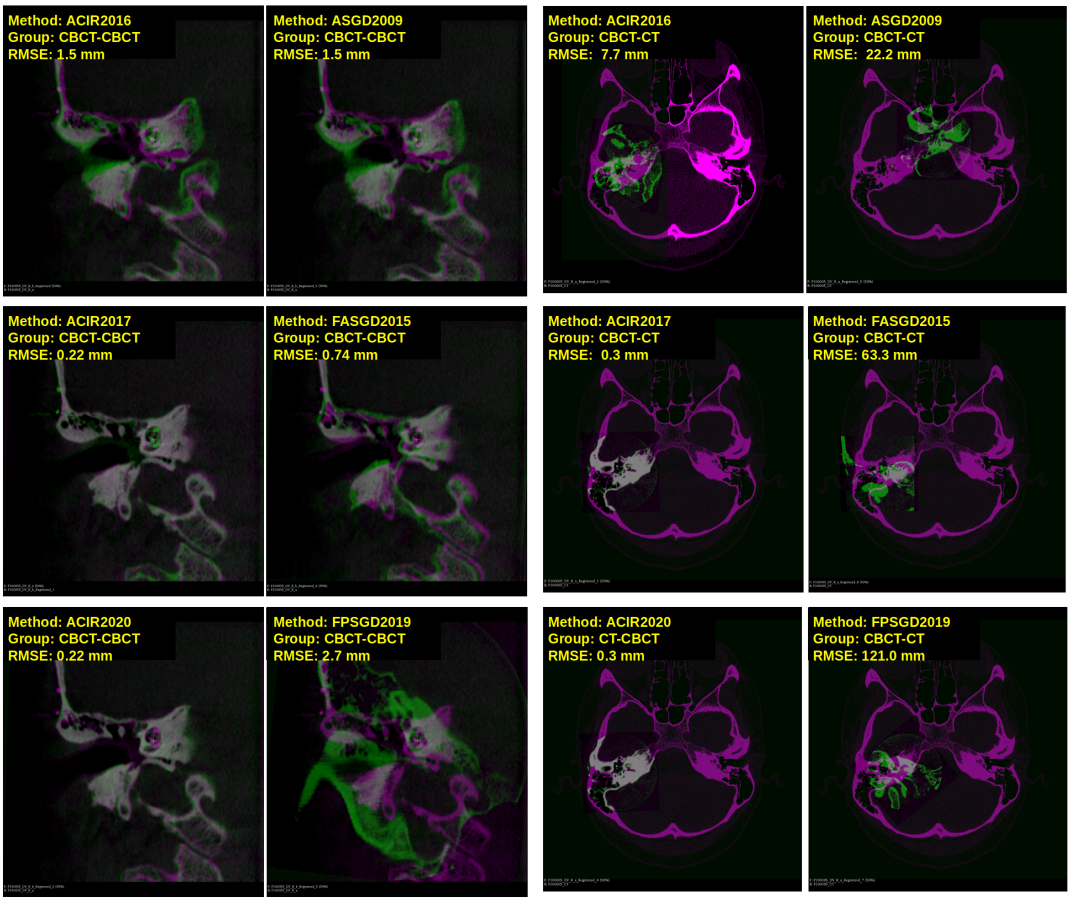


Figure 6.2: Sample results of registration CBCT to CBCT (coronal views) and CBCT to CT (axial views). The figure shows image types, methods and RMSE information.

that there is no difference between ACIR2017 and ACIR2020.

In figure 6.2, the visual samples of CBCT to CBCT and CBCT to CT image results of all 6 methods are shown. The fixed image in magenta colour and the moving image in green color. The RMSE error between the ground truth and the registered ground truth landmarks is also shown. Similarly, figure 6.3 shows visual samples of CBCT to MR and MR to CT image registration and fusion results. The three charts in figure 6.4 show the comparison of the 6 methods in more details terms of accuracy, time and robustness. They show the result for each modality group in addition to the total result from all images. ASGD, ACIR2017, and ACIR2020 methods were 100% robust.

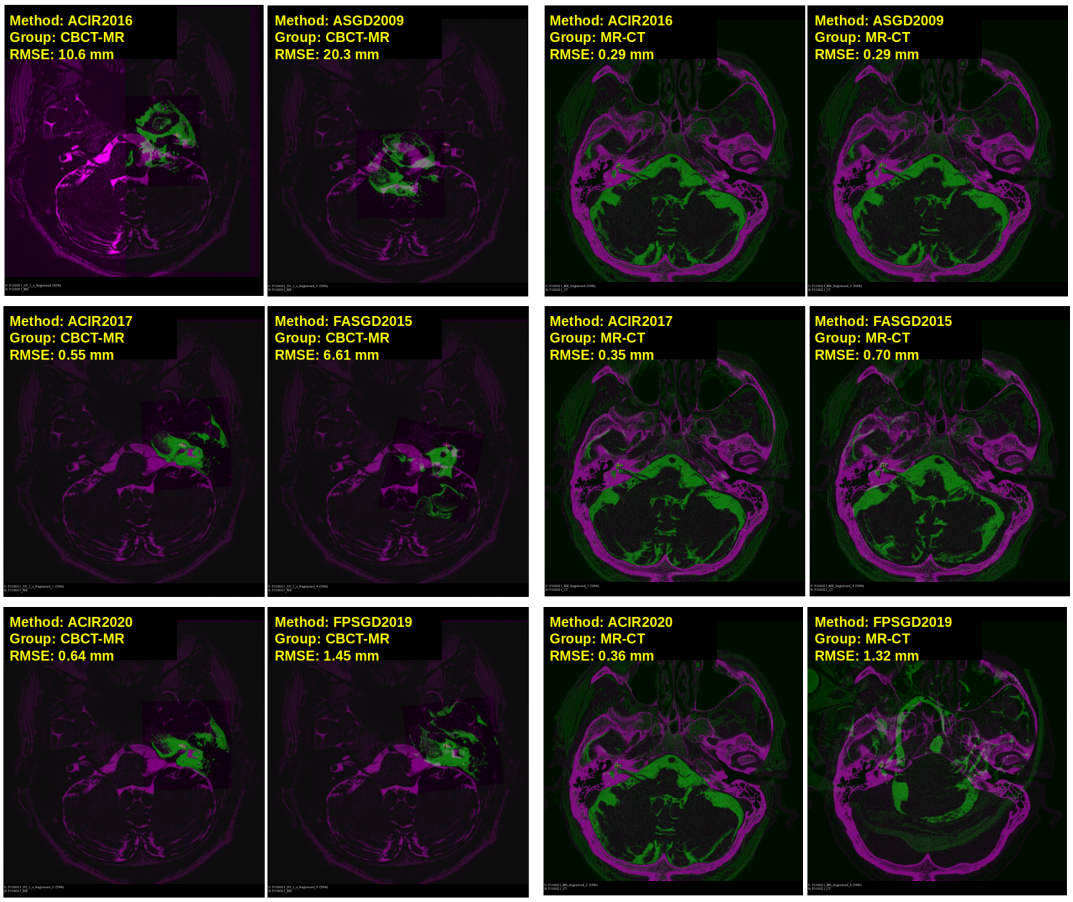


Figure 6.3: Sample results of registration CBCT to MR and MR to CT. The figure shows the axial views with image types, methods and RMSE information.

6.2 Cochlea Image Segmentation and Analysis Results

The results in this sub-section are divided into two parts. The first part is the result of the ACA2018 method [ADBP+18b]. The second part is the result of the improved ACA2020 method which is more accurate and provides more information. ACA2018 is not used anymore and replaced by ACA2020 in the update Slicer plug-in. I mention ACA2018 results here to show the historical development of the method and why it is necessary to have a better method, e.g. ACA2020.

In ACA2018 [ADBP+18b], information of implants is collected from the images of 21 patients of different age and gender[5]. All patients haves a CI in one or both ears. The length of CI in these images is used as a ground truth. This pre-

[5]In more details: 71 images, 39 CBCT, 25 CT, 7 MR, 32 left, 39 right, 47 before surgery, 24 after surgery.

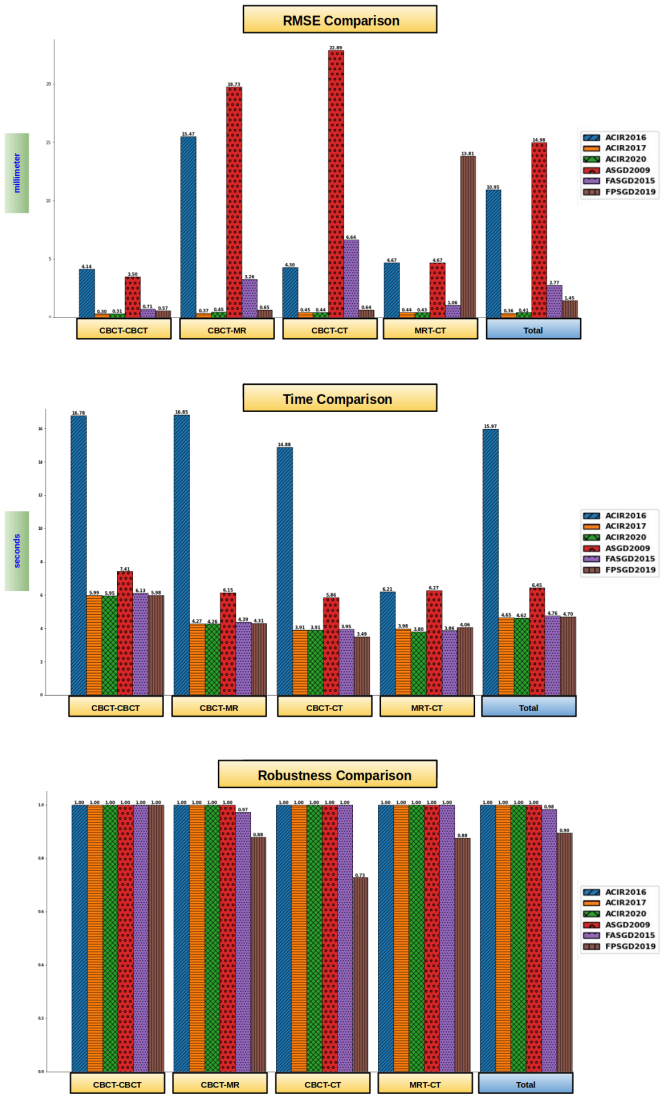


Figure 6.4: Results chart, the mean RMSE, time and robustness of each tested method grouped based on image-modalities. The most right group is all the images from all modalities. The robustness is computed based on equation (6.1). More details are provided in the text. Note that missing values are not computed in the mean RMSE and time computation.

knowledge of these CI types gives us more understanding about the artefact error, see figure 6.5[16], and table table 6.1.

The main challenging in this study is that the ground truth is not available due to the low resolution of the clinical images. Using information about the active

Table 6.1: Sample of CI types

Idx	CI	electrodes	ActiveLength	Notes
1	Cochlear Nucleus CI24RE Contour Advance	22	15	
2	Cochlear Nucleus CI512	22	15	
3	Cochlea CI422	22	20	
4	Cochlea Nucleus CI532	22	14	

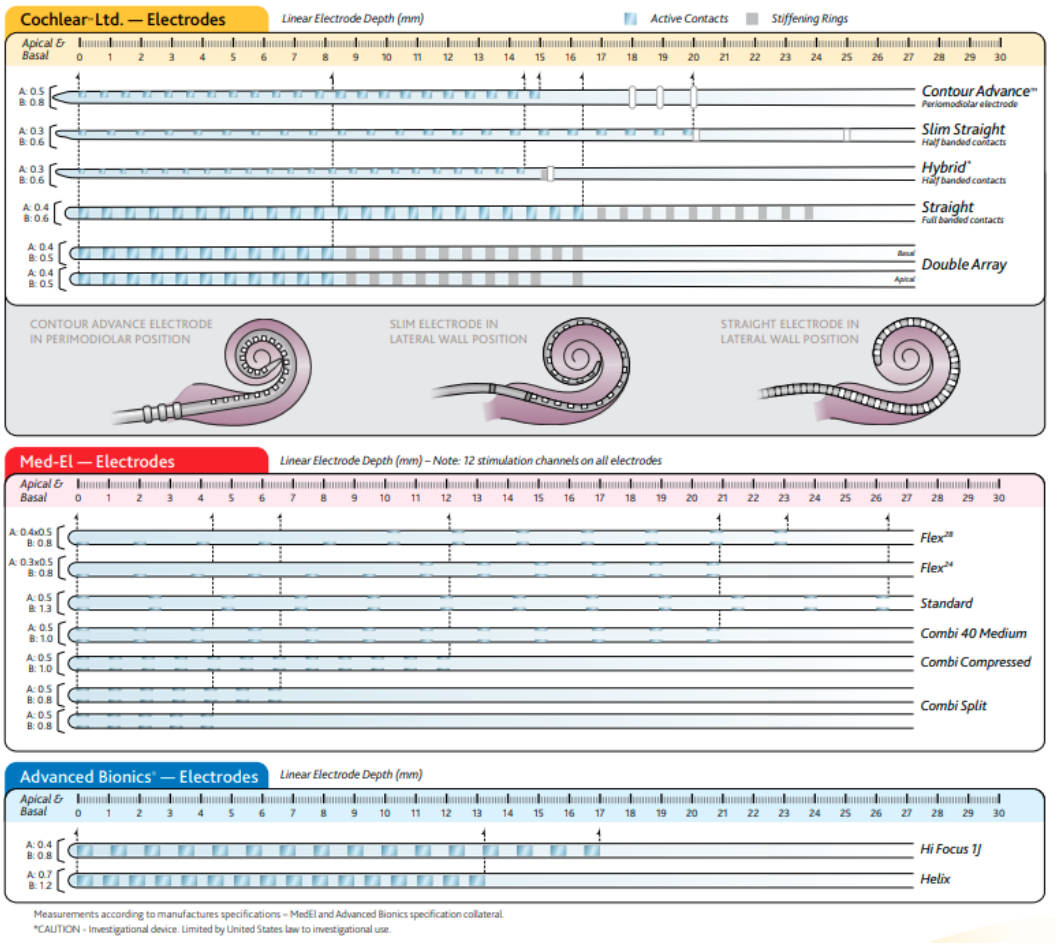


Figure 6.5: Different types of cochlea implants with different lengths from different companies [15].

length of CI may give an estimation of the error. The active length of the CI is measured from the images and compared to the actual length, see figure 6.5. This gives us information about how much the image artefact can be misleading.

To get a more accurate measurement, there are two challenges. The first, a public and well-described cochlea dataset from different modalities and regions is needed. The second challenge is related to the manual segmentation of a clinical

cochlea's image which is not accurate. For a specific image, the expert can not provide the same segmentation if he segments the image multiple time. An average manual segmentation should be computed from a number of manually segmented images of a specific image from the same person. This needs more resources i.e cochlea's experts, time and effort.

The cochlea locations are estimated 10 times manually by 3 different students[6]. The images are segmented automatically using ACA2018, then the size and the length of the scala tympani are extracted. Finally, the average and the standard deviation of the results from the ten estimated locations are recorded.

There is a number of factors that may affect the accuracy of the results. I studied a few important factors in my experiments and how they affect the measurement's accuracy. These factors are the cochlea localisation step and the artefact generated by the image scanners.

The only manual step in my method is locating the cochlea in which the user provides a point inside the cochlea region. This point is used in the proposed method for the cropping phase where the region around the cochlea is cropped and used in the third step in figure 5.3 i.e. registration phase. The point can be anywhere inside the cochlea so the accuracy here should not be a problem. Since this step is used only for cropping, the image registration process and the followed atlas transformations are fully automated.

The selected point is considered as a center of the cropping cube of 10 mm length. This generates a 3D cropped image of $80 \times 80 \times 80$ voxels with 0.125 mm spacing. I selected this length based on many experiments as I found that the main cochlea shape is completely located inside this cropped size in all the tested images in HCD dataset.

Since the selection of this cropping point can be different e.g. from user to another or if the same person does the locating multiple time, the result may change slightly. For justification, three students located the cochlea points 10 times for each input image and the average and the standard deviation are recorded.

In figure 6.6, and table 6.2, sample results from three cochlea different location points of the same image with their quantitative and visual results are shown. The points set detection is related to the number of the points that transformed to the new input image[7]. Notice that how the quantitative results slightly differ while visually they look almost the same.

To find out the error resulted from the artefact in the image, I used prior knowledge about the images in HCD dataset. I used the actual active length of

[6]It is easy to recognise the location of the cochlea by non-expert in the coronal view because of the cochlea's spiral shape

[7]Recall that ACA2018 transforms the points-model to an image, transforms the resulted image and extracts the points back which results some missing points due to the interpolation.

Table 6.2: Sample Of cochlea location and related results of a Patient.

Cochlea Location	PointSet Detection	Scala Tympani Length	Scala Tympani Size	Scala Vestibuli Size
216 247 78	0.91	28.4032	41.1152	34.8085
218 250 77	0.96	29.4105	41.3886	35.3632
223 254 77	0.98	30.4227	41.7539	36.8027

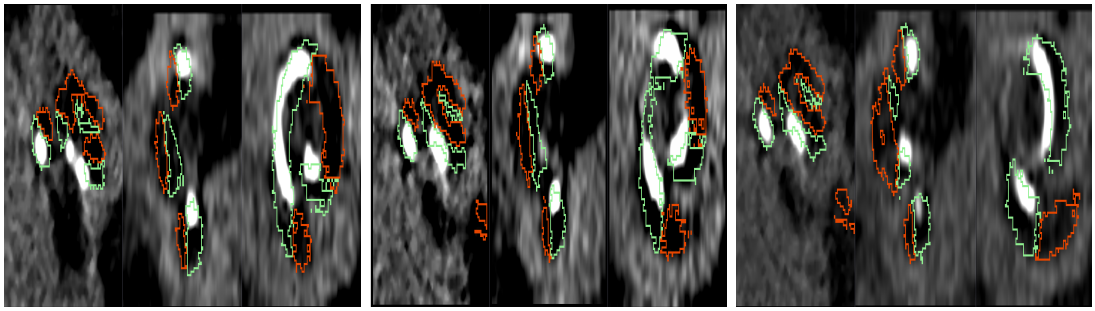


Figure 6.6: ACIR's Cochlea Location Change Effect

the electrode as a ground truth. In Tables 6.3 and Table 6.4, the estimated and the actual active length of the electrode are listed in mm from two different cochlea implant providers, i.e. Cochlear[8] and MedEL[9].

There is usually a small human error that comes from the image artefact, the human eye can not detect the end-points of CI correctly. One needs more data with variable electrode array length to make a better judgment of the estimated error.

From the two tables above, one can conclude that there will be a small human error ranges from 0.5 mm to 1.12 mm. However, more data and experiments should be done to make a solid statement.

[8] `www.cochlear.com`
[9] `www.medel.com`

Table 6.3: Cochlea implant measurement in mm (Cochlear company products).

Idx	Image	CI	CI Actual Length	CI Estimated Length	Error
1	P100001DVLa	CI24RE	15.00	14.50	0.50
2	P100002DVRa	CI24RE	15.00	14.35	0.65
3	P100003DVRa	CI24RE	15.00	14.27	0.73
4	P100004DVLa	CI24RE	15.00	14.28	0.72
5	P100005DVLa	CI24RE	15.00	14.49	0.51
6	P100005DVRa	CI24RE	15.00	14.16	0.84
7	P100006DVRa	CI24RE	15.00	14.39	0.61
8	P100007DVRa	CI24RE	15.00	14.81	0.19
9	P100008DVLa	CI24RE	15.00	14.82	0.18
10	P100009DVa	CI24RE	15.00	14.00	1.00
11	P100009DVRa	CI24RE	15.00	14.68	0.32
12	P100011DVRa	CI512	15.00	14.30	0.70
13	P100012DVLa	CI512	15.00	14.40	0.60
14	P100014DVLa	CI512	15.00	13.88	1.12

Table 6.4: Cochlea implant measurement in mm (MedEL company products).

Idx	Image	CI	CI Actual Length	CI Estimated Length	Error
1	P100013DVLa	MedEl Synchrony Medium	20.90	20.73	0.17
2	P100013DVRa	MedEl Synchrony Medium	20.90	20.95	-0.05
3	P100015DVLa	MedEl Synchrony Medium	20.90	20.96	-0.06
4	P100016DVLa	MedEl Synchrony Medium	20.90	20.44	0.46
5	P100017DVRa	MedEl Synchrony Medium	20.90	20.24	0.66
6	P100018DVRa	MedEl Synchrony Medium	20.90	20.98	-0.08
7	P100019DVRa	MedEl Synchrony Medium	20.90	21.15	-0.25
8	P100030DVLa	MedEl Synchony Flex28	23.10	23.17	-0.07
9	P100031DVRa	MedEl Synchony Flex28	23.10	23.46	-0.36
10	P100010DVLa	MedEl Synchrony PIN Medium	20.90	20.76	0.14

In Table 6.5, each row represents a patient, DVa is the CBCT image after the surgery, DVb is the CBCT image before the surgery and MRCT is the CT or MR image before the surgery[10].

The columns 3-5 represent the automatically computed length of the scala tympani. The columns 6-8 represent the difference length between each two different image types. Column 9 computes the average of scala tympani length from all patient's images.

From the table, notice that there is a small error when computing the length from image to another. This error is around 1.0 mm. Taking the average scala tympani's length of different images gives us a better estimation of the length. In the images I experimented with, the scala tympani length ranges from 28.79 mm to 30.04 mm. Since this method uses rigid transformation[11], it seems the average length does not represent the patient correctly and the experts must modify the end-points of the model-points manually to get an accurate length.

Similar to table 6.5, table 6.6 shows that there is a small error while computing the volume size of scala tympani from image to another. However, this error should not exceed 1.5 mm^3. Similarly, Table 6.7 shows a maximum error of 2.0 mm^3 while computing the volume size of scala vestibuli from image to another. Taking the average volume size of different images gives us a better estimation of the volume size. Based on the result tables, the scala tympani volume size ranges from 41.05 mm^3 to 41.77 mm^3 and the scala vestibule and scala media combined volume size ranges from 34.90 mm^3 to 36.66 mm^3. However, due to the rigid transformation used, the result are not accurate and one should use a non-rigid transformation to get more accurate results.

From the results above, it seems that the length of scala tympani is enough to have a longer cochlear implant than the ones available on the market today.

[10]This is based on the availability of the dataset, e.g. some patients have images with two ears, while others have images of one ear only. Also, some patients have CBCT and MRI, others have CBCT and CT.

[11]Recall that rigid transformation does not have scaling as explained in chapter 2

Table 6.5: Cochlea scala tympani estimated length.

Idx	Image	DVa	DVb	MRCT	DVa-DVb	DVa-MRCT	DVb-MRCT	AVG
1	P100001STL	29.23	29.86	29.98	0.63	0.75	0.12	29.69
2	P100002STL	29.59	29.76	29.76	0.17	0.18	0.00	29.70
3	P100003STL	29.52	29.44	29.42	0.08	0.09	0.02	29.46
4	P100004STL	29.23	29.10	29.70	0.13	0.47	0.60	29.34
5	P100005STL	29.58	28.94	29.28	0.64	0.30	0.34	29.27
6	P100005STL	29.77	30.71	29.64	0.95	0.13	1.08	30.04
7	P100006STL	29.83	29.93	29.68	0.10	0.15	0.25	29.82
8	P100007STL	29.66	29.38	29.36	0.28	0.30	0.02	29.46
9	P100008STL	29.47	0.00	29.45	0.00	0.03	0.00	29.46
10	P100009STL	29.97	29.95	29.44	0.02	0.52	0.51	29.79
11	P100009STL	29.86	29.86	29.58	0.00	0.27	0.28	29.76
12	P100010STL	29.64	29.66	29.72	0.02	0.08	0.06	29.68
13	P100011STL	29.84	29.58	29.77	0.26	0.07	0.19	29.73
14	P100012STL	29.78	29.45	29.31	0.32	0.47	0.15	29.51
15	P100013STL	29.42	29.17	29.71	0.24	0.29	0.54	29.43
16	P100013STL	29.21	29.51	29.51	0.30	0.30	0.00	29.41
17	P100014STL	29.57	29.88	29.78	0.31	0.21	0.10	29.74
18	P100015STL	28.61	29.36	28.39	0.75	0.23	0.97	28.79
19	P100016STL	29.25	28.74	29.62	0.51	0.37	0.88	29.20
20	P100017STL	29.44	29.49	29.66	0.05	0.22	0.18	29.53
21	P100018STL	30.11	29.61	29.89	0.50	0.23	0.28	29.87
22	P100019STL	28.85	29.27	29.69	0.42	0.84	0.42	29.27
23	P100030STL	28.96	29.27	29.69	0.31	0.73	0.42	29.31
24	P100031STL	29.58	29.96	29.56	0.38	0.01	0.39	29.70

Table 6.6: Cochlea scala tympani estimated volume size.

Idx	Image	DVa	DVb	MRCT	DVa-DVb	DVa-MRCT	DVb-MRCT	AVG
1	P100001STS	41.00	41.59	41.25	0.60	0.26	0.34	41.28
2	P100002STS	41.75	41.77	41.78	0.01	0.03	0.01	41.77
3	P100003STS	41.77	41.78	41.73	0.01	0.04	0.05	41.76
4	P100004STS	41.24	41.56	41.76	0.32	0.52	0.20	41.52
5	P100005STS	41.33	40.52	41.37	0.81	0.04	0.85	41.07
6	P100005STS	41.62	41.73	41.69	0.11	0.07	0.04	41.68
7	P100006STS	41.75	40.64	41.74	1.10	0.00	1.10	41.38
8	P100007STS	41.75	41.79	41.40	0.04	0.35	0.39	41.65
9	P100008STS	41.54	NA	41.68	NA	0.13	NA	41.61
10	P100009STS	41.74	41.31	41.74	0.43	0.01	0.43	41.60
11	P100009STS	41.51	41.75	41.65	0.24	0.14	0.11	41.64
12	P100010STS	41.77	41.76	41.73	0.01	0.04	0.04	41.75
13	P100011STS	41.63	41.75	41.52	0.12	0.11	0.23	41.64
14	P100012STS	41.61	41.46	41.67	0.14	0.07	0.21	41.58
15	P100013STS	41.59	41.76	41.76	0.18	0.17	0.01	41.70
16	P100013STS	41.47	41.78	41.75	0.31	0.28	0.03	41.67
17	P100014STS	41.75	41.79	41.58	0.04	0.17	0.21	41.70
18	P100015STS	41.05	41.52	40.57	0.48	0.47	0.95	41.05
19	P100016STS	41.44	40.99	41.52	0.45	0.08	0.53	41.32
20	P100017STS	41.76	41.71	41.65	0.06	0.12	0.06	41.71
21	P100018STS	41.75	41.68	41.72	0.07	0.02	0.05	41.72
22	P100019STS	41.38	41.61	41.70	0.23	0.32	0.09	41.57
23	P100030STS	41.38	41.49	41.72	0.11	0.34	0.23	41.53
24	P100031STS	41.76	41.56	41.64	0.20	0.12	0.08	41.65

Assuming an error of 1.0 mm and a range of 28.79 mm to 30.04 mm, the results suggest a CI active length range from 27.0 mm to 29.0 mm. However, to verify these measurements, one needs to experiment with more datasets from different locations.

The hardware used was a notebook Asus ROG G751 equipped an Intel i7 CPU,a 32 GB memory card and a Nvidia GTX 970m graphics card. The implementation provided as an open-source in 3D Slicer 4.8 software [KPV14] and Elastix 4.8 toolbox [KSM+10].

Table 6.7: Cochlea scala vestibule estimated volume size.

Idx	Image	DVa	DVb	MRCT	DVa-DVb	DVa-MRCT	DVb-MRCT	AVG
1	P100001SVS	34.8	35.94	35.86	1.14	1.06	0.08	35.53
2	P100002SVS	36.61	36.8	36.58	0.19	0.03	0.22	36.66
3	P100003SVS	36.74	36.75	36.36	0.01	0.38	0.4	36.62
4	P100004SVS	34.97	35.71	36.75	0.74	1.77	1.03	35.81
5	P100005SVS	35.57	34.39	35.31	1.18	0.26	0.92	35.09
6	P100005SVS	35.99	36.39	36.2	0.4	0.21	0.19	36.2
7	P100006SVS	36.68	35.88	36.37	0.8	0.31	0.49	36.31
8	P100007SVS	36.42	36.45	35.75	0.03	0.67	0.7	36.21
9	P100008SVS	36.3	NA	36.14	NA	0.16	NA	36.22
10	P100009SVS	36.42	35.52	36.69	0.9	0.27	1.17	36.21
11	P100009SVS	36.05	36.44	36.3	0.39	0.25	0.15	36.27
12	P100010SVS	36.4	36.41	36.45	0.01	0.05	0.04	36.42
13	P100011SVS	36.13	36.35	36.08	0.21	0.06	0.27	36.19
14	P100012SVS	35.89	35.48	36.1	0.42	0.2	0.62	35.82
15	P100013SVS	36.31	36.6	36.73	0.3	0.42	0.13	36.55
16	P100013SVS	35.83	36.68	36.62	0.85	0.79	0.05	36.38
17	P100014SVS	36.17	36.69	36.48	0.52	0.32	0.21	36.45
18	P100015SVS	34.53	35.66	34.51	1.13	0.03	1.15	34.9
19	P100016SVS	35.87	35.01	36.7	0.87	0.83	1.7	35.86
20	P100017SVS	36.47	36.14	36.06	0.32	0.41	0.08	36.22
21	P100018SVS	36.41	36.26	36.3	0.16	0.11	0.04	36.32
22	P100019SVS	35.47	35.91	36.27	0.44	0.8	0.36	35.88
23	P100030SVS	35.43	35.82	36.46	0.39	1.02	0.64	35.91
24	P100031SVS	36.5	36.43	35.9	0.07	0.61	0.53	36.28

	G	E	Total
Patients	41	26	**67**
Images	122	49	**171**
CBCT	75	12	**87**
CT	9	13	**22**
MR	38	0	**38**
Left	80	24	**104**
Right	88	25	**113**
Before surgery	79	25	**104**
After surgery	43	24	**67**

	G	E	Total
Patients	34	26	**60**
Images	96	53	**149**
CBCT	57	12	**69**
CT	7	14	**21**
MR	0	0	**0**
Left	33	22	**55**
Right	33	24	**57**
Before surgery	34	25	**59**
After surgery	32	21	**53**

Figure 6.7: ACA2020 dataset details. Left: Automatic experiments. Right: Manual experiments. G column represents dataset from Germany while E column represents dataset from Egypt.

In ACA2020, more datasets are used from different geographical locations. The total number of images used in this study was 171 multi-modal 3D images of 67 patients of mixed gender and age from Germany (41 patients) and Egypt (26 patients), see figure 6.7 for details. The total number of experiments was 217 for ACA2020 and 149 for the manual measurement using A-value method[12].

[12]Note that some images have two ear sides.

The challenging part of clinical cochlea images is that there is no ground truth, as the cochlea structure, i.e. cochlea scalae, is invisible. Since the round window and the top of the cochlea can be "somehow" seen in these images, locating landmarks can be a useful tool for validation and error estimation. Reliable landmarks should be done by many experts multiple times for each image then the average should be used. This may reduce a human error related to locating these landmarks manually. Unfortunately, I was able to obtain landmarks that are done by two experts only. However, this may provide a quantitative error estimation that helps to judge the results.

The method results accuracy has been evaluated using RMSE of two 3D point landmarks located by two experts. The landmarks represent the cochlea round window and the cochlea apex (helicotrema), see figure 5.1. The total number of landmarks used was 434 landmarks[13]. After aligning the model to the input image, the landmarks of the model are transformed by the same resulted transformation to the input image. Thereafter, RMSE in mm is measured between these transformed landmarks and the related input image landmarks. This gives a good quantitative error estimation and produces a good evaluation of the proposed method.

The time required to obtain the segmentation and all measurements have been recorded including the cropping process. Since the result time recorded might differ according to the hardware used, all the experiments of ACA2020 were done using the same hardware. The hardware used was a computer equipped an AMD Ryzen 3900 CPU, a 32 GB memory card and a Nvidia RTX2080Ti graphics card.

I used the same parameters and the original implementations of ASGD and S-LBFGS by their authors, which are provided in Elastix 5.0.0 toolbox [KSM$^+$10].

For justification, the results are repeated 3 times and the average values are used. To get detailed information, the images are divided into different groups based on their types CBCT, MR, CT and images with cochlea implant CI. This helps to get more details on how ACA2020 performs on a specific modality. In addition to these group results, the total results are presented to give a global evaluation as well.

Total RMSE average was 0.61 mm with a STD of 0.22 mm. For CBCT with implant group, the average RMSE was 0.54 mm with STD of 0.17 mm. The CBCT without implant group had the average RMSE of 0.60 mm with STD of 0.18 mm, while the CT showed the average RMSE of 0.64 mm with STD of 0.17 mm. As for MR group, the average RMSE was 0.60 mm with STD of 0.25 mm.

The total time average required for analysing an image was 5.21 seconds with STD of 0.93 seconds. The CBCT with an implant group resulted in the average time of 5.18 seconds with STD of 0.27 seconds. CBCT without an implant group had the average time of 5.32 seconds with STD of 0.3 seconds. The CT group

[13]Some images were too noisy to the point expert can not locate the landmark points.

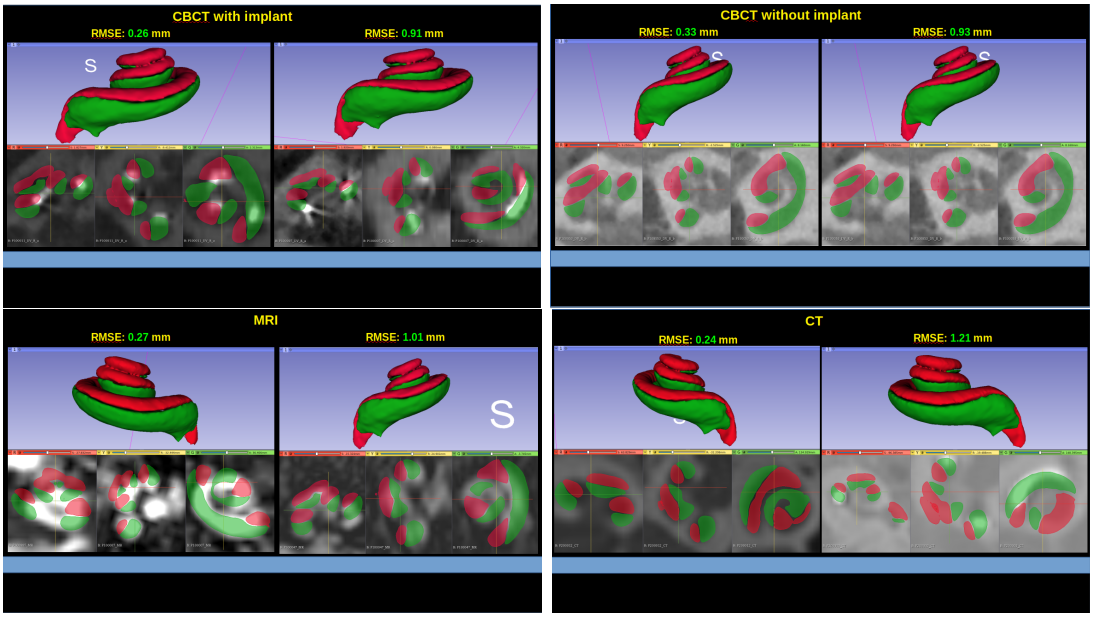


Figure 6.8: Sample from the results from different image types with RMSE values, for each group the one in the left has the lowest RMSE and the one in the right has the highest RMSE. Each result has the 3D model on top and axial, sagittal and coronal views in bottom.

presented the average time of 5.18 seconds with STD of 0.67 seconds, and for the MR group,it was 4.52 seconds with STD of 0.21 seconds.

Figure 6.8 shows visual samples from my experiment results while figure 6.9 shows charts of the RMSE error and the time required for different groups of images.

Since images of the same patient should report the same measurements, e.g. the cochlea length from MR image and CT image of the same patient. I define the difference as In-Patient-Error (IPE). This is an error produced by different scanners. IPE is computed for all measurements of three methods: ACA2020, automatic and manual A-value. In figure 6.10, a comparison of IPE between the manual and the automatic A-value methods is shown.

The measurement results include the average of the estimated scala vestibuli[14] length and the volume size, the average of the estimated scala tympani volume size and different lengths, and the A-value. The length measurements include a length that passes by the center of the scala (central length). Additional lengths are recorded for the scala tympani as it is more of the interest due to the fact that the cochlea implant is implanted in this part. These lengths passes by the scala

[14]Scala vestibule is actually the combined scala media and scala vestibule.

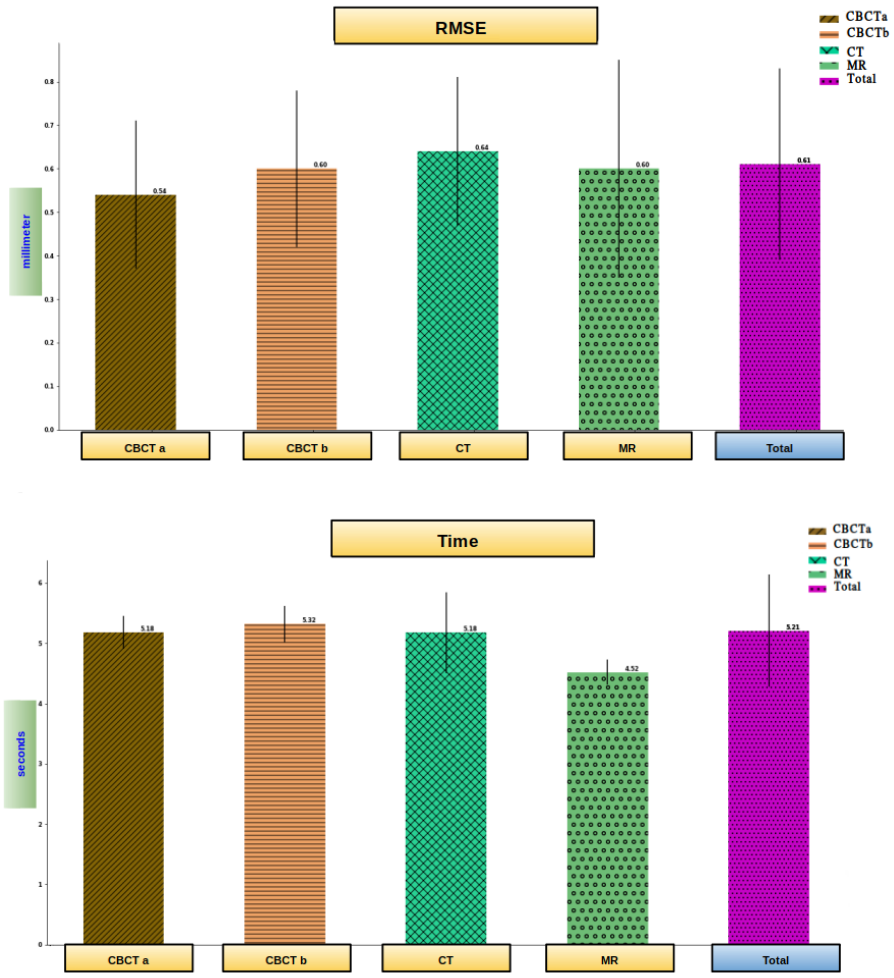


Figure 6.9: Top: average RMSE values of the registration for different groups. Bottom: average time required to perform the analysis per image for each group.

lateral wall and by the organ of Corti, see figure 1.3, figure 5.9 and figure 5.10. They are computed using three methods. These methods are my two proposed ACA2020 and automatic A-value methods and the manual A-value method done by the experts.

In table 6.8, and table 6.11, the cochlea's related measurements and their IPE values are listed from all the images. In table 6.9, table 6.10, table 6.12, and table 6.13 more details are provided for each dataset separately. Finally, in table 6.14, a comparison between the three methods is shown using only images available in both manual and automatic experiments from all datasets.

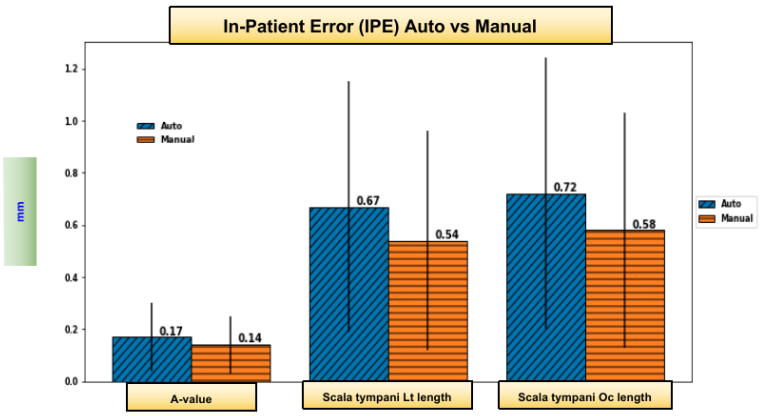


Figure 6.10: A-value method: Automatic vs Manual. In-patient error in mm. Lt and Oc are short for lateral wall and organ of Corti. MRI images are not included in the manual experiments.

Table 6.8: ACA2020 measurements with related errors (all datasets).

	Scala vestibuli volume size (mm^3)	Scala tympani volume size (mm^3)	Scala vestibuli center length (mm)	Scala tympani center length (mm)
Mean	41.69	48.57	35.95	30.06
STD	2.0	2.24	0.53	0.45
Min	34.98	38.98	35.21	29.53
Max	49.3	57.67	38.79	32.21
IPE	1.79	1.91	0.61	0.52
σ_{IPE}	1.26	1.3	0.47	0.4

The work of Iyaniwura et al. [IELA18] uses the same idea of cropping and points-model as I proposed in [ADBP+18b] using a points-model of the A-value end-points. However, my work is tested against realistic clinical images while their work was tested on only high resolution μCT images. Moreover, they didn't report how much time it takes to obtain the measurement. Since they are using a complicated 3-stage image registration pipeline, I think their method takes longer comparing to the methods I proposed. Finally, I showed how my method performs on different datasets and all the details are provided in the public source-code.

The number of reported experiments are different between the manual method and the automatic method as these two experiments are done independently. Some

Table 6.9: ACA2020 measurements (Germany dataset).

	Scala vestibuli volume size (mm^3)	Scala tympani volume size (mm^3)	Scala vestibuli center length (mm)	Scala tympani center length (mm)
Mean	41.97	49.03	36.01	30.12
STD	2.06	2.01	0.58	0.49
Min	34.98	43.05	35.21	29.53
Max	49.3	57.67	38.79	32.21

Table 6.10: ACA2020 measurements (Egypt dataset).

	Scala vestibuli volume size (mm^3)	Scala tympani volume size (mm^3)	Scala vestibuli center length (mm)	Scala tympani center length (mm)
Mean	40.72	46.97	35.72	29.85
STD	1.4	2.27	0.11	0.13
Min	36.02	38.71	35.6	29.69
Max	43.0	48.94	36.17	30.3

Table 6.11: Scala tympani length related measurements with their IPE using all and complete datasets.

Length (mm)	A-value		Scala tympani lateral wall			Scala tympani organ-of-corti		
Method	ACA2020	Manual	ACA2020	A-value auto.	A-value manual	ACA2020	A-value auto.	A-value manual
Mean	8.91	9.02	43.74	39.38	39.82	31.67	32.01	32.49
STD	0.16	0.44	0.65	0.61	1.69	0.47	0.66	1.82
Min	8.71	8.21	42.93	38.6	36.69	31.11	31.17	29.12
Max	9.57	10.06	47.19	42.92	45.91	34.08	34.75	39.05
IPE mean	0.17	0.14	0.73	0.67	0.54	0.53	0.72	0.58
IPE STD	0.13	0.11	0.57	0.48	0.42	0.41	0.52	0.45

images are found on the manual experiments but not in the automatic experiments and vice versa. The manual experiments use 149 images[15] while the number of automatic experiments was 217 [16]. The table 6.8 uses the complete reported experiments of each method while table 6.14 uses only the results of the same images found on both reported experiments (108 experiments).

The results show that automatic A-value method produces results similar to the ones produced by the A-value manual method. The IPE difference between them is less than 0.01 mm. All average lengths were also close to each other, but it has been noticed that the standard deviation was a little larger in the case of the manual experiments. Since these manual experiments are done only by two experts, the results are may be affected by some human error during locating the A-value points.

There was no large difference between measurement results of the datasets from Germany or Egypt. The averages from the automatic and manual method experiments were close in both datasets. This probably results from the small number of patients.

[15]Only 115 images are reported. 34 images are removed as they are too noisy to find the landmark locations.

[16]Recall that some images have two cochleae.

Table 6.12: Scala tympani length related measurements (Germany dataset).

Length (mm)	A-value		Scala tympani lateral wall			Scala tympani organ-of-corti		
Method	ACA2020	Manual	ACA2020	A-value auto.	A-value manual	ACA2020	A-value auto.	A-value manual
Mean	8.91	8.97	43.84	39.4	39.63	31.75	32.03	32.28
STD	0.15	0.39	0.7	0.6	1.49	0.51	0.64	1.6
Min	8.71	8.21	42.93	38.6	36.69	31.11	31.17	29.12
Max	9.38	9.8	47.19	41.19	42.81	34.08	33.97	35.7

Table 6.13: Scala tympani length related measurements (Egypt dataset).

Length (mm)	A-value		Scala tympani lateral wall			Scala tympani organ-of-corti		
Method	ACA2020	Manual	ACA2020	A-value auto.	A-value manual	ACA2020	A-value auto.	A-value manual
Mean	8.89	9.09	43.4	39.32	40.06	31.42	31.95	32.75
STD	0.17	0.51	0.13	0.65	1.95	0.12	0.7	2.1
Min	8.81	8.25	43.11	38.99	36.85	31.18	31.59	29.28
Max	9.57	10.6	43.97	41.92	45.91	31.82	34.75	39.05

Table 6.14: Scala tympani related lengths using the same images from both datasets (108 images).

Length (mm)	A-value		Scala tympani lateral wall			Scala tympani organ-of-corti		
Method	ACA2020	Manual	ACA2020	A-value auto.	A-value manual	ACA2020	A-value auto.	A-value manual
Mean	8.96	9.02	43.88	39.58	39.82	31.77	32.23	32.49
STD	0.17	0.45	0.74	0.67	1.72	0.54	0.72	1.86
Min	8.75	8.21	42.93	38.75	36.69	31.11	31.34	29.12
Max	9.57	10.6	47.19	41.92	45.91	34.08	34.75	39.05

However, there was a large difference in the tympani lateral wall length from ACA2020 comparing to the A-value (about 4 mm). I think the lateral wall length from A-value does not cover the complete scala tympani as the ACA2020 length provides a way to visualise the length and show the curve end-points.

The cochlea scala tympani volume size ranged from 38.98 mm^3 to 57.67 mm^3. The combined scala media and scala vestibuli volume size ranged from 34.98 mm^3 to 49.3 mm^3. The overall volume size of the cochlea should range from 73.96 mm^3 to 106.97 mm^3.

The lateral wall length of scala tympani ranged from 42.93 mm to 47.19 mm. The organ-of-Corti length of scala tympani ranged from 31.11 mm to 34.08 mm. Using the A-value method, the lateral wall length of scala tympani ranged from 36.69 mm to 45.91 mm. The organ-of-Corti length of scala tympani ranged from 29.12 mm to 39.05 mm.

The results are validated using standard statistical bootstrapping and tests [DCRB19]. The figure 6.11 shows the validated results of different measurements using 1000 bootstrapping. This ensures that the result means are good estimates of the population means.

The statistical tests[17] show that there is no difference between the results from automatic A-value method and the manual A-value method (p-value = 0.42). There is no difference also between length's measurements of the left and the right ear sides (p-value > 0.16).

Comparing the results from German and Egypt dataset shows there is no difference when using manual or automatic A-value methods (p-value > 0.20). However, there is a significant difference when using ACA2000 method between the German and the Egyptian results (p-value < 0.001). It seems ACA2020 provides more accurate measurements that capture small differences between different persons.

[17] In this case, paired t-test and independent t-test.

Bootstrapping 1000

Statistics	A-value (mm)		Scala tympani lateral wall (mm)			Scala tympani organ of corti (mm)			Scala tympani volumel (mm^3)	Scala vestibuli volume (mm^3)	Scala tympani centeral length (mm)	Scala vestibuli centeral length (mm)
	A-value Manual	A-value Auto	ACA2020	A-value Manual	A-value Auto	ACA2020	A-value Manual	A-value Auto	ACA2020			
Mean	9.019	8.908	43.740	39.804	39.375	31.673	32.470	32.008	48.551	41.683	30.060	35.943
STD	0.444	0.157	0.644	1.715	0.609	0.472	1.848	0.656	2.257	2.006	0.453	0.527
Min	8.213	8.707	42.927	36.693	38.6	31.111	29.117	31.172	38.708	34.979	29.529	35.214
Max	10.6	9.567	47.185	45.906	41.919	34.075	39.046	34.749	57.667	49.299	32.212	38.786

Figure 6.11: Cochlea's measurements using 1000 bootstrapping.

The length from ACA2020 method can be visualised and has a well-defined end-points. The ACA2020 method works on different modalities and different images despite the noise level or the resolution. In the other hand, the A-value method works neither on MRI[18] nor on noisy images. Hence, ACA2020 method may provide more reliable and accurate measurement than the A-value method.

[18] A-value method is usually not used with MRI images as it has accuracy issues.

Chapter 7

Generalisation: Spine Image Segmentation and Analysis

Using ACIR2017, I was able to solve spine segmentation, spine ligaments insertion and origin points detection [ADBP18a, ADBKP19] problems. These problems are challenging and they have many applications, e.g. in bio-mechanical simulation.

7.1 Spine Segmentation

Multi-modal Spine Segmentation can be solved using my proposed method ACIR2017 [ADBP18a]. I used clinical CT and MRI images of 12 different patients. These images are anonymised to protect patients' privacy. All the images have the same height and width, i.e. 512×512 pixels with varied depths and resolutions. The depth ranges from 53 to 1050 slices and the resolution ranges from 0.7 × 0.7 × 1.0 mm to 0.9× 0.9 × 2.0 mm. A sample of these images are shown in figure 7.2, figure 7.3, and figure 7.4.

These images are part of our proposed public cervical spine dataset (CSD) [1]. The number of images is small, but it keeps increasing.

Similar to ACA2018 and ACA2020, the spine segmentation problem can be solved using an atlas-model-based approach if an efficient image registration method is found for these images. I proposed an atlas-based segmentation method using ACIR2017 [ADBP18a] for solving the cervical spine segmentation problem. To prepare the atlas model, a CT image is selected from our dataset and cropped to the cervical spine area. The cervical spine is manually segmented with different colors as in figure 7.1.

After that, each vertebra of the cervical spine is located manually then cropped and saved separately, i.e. C1 to C7 in different files with their segmentation's.

[1] Images can be obtained via an email request to bauer@uni-koblenz.de

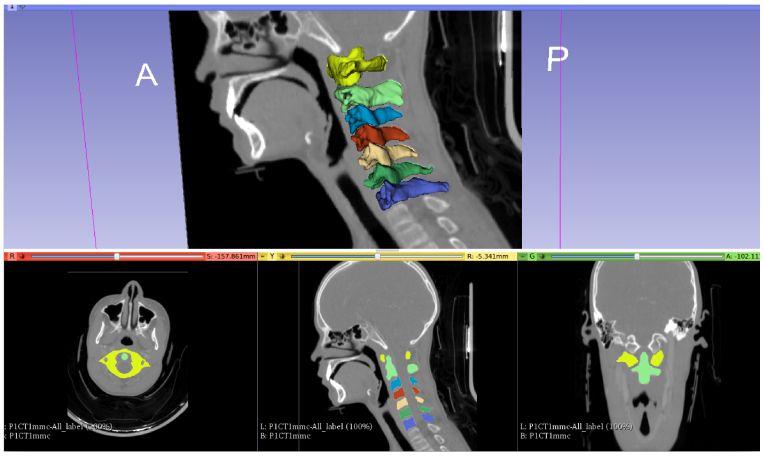


Figure 7.1: Cervical spine atlas model. The colored vertebrae are C1 to C7 (from top to bottom)

These saved images will be used in the next stage as an atlas model.

To segment a new image, C2 vertebra is located. Then the approximated locations of other vertebrae are predicted automatically. This prediction is based on a vertical distance of 15 mm between each vertebra in the sagittal view. These location points are used for cropping each vertebra with a cube of size 90 $\times$ 80 $\times$ 60 mm. The accuracy of the predicted points is not important if a vertebra still included in the cropping cube. The previous measurements are based on my experiments and they are sufficient for this purpose. Sometimes a small manual correction is needed to correct these locations which are still offered by the 3D Slicer plugin's [5, 29] friendly user interface I developed.

All cervical spine vertebrae are extracted by the cropping process above and registered using ACIR2017 to the related atlas model automatically. The result of the registrations is 3D rigid transforms' parameters. These transforms are used to transform the segmentation of the atlas model to the input image.

For all the tested images, the proposed method produced a visually satisfying segmentation in a few seconds, see figure 7.2 and figure 7.3 for result samples.

The results still suffer from a bad dice measurement and can not compete to the state of the art method such as deep learning. However, if good non-rigid transformation parameters are found and used as a second stage registration, this method may provide more accurate results and faster performance than other methods.

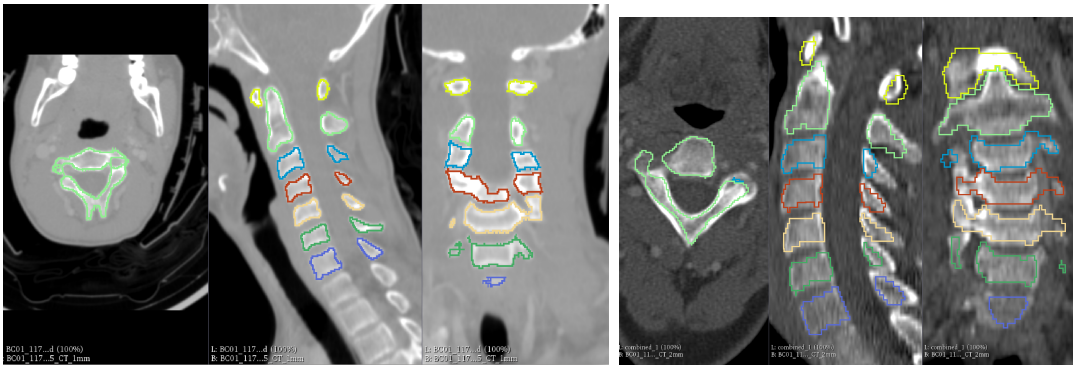


Figure 7.2: Sample results: axial, sagittal and coronal slices are shown of two CT scans of two different patients. Different vertebrae are segmented and represented in different colors.

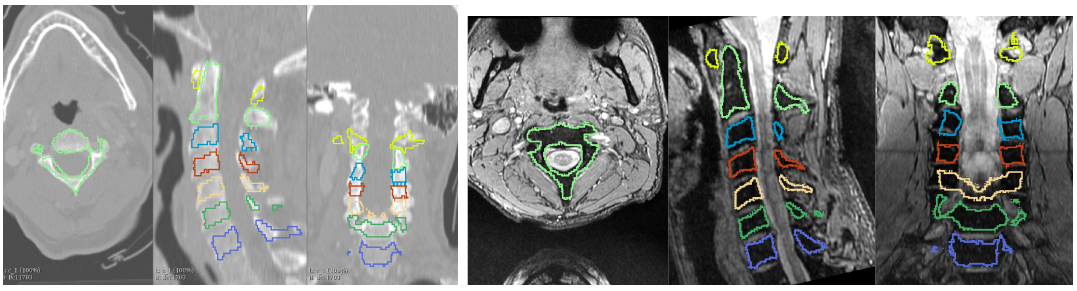


Figure 7.3: Sample results: axial, sagittal and coronal slices are shown. Left: CT image, right: MRI image. Different vertebrae are segmented and represented in different colors.

7.2 Spine Points Detection

Based on the above work [ADBP18a], I proposed a fast method to detect all Cervical Spine Origin and Insertion Points (CSOIP) automatically using the spine atlas-based segmentation [ADBKP19]. The method transforms a ligament points-model to a binary image, then this image is transformed using the transformation found by ACIR2017. Finally, the points are extracted from the transformed image. The detection rate was based on a number of points detected as there were missing points after the registration. The missing points were found due to the conversion process between the images and the points as a built-in linear interpolation was used[2].

The method above is improved by a second stage non-rigid registration using B-spline to cover the shape variance between different patients, similar to ACA2020. Seven iterations for the second stage registration are used to ensure a realistic

[2]It seems there is a bug or error when using Elastix with these points.

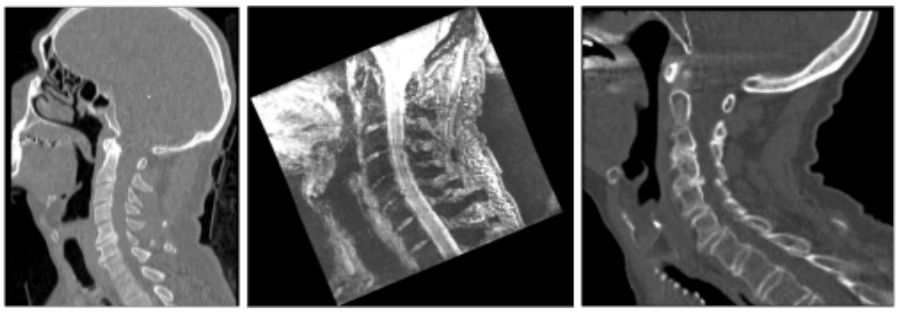


Figure 7.4: 3D cervical spine datasets' samples. Left to right: CT and MRI from CSD, CT from Verse.

deformation. The points-image conversion is removed so the transformation is applied directly to the points-model which produces no missing points. Finally, an additional points-model is added, i.e the muscle-points. This makes it possible to detect both ligament and muscle points automatically at the same time.

The method is tested against two multi-modal public and standard datasets. The first dataset is CSD which is the same as the dataset used in [ADBP18a]. It contains 8 CT and 13 MRI 3D cervical spine images of 21 patients. All MRI images have a larger size than CT images which affect the time required for the image registration process. The second dataset is part of the Large Scale Vertebrae Segmentation Challenge (Verse19) [SRK$^+$18] [1]. It contains 12 3D CT images of 12 patients.

The total number of 3D cervical spine multi-modal images from both datasets is 33. These images represent 33 patients. All patient information is already removed to protect their privacy. Three samples are shown in figure 7.4.

The following are the steps of the proposed method:

1. Localisation: the user locates a vertebra center approximately by clicking on any point around the vertebra's centre in an input image. This location is used only for cropping, explained in the next step.

2. Cropping: the point from the localisation is used as the center of a cropping box with dimensions (80, 80, 50)mm which includes the vertebra part. This vertebra is saved in a new smaller image which allows faster computation and more accurate results.

3. Registration (rigid): the cropped image is used as a fixed image in ACIR. The moving image is a pre-defined image that has segmentation and points-models. $ACIR$ finds the rigid transformation parameters that align the moving image to the fixed image. The transformation is applied on the cropped moving image to produce the cropped registered image.

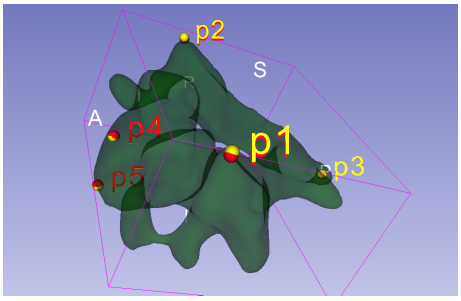


Figure 7.5: Sample of landmarks used for validation, the figure shows C5 five landmarks. Red: original landmarks. Yellow: a sample result.

4. Registration (non-rigid): the cropped registered image is used as a moving image in the second stage registration. The proposed method finds the non-rigid transformation parameters that align the cropped registered image to the fixed image. The rigid and non-rigid transformations are composed to produce the final transformation.

5. Transformation: the two points-models, i.e. ligament-points and muscle-points, are transformed using the final transformation to the input image.

The main tools used for implementation are the standard open-source 3D Slicer which is a standard medical image analysis and visualization tool and Elastix which is a popular standard medical image registration tool. The method and the datasets are available for a free download from a public server.

Using the transformation directly to transform the points instead of transforming points-image allows preserving of these points with their correct labels after the transformation. Hence, there are no missing points when using the improved method. Another advantage of this method is detecting the muscle-points in addition to the ligament-points. It also addresses the difference in size and shape by using the non-rigid B-spline transform. To the best of my knowledge, there is no method available to detect CSOIP automatically. This area of research is very nascent with no published research by experts.

Due to the lack of ground truth from experts, five 3D points landmarks that cover the vertebra shape are located in all the tested images, see figure 7.5. The RMSE of these landmarks has been measured after applying the same transformation.

The average RMSE was 2.058 mm with 0.786 mm standard deviation. Compared to a vertebra size this error is small, hence these points can be used in different simulation applications to produce approximated results. For more accuracy, users can manually correct the points' location in a few seconds using the

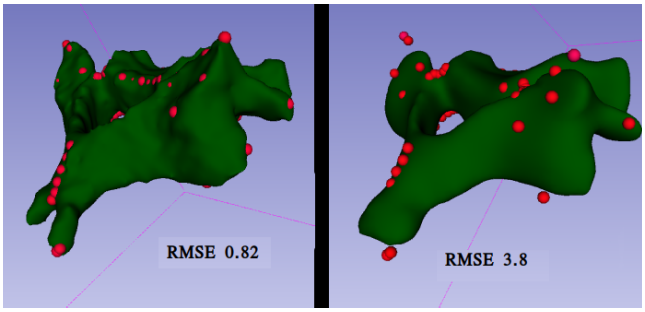


Figure 7.6: C5 vertebra. MSE results in mm vs visual results. Left: CT from CSD dataset. Right: CT from Verse dataset.

Table 7.1: Spine points detection RMSE mean and standard deviation of each group of images.

	RMSE result (mm)		**Time results (seconds)**	
	Mean	**STD**	**Mean**	**STD**
CSD CT	2.052	1.236	8.162	2.043
Verse CT	2.227	0.628	8.904	2.203
CSD MR	1.907	0.584	25.636	1.686
Total	2.058	0.786	15.316	8.667

friendly Slicer user-interface. The average time to detect CSOIP for a vertebra was 15.3 seconds with 8.667 seconds standard deviation.

The quality of the results is also affected by the localisation step. Experimentation with different locations often yields the desired results. In table 7.1, the average and the standard deviation of the landmarks RMSE and the time required to detect CSOIP of a vertebra are reported for the three groups of images.

In figure 7.6, the ground truth segmentation and the detected points are shown. The left side shows the results of a CT image from the CSD dataset. Even though the RMSE is 0.82 mm, all the points are detected correctly. The right side shows a high error of 3.8 mm but most of the points are detected correctly except a few ones need correction. This correction can be done manually in 3D Slicer within a few seconds.

The results can be improved if the problem is reduced to find these points using segmentation or a mesh instead of medical image. In this case, the similarity metric mutual information S_{MMI} is replaced by the mean squared error metric S_{MSE}, and the KNN interpolator is used to avoid introducing new colour values. I achieved[3] an average dice for of 92% for all the spine vertebrae. In figure 7.7 sample results

[3]This is part of a master thesis of Li Ying Yin Simon which I co-supervised. The related paper is submitted.

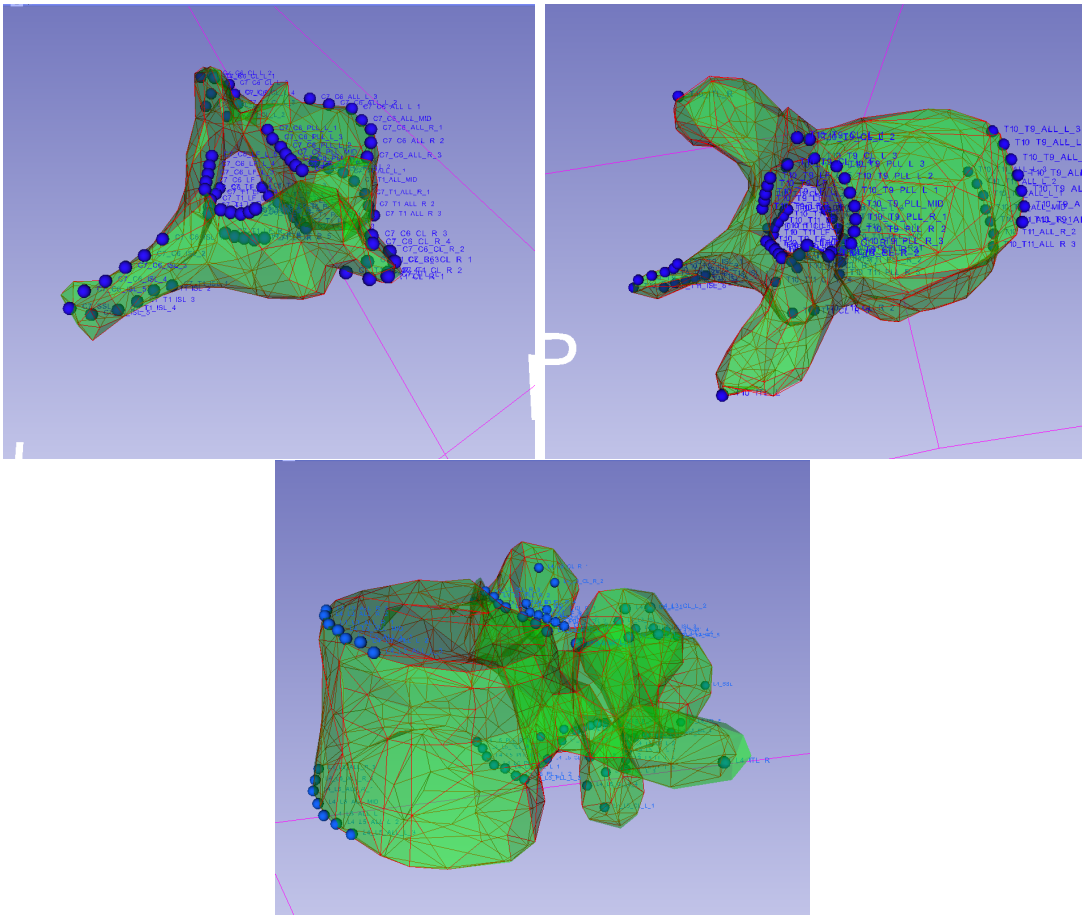


Figure 7.7: 3D vertebra alignment and ligament points detection of different vertebrae C7, T10, and L4. Red colour is the atlas, green colour shows the input vertebra, the blue colour represents the ligament insertion, origin points and their labels.

of three different vertebrae C7, T10, and L4 are shown from different angles.

Chapter 8

Conclusion and Future Works

In this study, I provided a practical solution for solving clinical multi-modal 3D cochlea image registration, fusion, segmentation and analysis problems.

These problems have important applications related to cochlea implant surgery and cochlea research. The study was part of a funded project COMBS[1] under grant number 5141056. The methods were tested and worked successfully on CBCT, CT, and MRI. The methods have been validated using two datasets from Germany and Egypt.

Since this a dissertation of the medical image analysis field, I concentrated on providing a practical solution rather than improving the current algorithms. Image registration and segmentation are challenging problems and usually one could work on a branch of one of them in PhD research.

Moreover, working with very popular and busy doctors has some disadvantages as the process of getting knowledge or some ground-truth was very slow and work has to be repeated many times. In addition to that, there was no standard dataset available, so I have to prepare one from scratch. I could not find any public code or dataset related to the cochlea literature, so I could not validate the other methods or benefit from their results. It seems this field is dominated by cochlea implant's producers, hence, neither their code nor datasets are publicly available. This study provides a free public dataset and an open-source code which may help the future of cochlea's research field.

The anatomy of the ear and the cochlea was introduced. The problem definition and the dissertation hypothesis were explained. General information about medical images and the concept of different technologies were introduced in some details. The mathematical concepts of image registration and its components such as mutual information similarity metric, adaptive stochastic gradient descent, and B-spline transformation were introduced.

[1]The project period was from 15.4.2015-31.7.2017 by Cochlear linited, Germany.

I could not find any detailed example in the literature about image registration, so I implemented one from all the basic components where I described all the related mathematics and how to implement it, e.g. the derivative of the similarity metric with respect to the transformation parameters. This may help newcomers to image registration field as the available implementation e.g. in ITK is too complicated to trace due to the concentrate on performance rather than code-readability.

Different image segmentation algorithms were explained such as threshold, edge detection, region growing, and level-sets. The result of applying these classical methods on cochlea images was shown as well. I mentioned all important related work to the cochlea problems, i.e. cochlea image registration, cochlea image segmentation, and cochlea image analysis. I described what they did, what they missed and what should be done.

My proposed methods for cochlea image registration, fusion, segmentation and analysis were explained in details with their historical development e.g. ACIR2016, ACIR2017, ACA2018, ACIR2020, and ACIR2020. Their results were discussed in details, sample figures, charts, and tables were shown. The results were compared to the state of the arts optimisers for medical image registration e.g. fast adaptive stochastic gradient descent [QvLS16] and efficient preconditioned for stochastic gradient descent [QLS19].

ACIR2016 was my first attempt to automate cochlea image registration. The total RMSE median of ACIR2016 was 9.52 mm with range [0.22-49.78] mm. The total time average required for registration of an image pair using this method was 16.78 seconds. This method was fully automated and work directly on the input images without pre-processing e.g. cropping. However, the method was complicated, needs about large time to register a pair of images and the accuracy was not great at around 11 mm RMSE.

To address the above issues, ACIR2017 and ACIR2020 were proposed. They are less complicated as the multi-registration and multi-resolution parts were removed but a manual step of locating the cochlea and a pre-processing step i.e. cropping were added. However, the results were more accurate and the required times was much less. Total RMSE average of ACIR2017 was 0.36 mm. The total time average required for registration of an image pair using this method was 4.65 seconds. The total RMSE median of ACIR2020 was 0.35 mm with range [0.07-2.13] mm. Using ACIR2020, the results were almost the same i.e. the total time average required for registration of an image pair was 4.62 seconds. These two methods are equivalent one of them can be used to get practical results at least for the tested images, see figure 8.1 and figure 8.2.

Using ACIR2017, and ACIR2020, real-time automatic methods for clinical

multi-modal 3D cochlear image segmentation and analysis is proposed. The proposed methods can be integrated into a surgical cochlear robot due to high accuracy and fast performance. Due to the absence of public ground truth, the error is estimated using the active length of the cochlear implant electrodes in the images in addition to landmarks located by experts. These landmarks represent the round window and the cochlea apex which usually they can be seen in the clinical cochlea images.

To the best of my knowledge, this is the first method that uses such error estimation in cochlea research. The experiments showed a small error in the length estimation. This error ranged from 0.15 mm to 1.1 mm using ACA2018.

Iyaniwura et al. [IELA18] used my ideas, e.g. cropping and points-model, from [ADBP$^+$18b] but unfortunately no citation is given [2]. They replaced the points-model of scala tympani by points-model of the A-value end-points. However, their work was tested on only high resolution μCT images. They didn't report the time which I think it is slower than my method due to the complicated 3 stages image registration pipeline they used.

I defined IPE which is the difference between patient measurements from different images e.g. the difference in length of scala vestibuli from MRI image and from CBCT image. The results show that automatic A-value, which is a part of ACA2020 I proposed, produces results similar to the one produced manually by the experts. IPE difference between them is 0.03 mm. All average lengths were also close to each other, but it has been noticed that the standard deviation was a little larger in the case of the manual results. Since these manual experiments are done only by two experts, the results are may be affected by some human error during locating the A-value points.

There was no large difference between the measurement results of the datasets from Germany or Egypt. The averages from the automatic and manual method experiments were close in both datasets. This probably results from the small number of patients (67 patients in total, 41 from German dataset and 26 from Egyptian dataset). One may argue that there is no large difference between the volume size and the length of different patients' cochlea. However, this needs strong proof from the ground truth e.g. using large dataset of μCT.

However, there was a large difference in the tympani lateral wall length from ACA2020 comparing to the A-value (about 4 mm). I think the lateral wall length from A-value does not cover the complete scala tympani as the ACA2020 length provides a way to visualise the length and show the curve end-points.

From all 217 images of 67 patients, the cochlea scala tympani volume size ranged from 38.98 mm^3 to 57.67 mm^3. The combined scala media and scala

[2]Moreover, despite the source-code is publicly available, many recent papers didn't compare their results to mine which shows how competitive the research in this field.

vestibuli volume size ranged from 34.98 mm^3 to 49.3 mm^3. The overall volume size of the cochlea should range from 73.96 mm^3 to 106.97 mm^3.

The lateral wall length of scala tympani ranged from 42.93 mm to 47.19 mm. The organ-of-Corti length of scala tympani ranged from 31.11 mm to 34.08 mm. Using the A-value method, the lateral wall length of scala tympani ranged from 36.69 mm to 45.91 mm. The organ-of-Corti length of scala tympani ranged from 29.12 mm to 39.05 mm.

The length from ACA2020 method can be visualised and has a well defined end points. The ACA2020 method works on different modalities and different images despite the noise level or the resolution. In the other hand, the A-value method does not work on MRI nor noisy images. Hence, ACA2020 method may provide a more reliable and accurate measurement than the A-value method.

The experiments are repeated 3 times for justification. I validated the results using bootstrapping and standard statistical tests. The proposed method is implemented as a public open-source plug-in for 3D Slicer software so other researchers can reproduce my results.

Future works include more enhancement in terms of speed and accuracy. Using a better histological model to get a segmentation of the 3 cochlear scalae also should be considered in future research.

It is also interesting to explore what deep learning and statistical shape models offer. To improve the ground truth, more experts should be involved to provide more landmarks and manual segmentation for evaluation.

Another idea is to test these methods using ground truth from higher resolution e.g. μCT. The idea is to have the same cochlea scanned by μCT, CT, CBCT, and MRI. If we have a number of cochlea, we can use the μCT for validation as we can get more accurate measurements. Moreover, we can train a deep learning network to convert from low-resolution e.g. MRI, and CBCT to the higher one i.e. μCT which gives the doctors better visualisation of different part of the cochlea. It would be interesting also to add more datasets from different geographical locations.

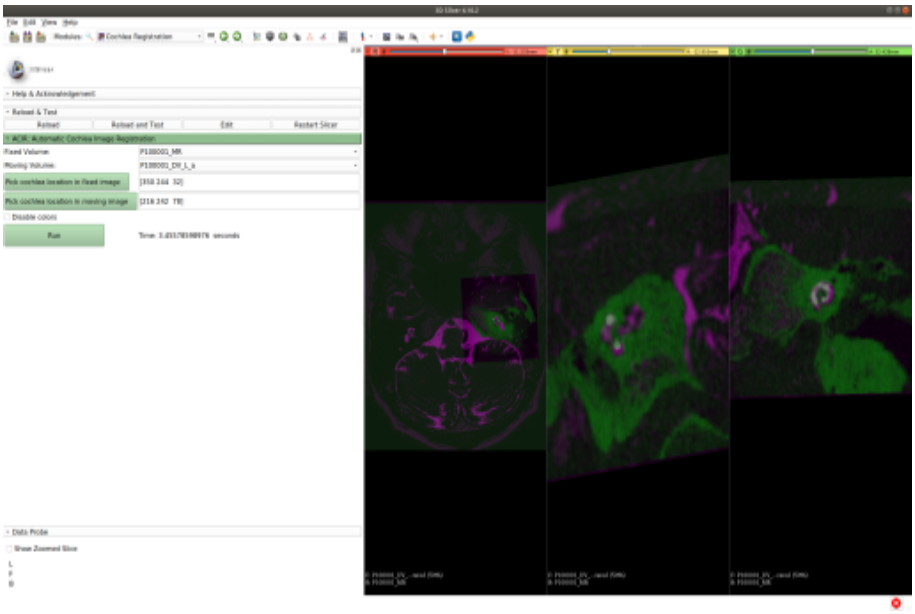


Figure 8.1: Slicer cochlea registration and fusion interface. A CBCT image with cochlea implant is registered to a MRI image of the same patient in 3.4 seconds. One can see the cochlea implant and details from both images in different colours (green for CBCT and magenta for MRI).

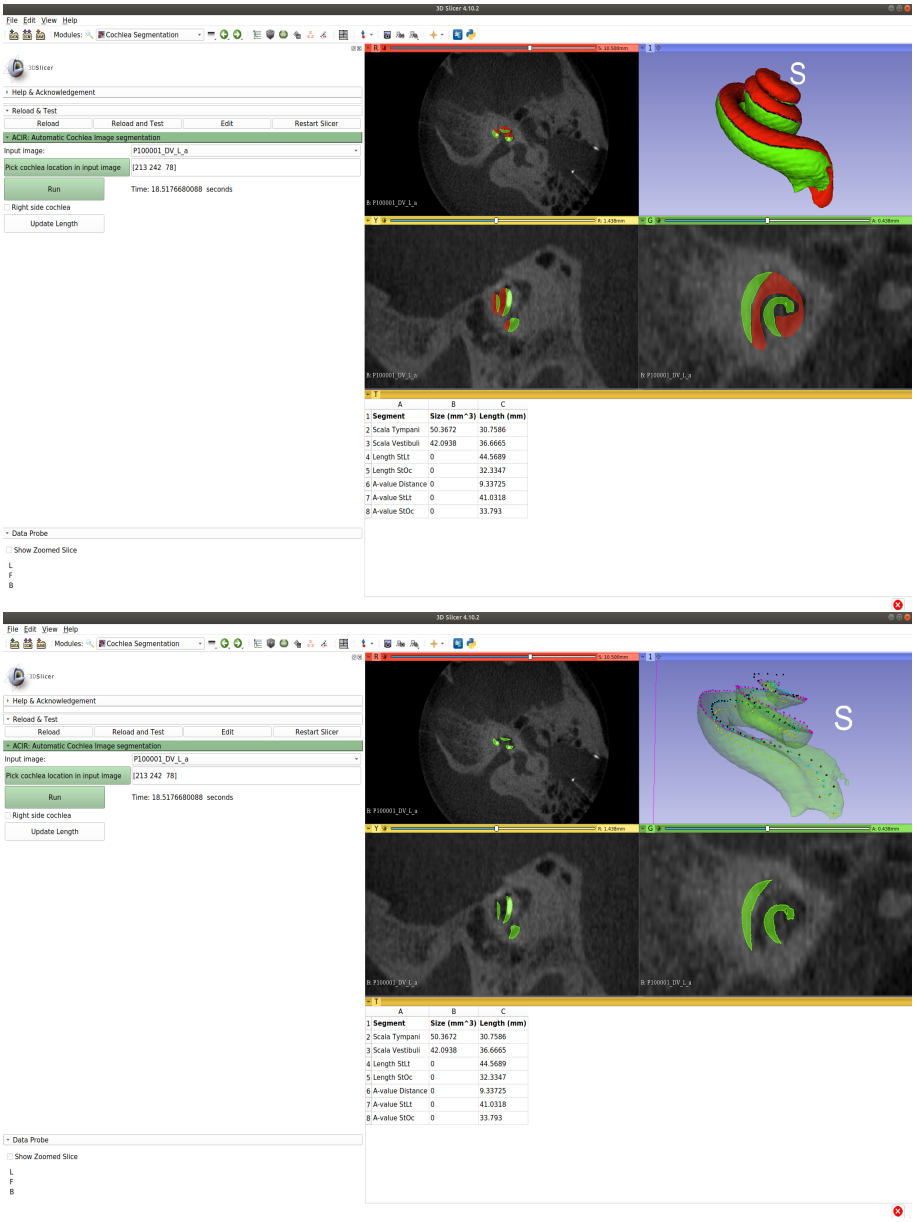


Figure 8.2: Slicer cochlea segmentation and analysis interface. The figure shows a CBCT image with a cochlea implant. One can see the implant inside the scala tympani which is represented by green colour. All related measurements are presented in addition to the 3D mesh representation. The bottom figure shows a visualisation of different points-model. Slicer provides a flexible way to visualize different parts of the image.

List of Tables

List of Figures

List of Abbreviations

SGD$_{RM}$ Robbins-Monro Stochastic Gradient Descent. 44

***μ*CT** Micro Computed Tomography. 3, 4, 10, 15, 49, 54, 58, 65–68, 86, 88, 110, 125, 126

2D two-dimensional. 5, 6, 8, 10, 20, 22, 25–27, 32, 34, 36, 38, 58, 76, 77, 131

3D three-dimensional. 4, 5, 7–16, 25, 26, 32, 42, 49, 51, 58, 67, 68, 71–75, 83, 85, 87, 88, 102, 106–108, 116, 118, 119, 121, 123, 125, 128, 131, 132

4D four-dimensional. 5

ASGD Adaptive Stochastic Gradient Descent. 15, 42, 44, 74, 75, 79, 80, 82–84, 90, 95, 97, 98, 107

ASM Active Shapes Models. 54, 58–60, 63–65, 132

BFGS Broyden-Fletcher-Goldfarb-Shanno. 44, 84

CBCT Cone-beam Computed Tomography. 5, 9–13, 15, 38, 58–61, 63, 66–68, 71–74, 82, 86, 88, 95, 98, 99, 104, 107, 123, 125–128, 132

CDL Cochlear Duct Length. 3, 66, 67

CI Cochlea Implant. 2, 3, 5, 19, 58, 72, 73, 88, 89, 99–101, 103, 105, 107, 129

CNN convolutional neural network. 67

CSD cervical spine dataset. 115, 118, 120

CSOIP Cervical Spine Origin and Insertion Points. 117, 119, 120

CT Computed Tomography. 5, 9–15, 19, 38, 45, 57, 59, 60, 62–68, 71–74, 82, 95, 98, 99, 104, 107, 108, 115, 117, 118, 120, 123, 126, 132

DICOM Digital Imaging and Communications in Medicine. 5, 6, 71, 72

ENT ears, nose and throat. 10

FASGD Fast Adaptive Stochastic Gradient Descent. 42, 95

fMRI functional MRI. 5

FPSGD Fast Preconditioned Stochastic Gradient Descent. 44, 45, 95

GAC Geodesic Active Contours. 53, 54, 59, 60, 64, 65, 132

GD Gradient Descent. 20, 21, 43, 44

GVF Gradient Vector Flow. 52, 53, 131

HCD Human Cochlea Dataset. xi, 15, 71–73, 102

IJK image coordinate system. 8

IPE In-Patient-Error. 108, 109, 111, 125, 129, 132

ITK The Insight Toolkit. 7, 34, 58, 86, 124

KNN K-Nearest Neighbour. 23, 120

LBFGS Limited memory BFGS. 44, 84, 86

LoG Lablacian of Gaussian. 50

MI Mutual Information. 38, 45

MITK Medical Image Interaction Toolkit. 7, 58

MMI Mattes's Mutual Information. 15, 41

MR magnetic resonance. 3, 12, 14, 15, 57, 71–74, 82, 95, 98, 99, 104, 107, 108, 120, 132

MRI Magnetic Resonance Imaging. 5, 10–14, 19, 38, 59, 60, 63, 65–68, 72, 82, 104, 110, 113, 115, 117, 118, 123, 125–127, 131, 132

MSE Mean Square Error. 36, 120

NCC Normalized Cross Correlation. 66

NN Nearest Neighbour. 89

OOP Object-Oriented Programming. 7

PCA Principal Component Analysis. 54

PCI Percutaneous Cochlear Implantation. 57

PDF Probability Density Functions. 39

RAS Right Anterior Superior. 8, 72

RMSE Root Mean Squared Error. 95–100, 107–109, 119, 120, 124, 129

ROI Region Of Interest. 16, 51, 83

S-LBFGS Stochastic Quasi-Newton. 43, 84–86, 107

SDF Signed Distance Function. 53

SGD Stochastic Gradient Descent. 41, 42, 44, 45

STD Standard Deviation. 96, 97, 107, 108, 110–112, 120

StLt Scala tympani lateral points model. 90

StOC Scala tympani organ of Corti points model. 91

StPt Scala tympani points model. 90

SvPt Scala vestibuli points model. 90

VR Value Representation. 6

VTK The Visualization Toolkit. 7, 58

List of Symbols

$*$ A convolution operator. 30

$\boldsymbol{A_X}$ A sample set from the image X. 40

$\boldsymbol{B}$ A Bernstein polynomial or a Bernstein matrix. 28, 30

$\boldsymbol{B_X}$ A sample set from the image X. 39

$\boldsymbol{C}$ A Bézier curve points vector. 30

$\boldsymbol{E_X}$ An estimated expected value of the image. 39

$\boldsymbol{E_{GVF}}$ GVF energy. 52

$\boldsymbol{E_{Snake}}$ Total energy. 52

$\boldsymbol{G}$ The Gaussian function. 40

$\boldsymbol{H}$ An approximation of Hessian matrix. 85, 86

$\boldsymbol{H_k}$ The Hessian matrix. 43

$\boldsymbol{I}$ The Identity matrix. 43

$\boldsymbol{I_F}$ Fixed image. 19, 20, 40, 42, 46, 75, 76, 83

$\boldsymbol{I_M}$ Moving image. 19, 20, 40, 42, 46, 75, 76, 83

$\boldsymbol{J_\mu}$ Jacobian matrix of the transformation. 21, 27, 37, 40

$\boldsymbol{K}$ A knots vector. 33

$\boldsymbol{L}$ Number of iteration. 85, 86

$\boldsymbol{L_F}$ Pixel values samples from fixed image. 76, 78

$\boldsymbol{L_M}$ Pixel values samples from transformed moving image. 76, 78

$\boldsymbol{M}$ The preconditioned matrix. 44, 45

$\boldsymbol{N}$ Number of samples. 76

$\boldsymbol{N_A}$ Length of samples. 39, 40

$\boldsymbol{N_B}$ Length of samples. 39

$\boldsymbol{O(n)}$ In the order of n. 84

$\boldsymbol{O(n^2)}$ In the order of n^2. 84

$\boldsymbol{P}$ Points. 19, 20, 25, 36, 37, 80

$\boldsymbol{P_i}$ A control point. 33

$\boldsymbol{Pr(x)}$ Marginal probabilities in x. 75, 76

$\boldsymbol{Pr(y)}$ Marginal probabilities in y. 75

$\boldsymbol{S}$ Similarity metric function. 21

$\boldsymbol{S_{MI}}$ Similarity metric based on Mutual Information. 75

$\boldsymbol{S_{MI}}$ Mutual Information. 20

$\boldsymbol{S_{MMI}}$ Similarity metric based on Mattes' Mutual Information. 74, 75, 83, 120

$\boldsymbol{S_{MSE}}$ Similarity metric based on Mean Square Error. 74, 75, 120

$\boldsymbol{S_{MSE}}$ Mean Square Error. 20

$\boldsymbol{T}$ A transformation function. 19, 30, 37, 75, 83

$\boldsymbol{X}$ An image. 39, 40

$\boldsymbol{X}$ An image. 40

$\boldsymbol{Z}$ A set of integer numbers. 22

$\boldsymbol{\Phi}$ A function. 53

$\boldsymbol{\Sigma}$ The covariance matrix. 80

$\boldsymbol{\alpha}$ A pole value or a weight parameter. 35, 52

$\boldsymbol{\beta}$ A weight parameter. 52

$\boldsymbol{\beta_i^n}$ B-spline basis function of degree n. 33

$\binom{n}{i}$ A binomial coefficient. 28

$\boldsymbol{\delta}$ The maximum incremental displacement allowed between the optimisation's iterations. 42, 45

$\boldsymbol{\epsilon}$ Random noise. 45

$\boldsymbol{\hat{I}_F}$ Samples from I_F. 42

$\boldsymbol{\hat{I}_M}$ Samples from I_M. 42

$\boldsymbol{\lambda}$ Step-size of Gradient Descent. 21, 42, 44, 52, 79

$\boldsymbol{\lambda_k}$ Step-size at step k. 79

$\boldsymbol{\mu}$ Transformation parameters. 19–21, 25, 27, 36, 37, 45, 75, 76, 80, 83, 85, 90, 93

$\boldsymbol{\nabla}$ The energy derivative. 52

$\boldsymbol{\phi(x)}$ A interpolator. 23

$\boldsymbol{\psi}$ A standard deviation. 40

$\boldsymbol{\tau}$ An index for the location at the function. 30

$\boldsymbol{\theta}$ Degrees of angle. 26

$\boldsymbol{\tilde{\mu}}$ Optimal transformation parameters.. 20

$\boldsymbol{\tilde{p}}$ A transformed point. 25

$\boldsymbol{\tilde{x}}$ Transformed value of x or mean value of x. 27

$\boldsymbol{\tilde{y}}$ Transformed value of y. 27

$\boldsymbol{a_j}$ Coefficients for a polynomial function. 23

$\boldsymbol{c[k]}$ An output signal. 34

$\boldsymbol{c_k}$ A coefficient. 22

$\boldsymbol{f}$ A function or a signal. 30, 34, 42, 43

$\boldsymbol{f(x)}$ A function which performs function operation. 21

$\boldsymbol{f[k]}$ An input signal. 34

$\boldsymbol{g}$ A function or a signal. 30

$\boldsymbol{h}$ Width of Parzen window. 76

$\boldsymbol{i}$ Number of invalid results. 96

$\boldsymbol{k}$ Optimisation iteration or a set of integer values or length of knots vector. 21, 23, 33, 44

$\boldsymbol{l_{H_F}}$ Length of the histogram bins of the fixed image. 41

$\boldsymbol{l_{H_M}}$ Length of the histogram bins of the fixed image. 41

$\boldsymbol{n}$ Number of pixels/points in the image or number of degree. 20, 21, 23, 26, 28–31, 33, 36, 37, 78, 88

$\boldsymbol{p}$ A point. 25, 26

$\boldsymbol{p(X)}$ The estimated image probability density function. 39

$\boldsymbol{p_F}$ The joint marginal probability distributions of fixed image. 41

$\boldsymbol{p_M}$ The joint marginal probability distributions of transformed moving image. 41

$\boldsymbol{p_{i,j}}$ A control point. 31

$\boldsymbol{p_j}$ An image pixel coordinate. 21

$\boldsymbol{q}$ A number of dimension. 22

$\boldsymbol{r}$ The robustness percentage. 96

$\boldsymbol{t}$ A ratio point in Bézier curve. 28, 30

$\boldsymbol{v}$ Number of valid results. 96

$\boldsymbol{w}$ Parzen window. 76, 77

$\boldsymbol{x}$ Coordinate point x or input value. 20–23, 36, 40, 42, 43

$\boldsymbol{y}$ Coordinate point y. 20, 36

$\boldsymbol{|k|}$ Length of the set. 23

$\boldsymbol{|x|}$ The absolute value of x. 23

$\mathbf{A}_{\boldsymbol{F_i}}$ Samples of fixed image at location i. 39

$\mathbf{A}_{\boldsymbol{M_i}}$ Samples of moving image at location i. 39

Literature

[ADBKP19] Al-Dhamari, Ibraheem; Bauer, Sabine; Keller, Eva; Paulus, Dietrich: Automatic Detection of Cervical Spine Ligaments Origin and Insertion Points. In: *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, 2019, S. 48–51

[ADBP+16] Al-Dhamari, Ibraheem; Bauer, Sabine; Paulus, Dietrich; Lissek, Friedrich; Jacob, Roland: Automatic Multimodal Registration and Fusion of 3D Human Cochlea Images. In: *The 14^{TH} INTERNATIONAL CONFERENCE ON COCHLEAR IMPLANTS CI2016*. Canada: American Cochlear Implant Alliance, 2016

[ADBP+17] Al-Dhamari, Ibraheem; Bauer, Sabine; Paulus, Dietrich; Lesseck, Fredirich; Jacob, Roland; Gessler, Anna: ACIR: Automatic Cochlea Image Registration. In: *Proc. SPIE 10133, Medical Imaging 2017: Image Processing* Bd. 10133, 2017, S. 1–5

[ADBP18a] Al-Dhamari, Ibraheem; Bauer, Sabine; Paulus, Dietrich: Automatic Multi-modal Cervical Spine Image Atlas Segmentation Using Adaptive Stochastic Gradient Descent. In: Maier, Andreas (Hrsg.); Deserno, Thomas (Hrsg.); Handels, Heinz (Hrsg.); Maier-Hein, Klaus (Hrsg.); Palm, Christoph (Hrsg.); Tolxdorff, Thomas (Hrsg.): *Bildverarbeitung feur die Medizin 2018*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2018, S. 303–308

[ADBP+18b] Al-Dhamari, Ibraheem; Bauer, Sabine; Paulus, Dietrich; Helal, Rania; Lisseck, Friedrich; Jacob, Roland: Automatic Cochlear Length and Volume Size Estimation. In: Stoyanov, Danail (Hrsg.); Taylor, Zeike (Hrsg.); Sarikaya, Duygu (Hrsg.); McLeod, Jonathan (Hrsg.); González Ballester, Miguel A. (Hrsg.); Codella, Noel C. (Hrsg.); Martel, Anne (Hrsg.); Maier-Hein, Lena (Hrsg.); Malpani, Anand (Hrsg.); Zenati, Marco A. (Hrsg.); De Ribaupierre, Sandrine (Hrsg.); Xiongbiao, Luo (Hrsg.); Collins, Toby (Hrsg.); Reichl, Tobias (Hrsg.);

Drechsler, Klaus (Hrsg.); Erdt, Marius (Hrsg.); Linguraru, Marius G. (Hrsg.); Oyarzun Laura, Cristina (Hrsg.); Shekhar, Raj (Hrsg.); Wesarg, Stefan (Hrsg.); Celebi, M. E. (Hrsg.); Dana, Kristin (Hrsg.); Halpern, Allan (Hrsg.): *OR 2.0 Context-Aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis*. Cham: Springer International Publishing, 2018, S. 54–61

[ADJ15] Alexiades, George; Dhanasingh, Anandhan; Jolly, Claude: Method to estimate the complete and two-turn cochlear duct length. In: *Otology and Neurotology* 36 (2015), Nr. 5, S. 904–907

[AP.85] AP., Walby: Scala tympani measurement. In: *The Annals of otology, rhinology, and laryngology* 94 (1985), Nr. 4, S. 393–397

[Ber12] Bernstein, Sergei: Proof of the theorem of Weierstrass based on the calculus of probabilities.(translation). In: *Comm. Kharkov Math. Soc.* 13 (1912), Nr. 1, S. 1–2

[BHNS16] Byrd, Richard; Hansen, Samantha; Nocedal, Jorge; Singer, Yoram: A stochastic quasi-Newton method for large-scale optimization. In: *SIAM Journal on Optimization* 26 (2016), 1, Nr. 2, S. 1008–1031

[BJ01] Boykov, Yuri; Jolly, Marie-Pierre: Interactive graph cuts for optimal boundary region segmentation of objects in N-D images. In: *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001* Bd. 1, 2001, S. 105–112.

[BL79] Beucher, Serge; Lantuejoul, Christian: Use of Watersheds in Contour Detection. In: *Int. Workshop on image processing: Real-time Edge and Motion Detection/Estimation* (1979)

[BPR+05] Bartling, Soenke; Peldschus, Kersten; Rodt, Thomas; Kral, Florian; Matthies, Herbert; Kikinis, Ron; Becker, Hartmut: Registration and Fusion of CT and MRI of the Temporal Bone. In: *Journal of Computer Assisted Tomography* 29 (2005), Nr. 3, S. 305–310

[Bux02] Buxton, Richard: *Introduction to Functional Magnetic Resonance Imaging: Principles and Techniques*. The Edinburgh Building, Cambridge CB2 2RU, UK: Cambridge University Press, 2002

[CBOFO09] Connor, Steve; Bell, D.J.; O'Gorman, Ruth; Fitzgerald-O'Connor, Alec: CT and MR Imaging Cochlear Distance Measurements May Predict Cochlear Implant Length Required for a 360° Insertion. In: *American Journal of Neuroradiology* 30 (2009), Nr. 7, S. 1425–1430

[CKS97] Caselles, Vicent; Kimmel, Ron; Sapiro, Guillermo: Geodesic Active Contours. In: *Int.J of Comp Vis* 22 (1997), Nr. 1, S. 61–79

[CTCG95] Cootes, Tim; Taylor, Christopher; Cooper, David; Graham, Jim: Active shape models - their training and application. In: *Computer Vision and Image Understanding* 61 (1995), S. 38–59

[CV01] Chan, Tony; Vese, Luminita: Active contours without edges. In: *IEEE Transactions on Image Processing* 10 (2001), Nr. 2, S. 266–277

[DAD10] Dalhoum, Abdel Latif A.; Al-Dhamari, Ibraheem: FMRI Brain Data Classification Using Cellular Automata. In: *Proceedings of the 10th WSEAS International Conference on Applied Informatics and Communications, and 3rd WSEAS International Conference on Biomedical Electronics and Biomedical Informatics*. Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS), 2010, S. 348–352

[DCRB19] Diez, David; Cetinkaya-Rundel, Mine; Barr, Christopher: *OpenIntro Statistics*. 4^{th} edition. Electronic copy, 2019 `https://leanpub.com/openintro-statistics`

[DH16] Dees, Guido; Hoof, Robert van Marcand S. Marcand Stokroos: A Proposed Method for Accurate 3D Analysis of Cochlear Implant Migration Using Fusion of Cone Beam CT. In: *Frontiers in surgery* 3 (2016), Jan, S. 2–2

[Dic45] Dice, Lee: Measures of the Amount of Ecologic Association Between Species. In: *Ecology* 26 (1945), Nr. 3, S. 297–302

[EJD+06] Escudé, Bernard; James, Chris; Deguine, Olivier; Cochard, Nadine; Eter, Elias; Fraysse, Bernard: The size of the cochlea and predictions of insertion depth angles for cochlear implant electrodes. In: *Audiol Neurootol* 1 (2006), S. 27–33

[ELRA11] Erixon, Elsa; Liu, Wei; Rask-Andersen, Helge: Anatomic Studies of the Human Cochlea: Implications for Cochlear Implantation. In: *Cochlear Implants Int.* 1 (2011), S. 8–13

[ETC98] Edwards, G.J.; Taylor, Christopher; Cootes, Tim: Interpreting face images using active appearance models. In: *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition* (1998), S. 300–305

[Fel14] Felix, Hausdorff: *Grundzuege der Mengenlehre*. Leipzig, Veit. Reprinted by Chelsea Publishing Company in 1949, 1914

[FH00] Farin, Gerald; Hansford, Dianne: *The Essentials of CAGD*. 1st. USA: A. K. Peters, Ltd., 2000

[FZA+19] Forouzandeh, Farzad; Zhu, Xiaoxia; Alfadhel, Ahmed; Ding, Bo; Walton, Joseph P.; Cormier, Denis; Frisina, Robert D.; Borkholder, David A.: A nanoliter resolution implantable micropump for murine inner ear drug delivery. In: *Journal of Controlled Release* 298 (2019), S. 27 – 37. – ISSN 0168–3659

[Gal54] Galambos, Robert: Physiological Acoustics. Ernest Glen Wever and Merle Lawrence. Princeton Univ. Press, Princeton, N.J., 1954. In: *Science* 119 (1954), Nr. 3095, S. 560–560. – ISSN 0036–8075

[Gos04] Goshtasby, Ardeshir: *Image Registration: Principles, Tools and Methods*. Springer-Verlag, London, 2004

[GRB+17] Gerber, Nicolas; Reyes, Mauricio; Barazzetti, Livia; Kjer, Hans; Vera, Sergio; Stauber, Martin; Mistrik, Pavel; Ceresa, Mario; Mangado, Nerea; Wimmer, Wilhelm; Stark, Thomas; Paulsen, Rasmus; Weber, Stefan; Caversaccio, Marco; Ballester, Miguel: A multiscale imaging and modelling dataset of the human inner ear. In: *Scientific Data* 4 (2017), Nr. 170132

[GV13] Golub, Gene; Van Loan, Charles: *Matrix Computations*. 4*th* edition. Baltimore, Maryland 21218-4363, USA: The Johns Hopkins University Press, 2013

[GW06] Gonzalez, Rafael; Woods, Richard: *Digital Image Processing*. 3. Prentice-Hall, Inc., 2006

[GZR+16] Ghani, Muhammad; Zhou, Zhongxing; Ren, Liqiang; Li, Yuhua; Zheng, Bin; Yang, Kai; Liu, Hong: Investigation of spatial resolution characteristics of an in vivo micro computed tomography system. In: *Nuclear instruments & methods in physics research. Section A, Accelerators, spectrometers, detectors and associated equipment* 807 (2016), Jan, S. 129–136

[Har38] Hardy, Mary: The length of the organ of Corti in man. In: *American Journal of Anatomy, Wiley-Blackwell* 62 (1938), Nr. 2, S. 291–311

[HH01] Hajnal, Joseph; Hill, Derek: *Medical Image Registration*. CRC press, 2001

[HVW+15] Homann, Georg; Vieth, Volker; Weiss, Daniel; Nikolaou, Konstantin; Heindel, Walter; Notohamiprodjo, Mike; Böckenfeld, Yvonne: Semi-Quantitative vs. Volumetric Determination of Endolymphatic Space in Menière's Disease Using Endolymphatic Hydrops 3T-HR-MRI after Intravenous Gadolinium Injection. In: *PLoS ONE* 10 (2015), Nr. 3, S. 1–12

[HW79] Hartigan, John; Wong, Anthony: Algorithm AS 136: A K-Means Clustering Algorithm. In: *Journal of the Royal Statistical Society, Series C* 28 (1979), Nr. 1, S. 100–108

[IELA18] Iyaniwura, John; Elfarnawany, Mai; Ladak, Hanif; Agrawal, Sumit: An automated A-value measurement tool for accurate cochlear duct length estimation. In: *Journal of Otolaryngology, Head and Neck Surgery* 47 (2018), Jan, Nr. 5, S. 1–8

[IOI86] Igarashi, Makoto; Ohashi, Kazuhiro; Ishii, Masanori: Morphometric comparison of endolymphatic and perilymphatic spaces in human temporal bones. In: *Acta Otolaryngol* 101 (1986), Nr. 3, S. 161–164

[JKB+17] Jagt, Annerie der; Kalkman, Randy; Briaire, Jeroen; Verbist, Berit; Frijns, Johan: Variations in cochlear duct shape revealed on clinical CT images with an automatic tracing method. In: *Scientific Reports 17566* 7 (2017), Nr. 1, S. 1–9

[KEZ+17] Koch, Robert; Elfarnawany, Mai; Zhu, Ning; Ladak, Hanif; Agrawal, Sumit: Evaluation of Cochlear Duct Length Computations Using Synchrotron Radiation Phase-Contrast Imaging. In: *Otology and Neurotology* 38 (2017), Nr. 6, S. 92–99

[KFV+16] Kjer, Hans; Fagertun, Jens; Vera, Sergio; Gil, Debora; Ballester, Miguel; Paulsen, Rasmus: Free-form image registration of human cochlear CT data using skeleton similarity as anatomical prior. In: *Pattern Recognition Letters* 76 (2016), S. 76–82

[KLEA17] Koch, Robert; Ladak, Hanif; Elfarnawany, Mai; Agrawal, Sumit: Measuring Cochlear Duct Length - a historical analysis of methods and results. In: *Journal of otolaryngology, head and neck surgery* 46 (2017), Mar, Nr. 1, S. 19–19

[KPSV09] Klein, Stefan; Pluim, Josien; Staring, Marius; Viergever, Max: Adaptive stochastic gradient descent optimisation for image registration. In: *International Journal of Computer Vision* 81 (2009), Nr. 3, S. 227–239

[KPV14] Kikinis, Ron; Pieper, Steve; Vosburgh, Kirby: 3D Slicer: a platform for subject-specific image analysis, visualization, and clinical support. In: *Intraoperative Imaging Image-Guided Therapy, Ferenc A. Jolesz* 3 (2014), Nr. 19, S. 277–289

[KSAP11] Klein, Stefan; Staring, Marius; Andersson, Patrik; Pluim, Josien: Preconditioned Stochastic Gradient Descent Optimisation for Monomodal Image Registration. In: Fichtinger, Gabor (Hrsg.); Martel, Anne (Hrsg.); Peters, Terry (Hrsg.): *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2011*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, S. 549–556

[KSC96] Kawano, Atsushi; Seldon, Henry; Clark, Graeme: Computer-aided three-dimensional reconstruction in human cochlear maps: measurement of the lengths of organ of Corti, outer wall, inner wall, and Rosenthal's canal. In: *Annals of Otology, Rhinology and Laryngology* 105 (1996), Nr. 9, S. 701–709

[KSM+10] Klein; Staring; Murphy; Viergever; Pluim: elastix: a toolbox for intensity-based medical image registration. In: *IEEE Trans Med Imaging. 2010* 29 (2010), Nr. 1, S. 196–205

[KSP07] Klein, Stefan; Staring, Marius; ; Pluim, Josien: Evaluation of Optimization Methods for Nonrigid Medical Image Registration Using Mutual Information and B-Splines. In: *IEEE Transactions on Image Processing* 16 (2007), Nr. 12, S. 2879–2890

[KSW+98] Ketten, Darlene; Skinner, Martha; Wang, Gongbo; Vnnier, Michael; Gates, George; Neely, John: In vivo measures of cochlear length and insertion depth of nucleus cochlear implant electrode arrays. In: *Ann Otol Rhinol Laryngol Suppl.* 175 (1998), S. 1–16

[KVF+15] Kjer, Hans; Vera, Sergio; Fagertun, Jens; Gil, Debora; González Ballester; Miguel Angel; Paulsen, Rasmus: Image Registration of Cochlear µCT Data Using Heat Distribution Similarity. In: Paulsen, Rasmus (Hrsg.); Kim, Pedersen (Hrsg.): *Image Analysis*. United States: Springer Science+Business Media, 2015, S. 234–245

[KWT98] Kass, Michael; Witkin, Andrew; Terzopoulos, Demetri: Snakes: Active contour models. In: *International Journal of Computer Vision* 1 (1998), S. 321–331

[LGF00] Leventon, Michael; Grimson, Eric; Faugeras, Olivier: Statistical shape influence in geodesic active contours. In: *Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No.PR00662)* Bd. 1, 2000, S. 316–323

[Mag66] Maggio, Elio: *The humoral system of the labyrinth.* Almqvist & Wiksell, 1966

[MBLS01] Malik, Jitendra; Belongie, Serge; Leung, Thomas; Shi, Jianbo: Contour and Texture Analysis for Image Segmentation. In: *International Journal of Computer Vision* 43 (2001), Nr. 1, S. 7–27

[MCB20] Moudgalya, Sanketh; Cahill, Nathan; Borkholder, David: Deep Volumetric Segmentation of Murine Cochlear Compartments from Micro-Computed Tomography Images. In: *2020 42nd Annual International Conference of the IEEE Engineering in Medicine Biology Society (EMBC)*, 2020, S. 1970–1975

[MHV+01] Mattes, David; Haynor, David; Vesselle, Hubert; Lewellyn, Thomas; Eubank, William: Non-rigid multimodality image registration. In: *Medical Imaging 2001: Image Processing, SPIE Publications* (2001), S. 1609–1620

[MJ16] Mistrak, Pavel; Jolly, Claude: Optimal electrode length to match patient specific cochlear anatomy. In: *European Annals of Otorhinolaryngology, Head and Neck Diseases* 133 (2016), S. 68–71

[MS89] Mumford, David; Shah, Jayant: Optimal approximations by piecewise smooth functions and associated variational problems. In: *Communications on Pure and Applied Mathematics* 42 (1989), Nr. 5, S. 577–685

[NLMD11] Noble, Jack; Labadie, Robert; Majdani, Omid; Dawant, Benoit: Automatic Segmentation of Intracochlear Anatomy in Conventional CT. In: *IEEE Transaction on Biomedical Engineering* 58 (2011), Nr. 9, S. 2625–2632

[NW06] Nocedal, Jorge; Wright, Stephen: *Numerical Optimization*. 2^{nd} edition. New York, NY, USA: Springer, 2006

[OS88] Osher, Stanley; Sethian, James: Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. In: *Journal of Computational Physics* 79 (1988), S. 12–49

[Par62] Parzen, Emanuel: On Estimation of a Probability Density Function and Mode. In: *The Annals of Mathematical Statistics* 33 (1962), Nr. 3, S. 1065–1076

[PB07] Preim, Bernhard; Bartz, Dirk; Bettany-Saltikov, Josette (Hrsg.): *Visualization in Medicine. Theory, Algorithms, and Applications*. 30 Corporate Drive, Suite 400, Burlington, MA 01803, USA: Morgan Kaufmann,an imprint of Elsevier, 2007

[PC04] Plakhov, Alexander; Cruz, João: A Stochastic Approximation Algorithm with Step-Size Adaptation. In: *Journal of Mathematical Sciences* 120 (2004), Mar, Nr. 1, S. 964–973

[PFS87] Pollak, Anita; Felix, Heidi; Schrott, Annelise: Methodological Aspects of Quantitative Study of Spiral Ganglion Cells. In: *Acta Otolaryngol* 436 (1987), S. 37–42

[Phi94] Phillips, Dwayne: *Image Processing in C: Analyzing and Enhancing Digital Images*. USA: R D Publications, Inc., 1994

[PKV+16] Pujadas, Esmeralda; Kjer, Hans; Vera, Sergio; Ceresa, Mario; Ballester, Miguel: Cochlea segmentation using iterated random walks with shape prior. In: Styner, Martin (Hrsg.); Angelini, Elsa (Hrsg.); International Society for Optics and Photonics (Veranst.): *Medical Imaging 2016: Image Processing* Bd. 9784 International Society for Optics and Photonics, SPIE, 2016, S. 778–786

[PZK+11] Poznyakovskiy, Anton; Zahnert, Thomas; Kalaidzidis, Yannis; Lazurashvili, Nikoloz; Schmidt, Rolf; Hardtke, Hans-Jürgen; Fischer, Björn; Yarina, Yury: A segmentation method to

obtain a complete geometry model of the hearing organ. In: *Hearing Research* 282 (2011), Nr. 1, S. 25–34

[QLS19] Qiao, Yuchuan; ; Lelieveldt, Boudewijn; Staring, Marius: An Efficient Preconditioner for Stochastic Gradient Descent Optimization of Image Registration. In: *IEEE Transactions on Medical Imaging* 38 (2019), Nr. 10, S. 2314–2325

[QSLS15] Qiao, Yuchuan; Sun, Zhuo; Lelieveldt, Boudewijn; Staring, Marius: A Stochastic Quasi-Newton Method for Non-Rigid Image Registration. In: Navab, Nassir (Hrsg.); Hornegger, Joachim (Hrsg.); Wells, William (Hrsg.); Frangi, Alejandro (Hrsg.): *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Cham: Springer International Publishing, 2015, S. 297–304

[QvLS16] Qiao, Yuchuan; van Lew, Baldur; Lelieveldt, Boudewijn; Staring, Marius: Fast Automatic Step Size Estimation for Gradient Descent Optimization of Image Registration. In: *IEEE Transactions on Medical Imaging* 35 (2016), Feb, Nr. 2, S. 391–403

[RALE+12] Rask-Andersen, Helge; Liu, Wei; Erixon, Elsa; Kinnefors, Anders; Pfaller, Kristian; Schrott-Fischer, Annelies; Glueckert, Rudolf: Human cochlea: anatomical characteristics and their relevance for cochlear implantation. In: *Anat Rec (Hoboken)* 295 (2012), Nr. 11, S. 1791–1811

[RBM+05] Rohlfing, Torsten; Brandt, Robert; Menzel, Randolf; Russakoff, Daniel; Maurer, Calvin: Quo Vadis, Atlas-Based Segmentation? In: *Handbook of Biomedical Image Analysis: Volume III: Registration Models* 3 (2005), S. 435–486

[RCH+17] Rivas, Alejandro; Cakir, Ahmet; Hunter, Jacob; Labadie, Robert; Zuniga, Geraldine; Wanna, George; Dawant, Benoit; Noble, Jack: Automatic Cochlear Duct Length Estimation for Selection of Cochlear Implant Electrode Arrays. In: *Otol Neurotol.* 38 (2017), Nr. 3, S. 339–346

[Ret84] Retzius, Gustaf: *Das Gehoerorgan der Wirbelthiere, morphologisch histologische Studien. Das Gehoerorgan der Reptilien, der Voegel und der Saeugethiere.* Stockholm: Gedruckt in der Centraldruckerei: In Commission bei Samson und Wallin, 1884

[RFB15] Ronneberger, Olaf; Fischer, Philipp; Brox, Thomas: U-Net: Convolutional Networks for Biomedical Image Segmentation. In: Navab, Nassir (Hrsg.); Hornegger, Joachim (Hrsg.); Wells, William (Hrsg.); Frangi, Alejandro (Hrsg.): *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Cham: Springer International Publishing, 2015, S. 234–241

[RM51] Robbins, Herbert; Monro, Sutton: A Stochastic Approximation Method. In: *The Annals of Mathematical Statistics,* 22 (1951), Nr. 3, S. 400–407

[RML+14] Reda, Fitsum; McRackan, Theodore; Labadie, Robert; Dawant, Benoit; Noble, Jack: Automatic segmentation of intra-cochlear anatomy in post-implantation CT of unilateral cochlear implant recipients. In: *Med Image Anal* 18 (2014), Nr. 3, S. 605–615

[RNLD12] Reda, Fitsum; Noble, Jack; Labadie, Robert; Dawant, Benoît: Fully Automatic Surface-Based Pre- to Intra-operative CT Registration for Cochlear Implant. In: Dawant, Benoît (Hrsg.); Christensen, Gary (Hrsg.); Fitzpatrick, Michael (Hrsg.); Rueckert, Daniel (Hrsg.): *Biomedical Image Registration*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, S. 89–98

[Run01] Runge, Carl: Über empirische Funktionen und die Interpolation zwischen äquidistanten Ordinaten. In: *Zeitschrift für Mathematik und Physik* 46 (1901), Nr. 1, S. 224–243

[Set99] Sethian, James: *Level Set Methods and Fast Marching Methods Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge Monograph on Applied and Computational Mathematics, Cambridge University Press, 1999

[SFS06] Scarfe, William; Farman, Allan; Sukovic, Predag: Clinical Applications of Cone-Beam Computed Tomography in Dental Practic. In: *Journal Canadian Dental Association (JCDA)* 72 (2006), Nr. 1, S. 75–80

[Sha14] Shaw, Chris: *Cone Beam Computed Tomography*. CRC Press, 2014

[SLY+09] Sobrinho, Fernando; Lazarini, Paulo; Yoo, Hea; Junior, Luiz; Meira, Altino: A method for measuring the length of the cochlea through magnetic resonance imaging. In: *Rev. Bras. Otorrinolaringol* 75 (2009), Nr. 2, S. 261–267

[Sny05] Snyman, Jan: *Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms*. Springer Publishing, 2005

[SRK+18] Sekuboyin, Anjany; Rempfler, Markus; Kukačka, Jan; Tetteh, Giles; Valentinitsch, Alexander; Kirschke, Jan; Menze, Bjoern: Btrfly Net: Vertebrae Labelling with Energy-Based Adversarial Learning of Local Spine Prior. In: *Medical Image Computing and Computer Assisted Intervention (MICCAI 2018)*. Cham: Springer International Publishing, 2018, S. 649–657

[SS73] Schoenberg, Isaac; Sharma, Ana: Cardinal interpolation and spline fucntions V. The B-splines for cardinal Hermite interpolation. In: *Linear Algebra and its Applications* 7 (1973), Nr. 1, S. 1–42

[SSBL07] Stakhovskaya, Olga; Sridhar, Divya; Bonham, Ben; Leake, Patricia: Frequency Map for the Human Cochlear Spiral Ganglion: Implications for Cochlear Implants. In: *J Assoc Res Otolaryngol.* 8 (2007), Nr. 2, S. 220–233

[SST91] Sato, Hiroaki; Sando, Isamu; Takahashi, Haruo: Sexual dimorphism and development of the human cochlea. Computer 3-D measurement. In: *Acta Otolaryngol.* 111 (1991), Nr. 6, S. 1037–1040

[Ste57] Steinhaus, Hugo: Sur la division des corps matériels en parties. In: *Bull. Acad. Polon. Sci.* 4 (1957), Nr. 12, S. 801–804

[Sue09] Suetens, Paul: *Fundamentals of Medical Imaging*. 2. Cambridge University Press, 2009

[SW17] Saputroa, Dewi; Widyaningsih, Purnami: Limited memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method for the parameter estimation on geographically weighted ordinal logistic regression model (GWOLR). In: *AIP Conference Proceedings* 1868 (2017), Nr. 1, S. 0400091–0400099

[Sø48] Sørensen, Thorvald: A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons. In: *Kongelige Danske Videnskabernes Selskab* 4 (1948), Nr. 1, S. 1–34

[TS89] Takagi, Akira; Sando, Isamu: Computer-aided three-dimensional reconstruction: a method of measuring temporal bone structures

including the length of the cochlea. In: *nnals of Otology, Rhinology and Laryngology* 7 (1989), Nr. 1, S. 515–522

[TU00] Thevenaz, Philippe; Unser, Michael: Optimization of mutual information for multiresolution image registration. In: *IEEE Transactions on Image Processing* 9 (2000), Nr. 12, S. 2083–2099

[UAE91] Unser, Michael; Aldroubi, Akram; Eden, Murray: Fast B-spline transforms for continuous image representation and interpolation. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13 (1991), Nr. 3, S. 277–285

[Uns99] Unser, Michael: Splines: a perfect fit for signal and image processing. In: *IEEE Signal Processing Magazine* 16 (1999), Nr. 6, S. 22–38

[UVJ87] Ulehlova, Libuse; Voldrich, Lubos; Janisch, Rudolf: Correlative study of sensory cell density and cochlear length in humans. In: *Hearing Research* 28 (1987), Nr. 2, S. 149–151

[VK05] Vezhnevets, Vladimir; Konushin, Vadim: GrowCut: Interactive Multi-Label N-D Image Segmentation By Cellular Automata. In: *Proc. Graphicon.* (2005), S. 150–156

[VV14] Vaid, Sanjay; Vaid, Neelam: Imaging for cochlear implantation: Structuring a clinically relevant report. In: *Clinical Radiology* 69 (2014), Nr. 7, S. 307–322

[VW97] Viola, Paul; Wells, William: Alignment by Maximization of Mutual Information. In: *International Journal of Computer Vision* 24 (1997), Nr. 2, S. 137–154

[Web88] Webb, Steve: *The Physics of Medical Imaging.* Boca Raton, Fla: CRC Press, 1988

[WGW+17] Weber, Stefan; Gavaghan, Kate; Wimmer, Wilhelm; Williamson, Tom; Gerber, Nicolas; Anso, Juan; Bell, Brett; Feldmann, Arne; Rathgeb, Christoph; Matulic, Marco; Stebinger, Manuel; Schneider, Daniel; Mantokoudis, Georgios; Scheidegger, Olivier; Wagner, Franca; Kompis, Martin; Caversaccio, Marco: Instrument flight to the inner ear. In: *Science Robotics* 2 (2017), Nr. 4, S. 1–30

[WLLM14] Würfel, Waldemar; Lanfermann, Heinrich; Lenarz, Thomas; Majdani, Omid: Cochlear length determination using Cone Beam Computed Tomography in a clinical setting. In: *Hearing Research* 316 (2014), S. 65–72

[WPRC+04] Watrin-Pinzano, A; Ruaud, J.-P; Cheli, Y; Gonord, P; Grossin, L; Gillet, P; Blum, A; Payan, E; Olivier, P; Guillot, G; Netter, P; Loeuille, D: T2 mapping: an efficient MR quantitative technique to evaluate spontaneous cartilage repair in rat patella. In: *Osteoarthritis and Cartilage* 12 (2004), Nr. 3, S. 191–200

[XP97] Xu, Chenyang; Prince, Jerry L.: Gradient Vector Flow: A New External Force for Snakes. In: *IEEE proc. Conf. on Comp. Vis. Patt. Recog CVPR* (1997), S. 66–71

[Yoo12] Yoo, Terry: *Insight into Images: Principles and Practice for Segmentation, Registration, and Image Analysis*. AK Peters Ltd, 2012

[ZLY96] Zhu, Song; Lee, T. S.; Yuille, Alan: Region competition: unifying snakes, region growing, energy/Bayes/MDL for multi-band image segmentation. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18 (1996), Nr. 9, S. 884–900

Internet-Resources

[1] Large Scale Vertebrae Segmentation Challenge 2019. https://verse2019.grand-challenge.org, last visit on 3/12/2020.

[2] OpenStax: Anatomy and OpenStax College physiology. https://openstax.org/books/anatomy-and-physiology/pages/14-1-sensory-perception, last visit on 1/12/2020.

[3] Calculus by Gilbert Strange. https://ocw.mit.edu/ans7870/resources/Strang/Edited/Calculus/Calculus.pdf, last visit on 3/12/2020.

[4] CBCT. https://commons.wikimedia.org/wiki/File:NAOMI-CT_CBCT_scanner.jpg, last visit on 1/12/2020.

[5] Slicer Cervical Spine code. https://github.com/MedicalImageAnalysisTutorials/SlicerCervicalSpine, last visit on 3/12/2020.

[6] Slicer Cochlea code. https://github.com/MedicalImageAnalysis Tutorials/SlicerCochlea, last visit on 3/12/2020.

[7] Snake contour Example. https://commons.wikimedia.org/wiki/File:Snake-contour-example.jpg, last visit on 3/12/2020.

[8] CT. https://en.wikipedia.org/wiki/File:UPMCEast_CTscan.jpg, last visit on 1/12/2020.

[9] Bezier curves by Jon Shiash. www.youtube.com/watch?v= 2HvH9cmHbG4, last visit on 3/12/2020.

[10] Youtube: How do X-Rays Work? www.youtube.com/watch?v=LtX3rYJASik, last visit on 1/12/2020.

[11] How does an MRI machine work? www.youtube.com/watch?v=nFkBhUYynUw, last visit on 1/12/2020.

[12] elastix. https://elastix.lumc.nl, last visit on 1/12/2020.

[13] ITK Software Guide. https://itk.org/ITKSoftwareGuide/html/Book1/ITKSoftwareGuide-Book1ch4.html, last visit on 1/12/2020.

[14] Cochlea implant. https://en.wikipedia.org/wiki/Cochlearimplant/media/File:Blausen 02, last visit on 1/12/2020.

[15] Cochlea implant types. www.cochlear.com, last visit on 3/12/2020.

[16] Cochlear implants comparisons. www.cochlear.com, last visit on 3/12/2020.

[17] Insight Toolkit (ITK). www.itk.org, last visit on 1/12/2020.

[18] Level-Set. https://en.wikipedia.org/wiki/Level-set_method, last visit on 3/12/2020.

[19] Matlab. www.mathworks.com/products/matlab, last visit on 1/12/2020.

[20] Interpolation methods. https://upload.wikimedia.org/wikipedia/ commons/9/90/Comparison_of_1D_and_2D_interpolation.svg, last visit on 1/12/2020.

[21] Medical Imaging Interaction Toolkit (MITK). www.mitk.org, last visit on 1/12/2020.

[22] MRI. https://commons.wikimedia.org/wiki/File:Petmr.jpg, last visit on 1/12/2020.

[23] A Primer on Bézier Curves. https://pomax.github.io/bezierinfo, last visit on 3/12/2020.

[24] openGL. www.opengl.org, last visit on 3/12/2020.

[25] 3D Slicer. www.slicer.org, last visit on 1/12/2020.

[26] ITK Snap. www.itksnap.org, last visit on 1/12/2020.

[27] Coordinate systems. www.slicer.org/wiki/Coordinate_systems, last visit on 1/12/2020.

[28] Slicer Cochlea video. www.youtube.com/watch?v=A_mTcT3eT_c&t=7s, last visit on 3/12/2020.

[29] Slicer Crvical Spine video. www.youtube.com/watch?v=GX8FQMkhRuU&t= 51s, last visit on 3/12/2020.

[30] Visualisation Toolkit (VTK). www.vtk.org, last visit on 1/12/2020.

[31] A Wolfram Web Resource. Weisstein, Eric W. B-Spline. From MathWorld. https://mathworld.wolfram.com/B-Spline.html, last visit on 3/12/2020.

[32] X-ray. https://en.wikipedia.org/wiki/X-ray, last visit on 1/12/2020.