UNIVERSITY OF VAASA

FACULTY OF TECHNOLOGY

COMPUTER SCIENCE

Suvi Karhu

EVALUATION OF THE RESULTS OF ORTHODONTIC TREATMENT BY NON-RIGID IMAGE REGISTRATION AND DEFORMATION-BASED MORPHOMETRY

Master's thesis for the degree of Master of Science in Technology submitted for inspection, Vaasa, 10.1 2014.

Supervisor	Prof. Jouni Lampinen
Instructors	Prof. Jarmo Alander, D.Sc. Vladimir Bochko

FOREWORD

I would like to thank Professor Jarmo Alander who suggested me this research topic and gave me the possibility to work in HammasSkanneri research project. I thank him also for sharing his valuable comments and advice about the thesis, and for providing references to many scientific articles about image registration.

I thank researcher, D.Sc. Vladimir Bochko for all important advice and the MATLAB code which he gave me. I also thank my supervisor, Prof. Jouni Lampinen, for guidance in the thesis writing process. I thank B.Sc. Jaakko Yli-Luukko for helping me with the imaging setup and D.Sc. Petri Välisuo for sharing his knowledge about 3D imaging. I also express my gratitude towards Acting Chief Dental Officer of the City of Vaasa, Lic. Katri Palo for answering my questions about dentistry, as well as orthodontist, D.Sc. Mathias Grön for borrowing his orthodontics books.

I also want to thank my family and friends for their support during this project. Finally, I deeply thank God who gives His strength and grace for every day.

In Isokyrö, 20.11.2013

Suvi Karhu

TABLE OF CONTENTS **page**

SYMBOLS AND ABBREVIATIONS

Symbols

β^3	A cubic B-spline
f	focal length

Mathematical notations

$\sum_{n \in \mathrm{Z}}^{N}$	sum over a range from n to N, where n belongs to the set of integers
$x \in S$	Variable x belongs to the set S
Z	the set of integers

Abbreviations

CT	Computerized Tomography, a medical imaging method
CMVS	Clustering Views for MultiView Stereo, an algorithm for assisting MVS algorithms
FMI	Focused mutual information, a modification of MI
GNU	GNU's Not Unix, an operating system
DSSIM	Structural Dissimilarity, a similarity metric based on SSIM
DLL	Dynamic Link Library, a library file used by a program
FFD	Free-Form Deformation, a type of image deformation
FPGA	Field Programmable Gate Array, a logic device
GUI	Graphical User Interface, a type of user interface
ISO	International Organization for Standardization
ITK	Insight ToolKit, an image registration and segmentation toolkit
MI	Mutual Information, a similarity metric
MRI	Magnetic Resonance Imaging, a medical imaging method
PMVS	Patch-Based MultiView Stereo, a type of MVS algorithm
SIP	a tool for connecting C/C++ programs or libraries with Python
SfM	Structure-from-Motion, a 3D construction approach
MVS	MultiView Stereo, an algorithm for constructing 3D

PIL	Python Imaging Library, a Python library for image processing
PPT	Python Photogrammetry Toolbox, a photogrammetry software
PyQt4	an interface for using Qt with Python
Qt	a cross-platform application framework for application software and GUIs
RGB	Red-Green-Blue, a color model
RMI	Regional Mutual Information, a modification of MI
ROI	Region-of-Interest, a specific area in an image
SIFT	Scale Invariant Feature Transform, feature extraction method
VTK	Visualization ToolKit, an image visualization toolkit
VV	an image visualization tool
VXL	Vision-*Something*-Library, a C/C++ library for computer vision tasks
.DICOM	a 3D raster image file format

VAASAN YLIOPISTO
Teknillinen tiedekunta
Tekijä: Suvi Karhu
Diplomityön nimi: Hampaiden oikomishoidon tulosten arviointi elastisella kuvien kohdistuksella ja muodonmuutokseen perustuvalla morfometrialla
Valvojan nimi: Prof. Jouni Lampinen
Ohjaajan nimi: Prof. Jarmo Alander, D.Sc. Vladimir Bochko
Tutkinto: Diplomi-insinööri
Koulutusohjelma: Tietotekniikan koulutusohjelma
Suunta: Ohjelmistotekniikka
Opintojen aloitusvuosi: 2007
Diplomityön valmistumisvuosi: 2014 **Sivumäärä:** 73

TIIVISTELMÄ

Tämän tutkimuksen tavoitteena oli selvittää, voidaanko hampaiden oikomishoidon tuloksia arvioida käyttämällä hammaskipsimalleista otettujen valokuvien elastista kohdistusta, sekä kehittää ohjelma, jonka avulla valokuvien arviointiprosessi voidaan automatisoida. Arviointia haluttiin kokeiltavan myös kipsimalleista tehdyillä kolmiulotteisilla malleilla. Tämä tutkimus rajattiin koskemaan vain hammaskaaren sisällä ilmeneviä purentavirheitä, eikä kaarten välisiä suhteita otettu huomioon. Tämä tutkielma tehtiin osana Vaasan yliopiston Sähkö- ja energiatekniikan yksikön HammasSkanneri-tutkimusprojektia, jonka tavoite on automatisoida hammaskipsimallien digitointi ja arkistointi.

Tässä tutkimuksessa käytettiin kaksiulotteisia valokuvia kipsimalleista, jotka oli otettu oikomishoidetuista potilaista ennen ja jälkeen hoidon. Elastinen kuvien kohdistus suoritettiin käyttämällä Fiji-ohjelmiston rekisteröintityökalua. Kohdistuksen tarkkuutta arvioitiin mittaamalla manuaalisesti asetettujen maamerkkien etäisyyksiä, sekä vertaamalla kohdistettujen kuvien ja alkuperäisten kohdekuvien viiva- ja kulmaparametrien arvoja. Hampaiden siirtymiä approksimoitiin käyttämällä muodonmuutokseen perustuvaa morfometriaa.

Kuvien kohdistuksen tarkkuus on kohtuullisissa virherajoissa, jos kuva otetaan suoraan kipsimallin yläpuolelta, ja kohdistus suoritetaan käyttämällä apuna ihmisen syöttämiä maamerkkejä. Muutosten arviointi osoitti, että hampaiden liikkeitä voidaan mitata karkeasti käytämällä muodonmuutokseen perustuvaa morfometriaa, joka perustuu Jacobian-estimaatteja muistuttaviin muutosestimaatteihin. Tarkkuuden ja muutosten arviointiin kehitettiin työkaluja, jotka osittain automatisoivat arviointien suorittamisen. Kipsimallien kolmiulotteinen kuvantaminen epäonnistui, minkä vuoksi kolmiulotteisen arviointijärjestelmän kehittäminen jätettiin jatkotutkimusten kohteeksi.

AVAINSANAT: Elastinen kuvien kohdistus, hampaiden oikomishoito, kipsimalli, morfometria

UNIVERSITY OF VAASA
Faculty of Technology
Author: Suvi Karhu
Topic of the Thesis: Evaluation of the results of orthodontic treatment by non-rigid image registration and deformation-based morphometry
Supervisor: Prof. Jouni Lampinen
Instructor: Prof. Jarmo Alander, D.Sc. Vladimir Bochko
Degree: Master of Science in Technology
Degree Programme: Degree Programme in Information Technology
Major of Subject: Software Engineering
Year of Entering the University: 2007
Year of Completing the Thesis: 2014 **Pages:** 73

ABSTRACT

The goal of this research was to find out, whether the non-rigid registration of dental casts can be used in the evaluation of orthodontic treatment and to develop a program, which would at least partially automatize the evaluation process of images. The aim was also to experiment the evaluation of three-dimensional models of the casts. This research was delimited to cover only the evaluation of malocclusions within the dental arch. The relationships between the dental arches were not considered. This thesis was done in the University of Vaasa at the Department of Electrical Engineering and Energy Technology as a part of the HammasSkanneri research project, whose aim is to automatize the digitization and archiving of dental casts.

This research used two-dimensional images of dental casts which were taken of orthodontically treated patients before and after orthodontic treatment. Non-rigid registration was performed by using a registration tool of Fiji software. The evaluation of the accuracy of the registration was performed by measuring distances between manually inserted landmarks, and by comparing the linear and angular parameters of the registered images and the original target images. The displacements of the teeth were approximated with the help of deformation-based morphometry.

The accuracy of registration is within reasonable error limits, if the image is taken straight from above of the cast and the registration is performed with the help of landmarks inserted by a human. Estimation of the changes showed that the movement of teeth can be coarsely measured by using deformation-based morphometry based on change estimates that resemble the Jacobian estimates. A set of programs, which partially automatize the evaluation of the accuracy and the changes, were developed. Three-dimensional imaging of the casts was unsuccessful, and thus the development of 3D evaluation system was left as a future research topic.

KEYWORDS: non-rigid image registration, orthodontic treatment, dental cast, deformation-based morphometry

1. INTRODUCTION

Problems in occlusion are very common. Hardly anyone has ideal occlusion, but often malocclusions are not severe enough to require treatment. Malocclusions which cause eating or speaking difficulties, mouth breathing, pain or aesthetical problems, require orthodontic treatment. Orthodontic treatment, a process of aligning and straightening teeth, is a part of standard dentistry in developed countries. It is often performed by using dental braces with metal wires, which force teeth to correct positions by applying pressure to them. Many other orthodontic treatment methods exist, too, and the most severe malocclusions are also treated surgically. (Proffit, Fields, Ackerman, Sinclair, Thomas, Tulloch 2004: 4-7).

Orthodontists need exact information of patient's dentition to plan orthodontic treatment and evaluate its results. This information is acquired with the help of dental casts. Orthodontists use these models to inspect how the jaws relate to each other and how the teeth are arranged inside the jaws.

Dental casts are stored several years in dental offices. They need annual maintenance to check if any damages have appeared in the model (Sinthanayothin, Phichitchaiphan, and Bholsithi 2010). Because storing the models is laborious and requires a lot of space, digital models of the casts have become an interesting alternative to physical models. In this thesis two-dimensional digital images and three-dimensional models of dental casts are used to take measurements which help orthodontists in evaluation of the results of the treatment.

1.1. Literature review

In this thesis, non-rigid image registration is used for evaluating the results of orthodontic treatment. Results of the registration are used for deformation-based morphometry to evaluate the changes between the images before and after treatment. Evaluation of changes is done with the help of ratio-of-areas estimates which are quite

similar to Jacobian determinants. Evaluation is implemented for 2D images and experimented with 3D models.

The methods and techniques used in this thesis have been employed in many other studies. Image registration has been used by many researchers in medical imaging problems. It is used e.g. to register images obtained from different sensors or from different viewpoints or at different times. Particlarly, non-rigid image registration is used to register images of soft moving organs, like heart. Registration of human teeth has been used e.g. in post-mortem identification of individuals (Nassar, Origala, Adjeroh & Ammar (2006) *et al.* 2006; Santamaría, Cordón, & Damas 2011a).

Jacobian determinants have showed to be the most meaningful morphological measure of brain tissue growth (Chung, Worsley, Paus, Cherif, Collins, Giedd, Rapoport, Evans 2001). Riddle, Li, Fitzpatrick, DonLevy, Dawant & Price (2004) used color-coded Jacobian values of 3D images for estimating changes in both artificial images and real-structure MRI images. They found color-coded Jacobian values to be convenient method for identifying changes between images.

Digital models of dental casts have been used by Horton, Miller, Gaillard and Larson (2010) to measure Bolton Index, which is an index used by orthodontists to plan orthodontic treatment. Sinthanayothin, Phichitchaiphan, Wongwaen, Bholsithi (2010) developed a system for archiving, communication and analyzing of 3D dental cast models. The analyzing module, AnaDent3D Viewer, could be used for analyzing overjet and overbite, occlusion contact and Bolton Index. Sinthanayothin, Phichitchaiphan, and Bholsithi (2010) developed an online dental database for post-mortem dental identification system. Yamamoto, Hayashi, Nishikawa, Nakamura and Mikami (1991) developed a system for measuring three-dimensional tooth movement during orthodontic treatment. They used stereophotogrammetry and triangulation for image aqcuisition and rigid-body registration of dental cast profiles for measurements. Also holography has been used in evaluation of orthodontic treatment. Positional changes of teeth can be detected as discrepancies between the cast and the hologram image.

1.2. Scope of this work

This thesis was done in the University of Vaasa at the Department of Electrical Engineering and Energy Technology as a part of the HammasSkanneri research project, the aim of which is to automatize the digitization and archiving of dental casts. The goal of this thesis was to find out, whether the non-rigid registration of dental casts can be used in the evaluation of orthodontic treatment and to design a program, which would at least partially automatize the evaluation process of 2D images and be easy to use. The evaluation was also wanted to be experimented with 3D models of the casts.

The research questions were

1. How accurately non-rigid registration can register 2D images of tooth cast models?
2. How can tooth displacements be evaluated using the results of non-rigid registration of 2D images?
3. How the evaluation could be extended to cover also 3D images?

This research was limited to cover only evaluation of malocclusions within one dental arch. Relationships between the dental arches were not considered. The research approach was qualitative: the phenomenons happening during the orthodontic treatment were tried to be understood, and image analysis techniques were tested on a small set of samples.

2. ORTHODONTIC TREATMENT

Teeth do not always grow correctly. Malocclusion – a condition of misaligned tooth or teeth – is actually very common. The solution to malocclusions is orthodontic treatment. Orthodontic treatment is the process of aligning and straightening teeth. This specialty of dentistry started developing in the late 1800s by the work of Edward Angle, who dedicated his life for developing orthodontic practices. Nowadays orthodontic treatment is a part of standard dentistry in developed countries.

This chapter describes the basics of orthodontic treatment. First the anatomy of teeth is presented and then malocclusions and their treatment are explained.

2.1. Anatomy of teeth

Teeth are divided into two arches: lower jaw (mandible) and upper jaw (maxilla). There are four types of teeth: incisors, canines, premolars and molars. Incisors, or “front teeth”, are flat-shaped and have a sharp, horizontal edge for cutting food. Canines are strong corner teeth with one cusp (a point-shaped biting surface). Canines are meant for tearing food. Premolars have two or three cusps, and their job is chewing. Molars, also meant for chewing, are similar to premolars but have four or five cusps. Molars are the largest teeth in the back of the mouth. (Fig. 1) (Simplyteeth 2012.)

In primary dentition there are two incisors, one canine and two molars in both sides of each jaw. In permanent dentition there are also two premolars and possibly a third molar (wisdom tooth). Teeth are numbered so that the central incisors have number 1, lateral incisors number 2, canines number 3 and so on (Fig. 1).

Terminology of anatomical locations and directions is used in dental texts. In the context of mouth, anterior means something that is in the front of the mouth, while posterior means the back of the mouth. Almost in similar meanings are used terms mesial and distal. Mesial means something that is nearer the middle and front of the

dental arch, and distal is the opposite. Labial means towards lip, lingual towards tongue, buccal towards cheek and palatal towards the palate. (Fig. 1)

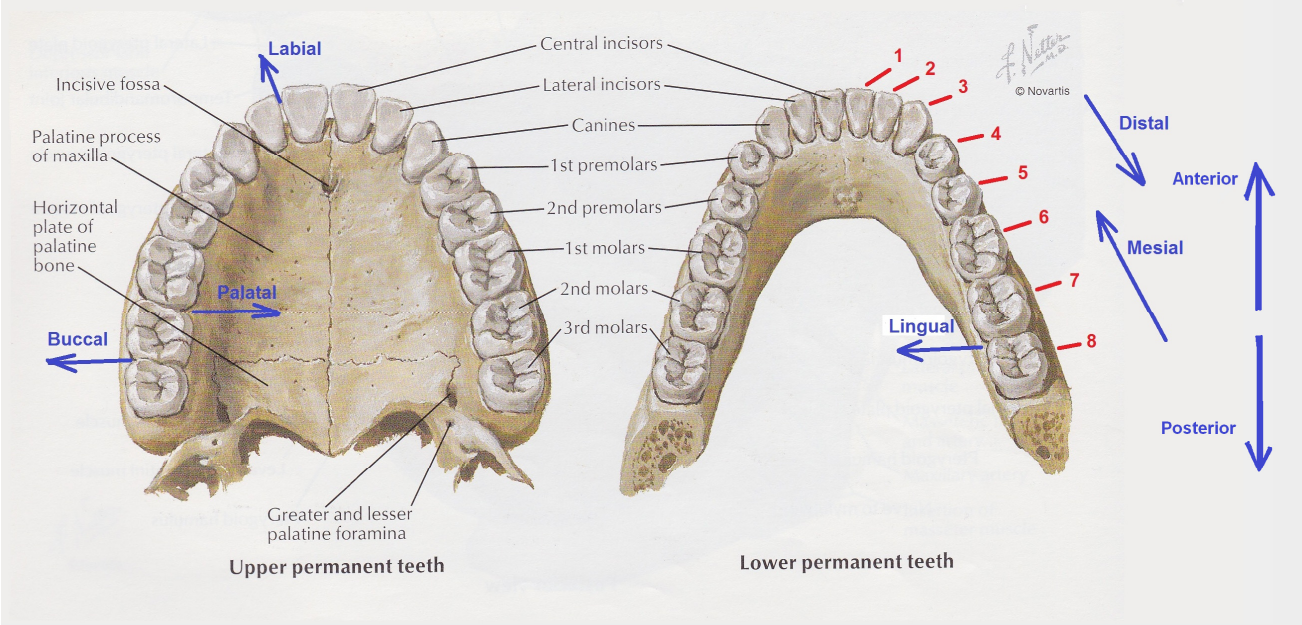

Figure 1. Parts of upper and lower arch along with terminology of anatomical locations and numbers of teeth (modified from Netter 1989: 50).

2.2. Development of dentition and occlusion

Development of primary, or deciduous, teeth starts in utero with hard tissue formation. Crowns of teeth start mineralizing also before birth, and have completed approximately by the age of 12 months. Primary teeth start erupting at the age of 9 months, and have become fully erupted usually a bit after 2 years. Central incisors erupt first, and the next ones are lateral incisors, first molars, canines and second molars, in this order. Formation of tooth root continues after eruption and is completed around the age of 3 years. (Haavikko 1985: 46–52).

The resorption (destruction) of roots of primary teeth starts from central incisors at the age of 4-5 years. When the primary teeth exfoliate one by one, permanent teeth may erupt. The order of the eruption of permanent teeth may vary, but the most frequent

order is presented in Figure 2, which shows that for most of the teeth in lower jaw erupt a bit earlier than their antagonists in the upper jaw. Deviation from the normal order of eruption may affect the occlusion: for example eruption of the second molars before the second premolar or canine may cause crowding. (Haavikko 1985: 46–52).

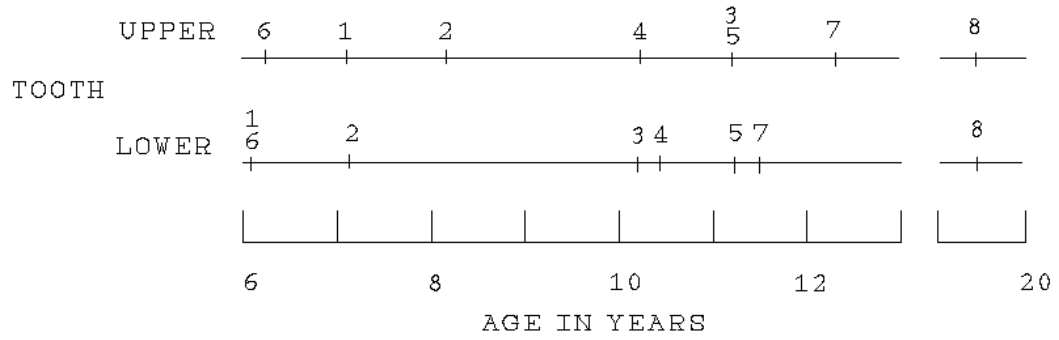

Figure 2. The most frequent order of eruption of permanent teeth (Haavikko 1985: 51). Teeth are numbered so that the central incisors have number 1, lateral incisors number 2, canines number 3 and so on.

The occlusion is mainly established in childhood, but continues to change to some extent throughout life. (Rönning 1985: 63). Development of occlusion is a combination of genetic and enviromental factors.

2.3. Malocclusions

Malocclusion is a condition of misaligned tooth/teeth or incorrect relation between the upper and lower dental arch. Malocclusions are common, but often not serious enough to require treatment. Severe malocclusions require orthodontic or surgical treatment.

Several things may cause malocclusions. Some childhood habits, like nail biting, pencil biting, finger sucking and prolonged use of pacifier or a baby bottle often cause malocclusions. Sometimes malocclusion is linked to other disorders of the patient. For example diseases that cause weak bite force, mouth breathing, abnormal posture and swallowing disorders, may lead to malocclusion.

The most common classification of malocclusions is the Angle's classification. However, this classification is limited to antero-posterior relationships of the arches, and thus a more comprehensive classification, presented by Lundström (1985) is used in this thesis. This classification divides the malocclusions to malpositions. which are misalignments within the dental arches, and malrelations, which are improper relationships between the dental arches.

2.3.1. Malpositions

Malpositions include single-tooth misalignments and space problems within dental arch. They belong to Angle's Class I, if not combined with malrelations.

Single-tooth misalignments can be displacements, inclinations or rotations (Fig. 3). They may appear as ectopic eruption (abnormal path of eruption), impaction (non-eruption), supraocclusion (over-eruption), infraocclusion (undereruption). Also single-tooth crossbites and scissor bites, which are incorrect relations between a lower and an upper tooth, may develop. (Lundström 1985: 92).

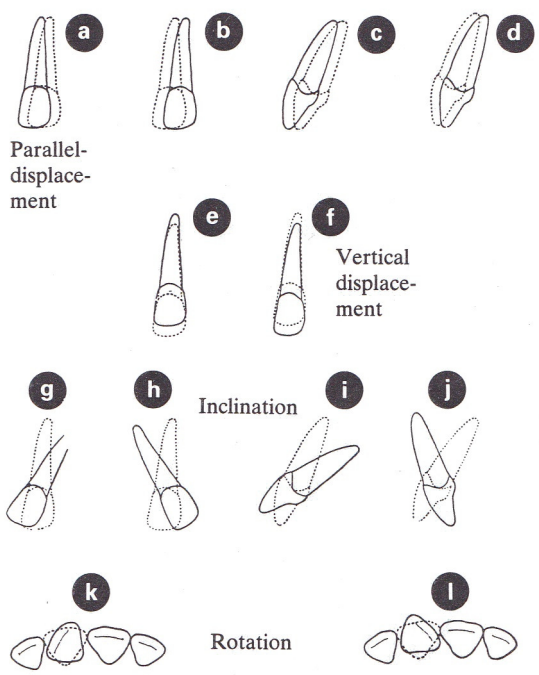

Figure 3. Single-tooth displacements. (Lundström 1985: 91).

Space discrepancies can be divided to crowding and spacing. In crowding teeth overlap because of too little space in the dental arch. This may be caused by abnormally small arch, big teeth or too many teeth. The opposite phenomenon is spacing, where there are gaps between the teeth. Space discrepancies can be evaluated with the help of Bolton Index. Bolton Index is the sum of lower jaw mesial-distal tooth widths divided by the sum of their upper-arch antagonists widths. The ratio tells the orthodontist if modifications in tooth size need to be done to achieve good treatment results. (Horton, Miller, Gaillard & Larson 2010).

2.3.2. Malrelations

The two dental arches may be incorrectly related to each other in three planes: sagittal, transversal and vertical.

Most malrelations occur in the sagittal relationships of the jaws. In distal occlusion (Fig. 4) the lower arch is posterior in its relation to the upper arch. Distal occlusion is usually combined with overjet, which is horizontal protrusion of upper central incisors past the lower ones (Fig. 5). Big overjet is usually caused by proclination (labial inclination) of upper central incisors, whereas patients with smaller, but still remarkable overjet, usually have retroclined (palatally inclined) upper incisors. Distal occlusion corresponds to Angle's Class II malocclusion, and is much more common than its opposite, mesial occlusion. (Lundström 1985: 101-106). Mesial occlusion is shown in Figure 6.

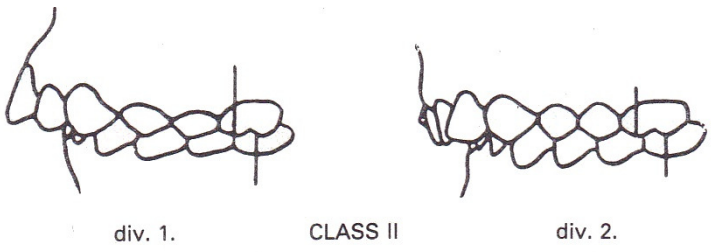

Figure 4. Distal occlusion (Angle Class II malocclusion), divided into two subclasses on the ground of inclination of the incisors: In the subclass 1 the incisors are proclined, causing a big overjet. In subclass 2 the incisors are retroclined. (Lundström 1985: 90).

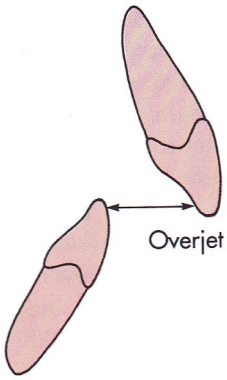

Figure 5. Overjet is defined as horizontal ovarlapping of upper central incisors. (Proffit *et al*. 2004: 8).

In mesial occlusion the lower arch is anterior in its relation to the upper arch (Fig. 6). Mesial occlusion is often combined with proclination of upper central incisors. Mesial occlusion corresponds to Angle's Class III malocclusion (Lundström 1985: 101-106).

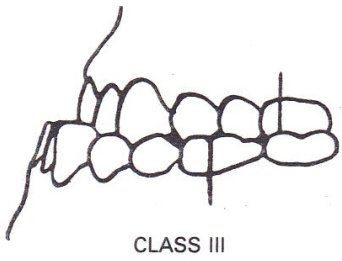

Figure 6. Mesial occlusion (Angle Class III malocclusion). Lower arch is anterior to the upper arch. (Lundström 1985: 90).

Deviations in transversal plane include cross bites and scissor bites. In these malocclusions upper and lower teeth do not meet correctly because others are too much buccal (near cheek) or lingual (near the tongue) or palatal (near the palate).

Vertical malocclusions include deep overbites and open bites. Overbite of 3-5 mm is normal, but deep overbite happens when upper central incisors go too far down past the lower ones. Open bite is the opposite of deep overbite. Overbite and open bite are illustrated in Figure 7.

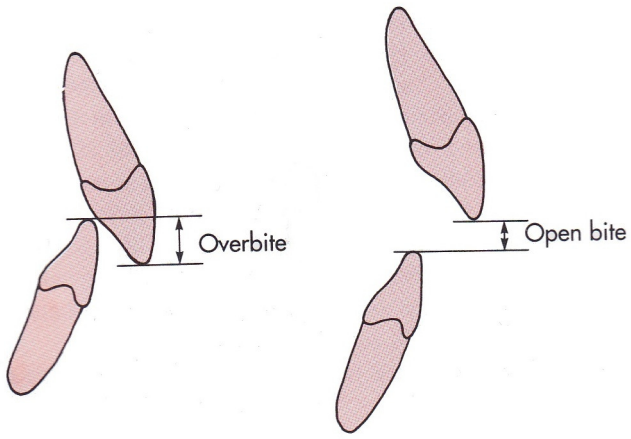

Figure 7. Overbite and open bite are measured as vertical distance between upper and lower incisors. (Proffit *et al.* 2004: 8).

2.4. Orthodontic treatment

Orthodontic treatment is based on applying forces to teeth. These forces make the teeth to move to desired locations.

2.4.1. Biological basis of orthodontic treatment

Teeth are connected to tooth sockets by periodontal ligament fibers. When a tooth is moved during the orthodontic treatment process, some areas of the socket exhibit tension and some compression (Fig. 6). At the tensioned zone the osteoblasts start forming bone and at the compressed zone osteoclasts start removing bone. These processes stabilize the tooth to its new position. (Avery 1992: 148-149).

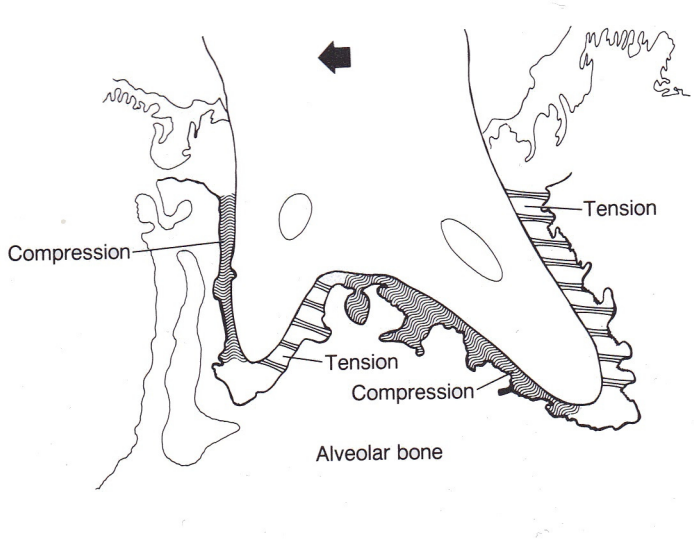

Figure 8. Movement of a tooth. (Avery 1992: 148).

Teeth also have natural movements. Particularly, a phenomenon called mesial drift means teeth's tendency of moving from back of the arches towards the front and midline of the mouth. Avery (1992: 146) assume that all the teeth of the mouth drift mesially, from 0.05 to 0.7 mm per year. Proffit *et al.* (2004: 220-221) state that mesial drift occurs mainly in second molars.

2.4.2. Dental casts

A patient to be orthodontically treated is first examined by visually inspecting the deformities of patient's face and mouth. After this the orthodontist needs a dental cast to plan the treatment.

Dental casts are made by casting plaster to a dental impression. Dental impression is taken by putting a dental impression tray into the mouth of the patient. The tray is filled with some viscous liquid material, usually sodium alginate. When the patient bites the material, an imprint of his/her dentition forms to the material. A dental technician then casts the positive mould by casting plaster to the dental impression.

2.4.3. Orthodontic appliances

Orthodontic treatment is often performed with dental braces which consist of metal wires or springs with attachments. These devices are familiar to almost everyone, but also many other orthodontic appliances exist. Single teeth can be moved by using springs. Width of the dental arch may be expanded with palatal expanders. Certain types of malocclusions are treated with functional appliances that use the natural forces of muscle activity, growth and tooth eruption to guide teeth and jaws to correct positions. When the desired occlusion has been achieved, patients usually have to wear retainers, which maintain the occlusion. In some cases retainers need to be used throughout the rest of the life.

2.4.4. Evaluation of the results of orthodontic treatment

Evaluation of the results of orthodontic treatment can be done by visual inspection and sometimes by taking measurements from dental casts with a ruler and/or a calliper. Also X-ray images may be used.

Evaluation of orthodontic treatment is sometimes done by inserting metal implants into jaws. Changes in the locations of jaws and teeth with respect to these landmarks can then be observed from X-ray images. Earlier metal implants were routinely inserted to jaws before orthodontic treatment, but later it has also been noticed that certain parts of the oral cavity stay stable in their locations and can be used as "natural landmarks" (Rönning 1985:80). For example the palate doesn't significantly deform during the orthodontic process and thus it can be used as a reference. (Yamamoto *et al.* 1991). Other such areas are the anterior surface of zygomatic process from 10 years onwards, tooth buds until their root formation starts, inner cortical structure of the symphysis, and mandibular canal. (Rönning 1985:80). However "neither implants nor 'natural landmarks' make a complete separation of growth changes from those produced by orthodontic treatment possible". (Rönning 1985:80).

3. NON-RIGID IMAGE REGISTRATION

Image registration is a process of matching corresponding points between images which have been taken for example from different viewpoints, at different times or with different sensors. Registration problems can be classified into rigid and non-rigid. In rigid-body registration the object has not deformed between the images, and the transformation is *affine*, which means that parallel lines remain parallel (Fig. 9). In non-rigid body registration the object has deformed between the images (Fig. 10). This sort of registration - also called elastic registration - is known to be a complex and slow process. It is needed especially in medicine to register images of soft moving organs, like heart, or images of different modalities like magnetic resonance imaging (MRI) and computed tomography (CT) images. (Crum, Hartkens & Hill 2004).

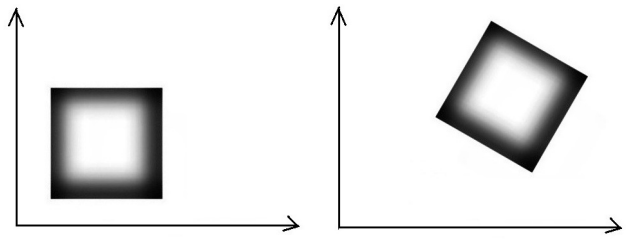

Figure 9. An image pair where a translation and rotation has occurred between the images. Finding the corresponding points between the images imposes a rigid registration problem, which is rather easy and fast to solve.

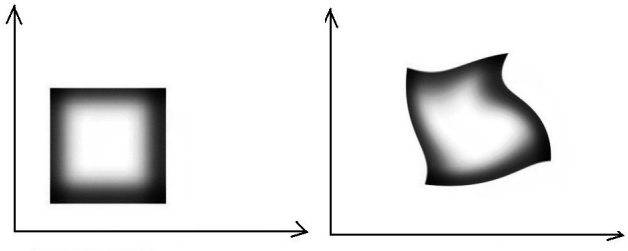

Figure 10. An image pair where elastic transformation has occurred between the images. Finding the corresponding points imposes a non-rigid (elastic) registration problem, which is much more complex and slow to solve than rigid registration problem.

Registration algorithms need two input images: a source image and a target image. The source image is then registered to the coordinate system of the target image. Searching the best transform is an optimization problem, which consists of four components: image representation, transformation, similarity metric, and optimizer. These components are described more detailed in the following sections.

3.1. Image representation

Registration algorithms take two images as their inputs: source image and target image. The source image often has to be evaluated at non-integer positions, *i.e.* at subpixel accuracy. The values between two integer positions are evaluated by using some interpolation or approximation function. One possible solution is to use approximation based on cubic B-splines. B-splines are flexible curves consisting of segments of polynomials, which are called blending functions or basis functions. The shape of a B-spline can be controlled with the help of control points. B-splines are defined as

$$P(u) = \sum_{k=0}^{n} p(k) B_{k,d}(u), \qquad u_{\min} \le u \le u_{\max} \qquad 2 \le d \le n+1 \tag{1}$$

where $p(k)$ is a set of $n+1$ control points and $B_{k,d}(u)$ are the blending functions. The blending functions are defined by Cox-deBoor recursion formulas (de Boor 1978: 89)

$$B_{k,1}(u) = \begin{cases} 1, & \text{if } u_k \le u < u_{k+1} \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

$$B_{k,d} = \frac{u - u_k}{u_{k+d-1} - u_k} B_{k,d-1}(u) + \frac{u_{k+d} - u}{u_{k+d} - u_{k+1}} B_{k+1,d-1}(u) \tag{3}$$

A B-spline with $n+1$ control points has $n+1$ blending functions. Each blending function is defined over d subintervals of the total range of u. The polynomial curve is of degree $d-1$. For cubic B-splines, $d = 4$, and thus the curve has degree 3. (Hearn & Baker 1997: 335.)

Images can be represented with B-splines in the following way:

$$I_i(x,y) = \sum_{k,l \in Z^2} c_{k,l} \beta^3\left(\frac{x}{h} - k\right) \beta^3\left(\frac{y}{h} - l\right), \quad (4)$$

where β^3 is a cubic B-spline, $c_{k,l}$ are the B-spline coefficients (control points) and h is a parameter that controls the level of the detail of the representation. This representation provides a good trade-off between accuracy and speed. (Arganda-Carreras, Sorzano, Thévenaz, Muñoz-Barrutia, Kybic, Marabini, Carazo & Ortiz-de-Solorzano 2010.)

Registration algorithm may use either raw image data or features extracted from images. Feature-based representations use lines, points, edges and/or corners to represent the most relevant features of the image. A common feature descriptor is Scale Invariant Feature Transform (SIFT) descriptor. The SIFT descriptor is built using SIFT algorithm, which creates unique and highly descriptive features from an image (May, Turner & Morris 2011). These features are "invariant to rotation and robust to changes in scale, illumination, noise and small changes in viewpoint" (May *et al.* 2011).

3.2. Transformation

Several transformation models can be used in an elastic registration algorithm. To obtain an initial, coarse solution, a rigid transformation - consisting of translation and rotation - can be used. To accomplish a more accurate alignment, a similarity transform, which allows translation, rotation and uniform scaling, and affine transform, which allows translation, rotation, uniform or nonuniform scaling, mirror and shear, may be useful.

All of the transformations described above are represented with matrices that contain the coefficients of the transformation. These kind of parametric transformations are suitable for capturing the overall motion of an object, but they are not capable of describing local deformations (Rueckert, Sonoda, Hayes, Hill, Leach & Hawkes 1999). Local deformations can be modelled with free-form deformation (FFD), the idea of

which is to deform an object by manipulating an underlying mesh of control points (Rueckert *et al.* 1999). Free-form deformation is often based on cubic B-splines. B-splines are computationally light, differentiable and allow close control of the level of the detail of the transformation. (Arganda-Carreras *et al.* 2010). Transformation can be represented as a linear combination of cubic B-splines:

$$T(x,y) = \sum_{k,l \in Z^2} (c^{ij}_{1,k,l}, c^{ij}_{2,k,l}) \beta^3 \left(\frac{x}{s_x} - k \right) \beta^3 \left(\frac{y}{s_y} - l \right), \quad (5)$$

where s_x and s_y are the sampling steps that control the level of the detail of the deformation field (Arganda-Carreras *et al.* 2010).

3.3. Similarity metric

Commonly used similarity measures are *i.a.* mutual information, cross-correlation and mean square error. Similarity measure can also consist of several terms, like consistency term and similarity term as done by Arganda-Carreras *et al.* (2006), or intensity difference and intensity gradient as in optical flow algorithm.

Mutual information (MI) works well with high-resolution images but becomes statistically inconsistent when applied to small, low-resolution images that have no clear structure (Andronache *et al.* 2007). Andronache *et al.* have proposed a method that avoids problems of MI by using stopping criterion for subdivision and applies cross-correlation instead of MI for small patches. Fookes and Maeder (2003) have combined MI with viscous-fluid algorithm which recovers local misregistrations. Russakoff, Tomasi, Rohlfing and Maurer (2004) have developed regional mutual information (RMI) algorithm which improves performance of MI by taking into account not only the relationships between individual pixels but also the neighbourhood of the pixels.

Another similarity measure, Structural Dissimilarity (DSSIM), introduced by Loza, Mihaylova, Canagarajah and Bull (2006) has in recent years become accepted among

similarity metrics. DSSIM is based on Structural Similarity (SSIM) index, introduced by Wang, Sheikh, Bovik, and Simoncelli (2004). DSSIM compares the structural and spatial characteristics of two images, and is designed to correspond to human visual system. DSSIM is claimed to be robust to contrast and illumination changes and has proved to perform well in many cases. Also normalized histogram, histogram intersection or Earth mover's distance can be used as similarity measure.

3.4. Search strategy

A common way to solve the non-rigid body registration problem is to use multiresolution pyramid. This method utilizes coarse-to-fine strategy. The registration is first performed in the coarsest level and then those results are used at the next (lower) level, which has higher resolution. At the lowest level of the pyramid, the full resolution of the original image is used. Sometimes registration strategies are changed between the levels. For example it may be useful to use translation transform at the coarsest level and change to affine transform at finer levels. (Insight Software Consortium 2003.)

Multiresolution pyramid approach requires the image to be subsampled, *i.e*. decimated. Badshah, O'Leary, Harker and Sallinger (2011) have emphasized the importance of the decimation method. They presented an algorithm that uses Savitzky-Golay smoothing (Savitzky & Golay 1964) for decimating, and modified normalized phase correlation to perform local registrations. Their algorithm reduced aliasing and Gibbs error (Gibbs 1898), which, according to the authors, are the reason for problems in many registration algorithms.

Some algorithms divide the images into subimages, patches, which are registered individually and then combined to produce the final result. A combination of subdivision method and multiresolution pyramid can be called hierarchical subdivision. This method has been used *i.a.* by Andronache, Siebenthal, Székely and Cattin (2007) and by Badshah *et al.* (2011).

Usually registration is performed unidirectionally, *i.e.* the source image *S* is registered to the target image *T*. Another approach is to register images bidirectionally to achieve consistent registration. In bidirectional registration both direct and inverse registrations are performed. Arganda-Carreras *et al.* (2006) have developed a consistent elastic registration algorithm, which uses consistency term as a part of the similarity measure.

Because non-rigid registration problem is a complex problem and its search space is huge, heuristic methods are used to search the solution. Evolutionary methods, which formally utilize the concepts of evolution theory, have been employed by Santamaria, Cordon and Damas (2011a, 2011b), Okaha and Saitoh (2011), and Battezzato, Gastaldi and Pastorelli (2011). Other population-based methods, like particle-swarm optimization have been employed by Zhao, Zeng, Lei and Ma (2012).

3.5. Implementation

Many software packages have been developed for image registration purposes. Insight Toolkit (ITK) is a C++ image processing framework that contains many algorithms suitable for non-rigid registration. ITK can be used with both 2D and 3D images. However, ITK is quite a complex framework and requires time to install and get familiar with. Simpler interface to ITK is implemented in SimpleITK layer, which can be used with Java or Python. Even easier way is to use Elastix toolbox, which is based on ITK and designed particularly for non-rigid (elastic) registration. Elastix is able to register both 2D and 3D images and provides a command-line user interface.

ITK concentrates mostly on intensity-based registration, which means that the algorithms work with raw pixel data. Sometimes it is better to use features, which contain only the most relevant information of the image. A C/C++ library called RGRL is targeted for this purpose. RGRL is a part of Vision-*Something*-Library (VXL), and compiles with ITK so that components of both RGRL and ITK can be used in a particular program.

Image registration algorithms may also be implemented with commercial scientific computing software Matlab or similar free software GNU Octave. These tools have image registration functions in their image processing packages, and also many user implementations can be found from the web. Also a Java-based free software called ImageJ is can be used in medical imaging. Fiji, a distribution of ImageJ, is especially targeted for life sciences image processing. Fiji is easy to install and use, and comes with lots of plugins useful in image registration and segmentation. Particularly, a plugin called *bUnwarpJ,* developed by Arganda-Carreras *et al.* (2006) performs consistent and bidirectional registration by using B-splines. However, this tool is able to register only 2D images. In addition to these tools, also other image registration tools, like Syntegra from Philips Medical Systems, are commercially available.

Executing registration procedures is computationally intensive. Large clusters of processors, multi-threaded solutions and Field Programmable Gate Array (FPGA) -based implementations have been used to reduce execution time from hours to minutes and seconds (Dandekar 2007; Cong, Huang & Zou 2011; Buder 2012). Badshah *et al.* (2011) have developed a Matlab program that requires only 1.3 seconds to register images of size 800 x 500 pixels.

3.6. Algorithm evaluation

The accuracy of a registration algorithm needs to be evaluated somehow. Often researchers do this by applying a known deformation to an image and compare the results to the correct solution, or by placing landmarks to source and target images and determining how well the registration matches those points. One way to check the correctness of the registration is to use consistency measure, which determines if the registration from source to target and target to source produce the same alignment. Sometimes only visual inspection can be done to validate the results of the registration. (Crum et al. 2004). Computing the difference between deformed image and target image may also help, and calculating the Jacobian values of the deformed image may provide useful information of the correctness of the registration. (Klein & Staring 2012.)

3.7. Estimation of changes

Non-rigid registration is used in deformation-based morphometry and deformation-based volumetry, which deal with estimating differences between images. These estimation techniques use deformation grids that can be made by creating a regular grid and applying the deformation obtained from the registration to the grid. For example Jacobian determinants can be used for the estimation of differences from those grids. Jacobian matrix **J** is a matrix of partial derivatives, and the Jacobian determinant is the determinant det(**J**) of that matrix:

$$\det(\mathbf{J}) = |\mathbf{J}| = \begin{vmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \end{vmatrix} = \frac{\partial x}{\partial u}\frac{\partial y}{\partial v} - \frac{\partial x}{\partial v}\frac{\partial y}{\partial u} \tag{5}$$

Geometrically the Jacobian determinant tells the scaling factor between the area of an undeformed rectangle and a parallelogram which approximates the deformed rectangle (Andrilli & Hecker 2010; Knisley & Shirley 2001) (Fig. 11):

$$\det(\mathbf{J}) = \frac{D_2}{D_1} = \frac{\left\|\vec{w}_1 \times \vec{w}_2\right\|}{\Delta u \Delta v} = \frac{\left(\frac{\partial x}{\partial u}\frac{\partial y}{\partial v} - \frac{\partial x}{\partial v}\frac{\partial y}{\partial u}\right)\Delta u \Delta v}{\Delta u \Delta v} = \frac{\partial x}{\partial u}\frac{\partial y}{\partial v} - \frac{\partial x}{\partial v}\frac{\partial y}{\partial u} \tag{6}$$

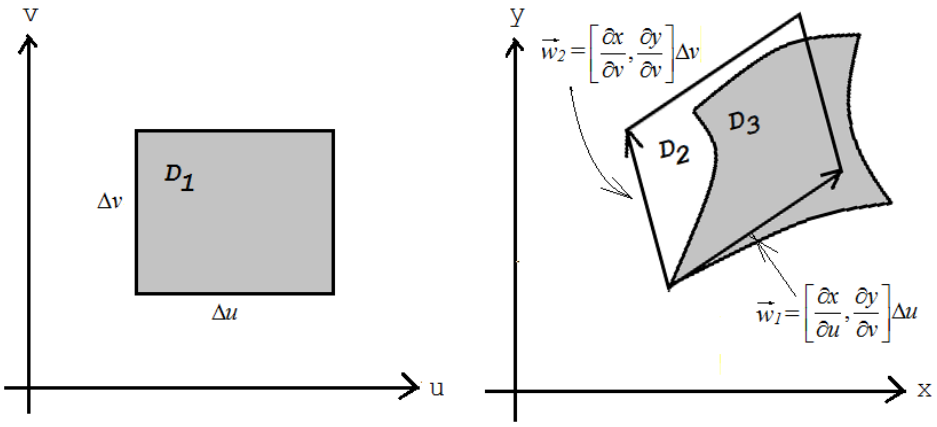

Figure 11. Jacobian determinant in 2D (reproduced from Andrilli & Hecker 2010; Knisley & Shirley 2001).

In this research, however, the actual area of the deformed rectangle was used instead of the area of the approximating parallelogram (Fig. 11). Thus the ratio of the areas was calculated as $R = \frac{D_3}{D_1}$. The percentual change, *C,* was then calculated with the help of *R*: $C = (R-1)\cdot 100\%$. In this thesis, this value *C* will be called a ratio-of-areas estimate, a change estimate, or simply percentual change. When emphasizing its similarity with Jacobian, it can also be called "change estimate, which resembles Jacobian estimate".

3.8. Applications

Non-rigid body registration is widely used in medicine. Often it is used to register images of different modalities, *e.g.* MRI (magnetic resonance imaging) and CT (computed tomography) images. Image registration can be used for example in radiotherapy planning (Loi *et al.* 2008), imaging-guided interventions (Battezzato *et al.* 2011), monitoring swallowing (Aung, Goulermas, Hamdy & Power 2010), detecting breast cancer (Rueckert *et al.* 1999), cardiology, and brain imaging.

Non-rigid body registration has also been applied in dental sciences. Nassar *et al.* (2006) have developed an algorithm to register radiography images of teeth. They used edge detection, multiresolution pyramid and genetic algorithm with Hausdorff distance similarity measure. The algorithm was designed for post-mortem identification of individuals. Also Santamaría *et al.* (2011a) have applied image registration to human teeth in forensics. Teeth were rotated on a turntable and scanned with 3D scanner at every 60 degrees. They used evolutionary algorithms to register images. Bro-Nielsen, Kramkov and Kreiborg (1997) have used bone growth model to register images of mandibles. The model simulates the actual physical process of bone growth.

In addition to medical applications, non-rigid registration is also needed in remote sensing, artificial vision, computer-aided design, surveillance, strain measurements and in many other applications. Strain measurements by non-rigid registration have been studied *i.a.* in the University of Vaasa by Koljonen (2010).

4. IMAGING

Acquisition of images is an important step in solving a computer vision problem. Especially appropriate illumination of the object is crucial. When the illumination is even, algorithms can perform more robustly and effectively. If several images of the same object are taken, it is important that the illumination is similar between the images, because many computer vision algorithms perform worse if the intensity level between the images varies. Thus all the images should be taken under the same conditions. It is also important that the object is not over- or underexposed, and that the image has sufficient depth of field. These properties can be controlled by the camera's aperture size and exposure time.

Three-dimensional imaging of the dental casts is even more challenging than 2D imaging. In this chapter, two methods, namely imaging based on structured light and stereophotogrammetric imaging are considered.

4.1. Devices

The images were taken with a digital system camera, Nikon D200. Images were processed with Lenovo ThinkPad T400 and E520 computers with Fiji software. In 3D imaging a turntable Thorlabs NR360S/M was used.

4.2. 2D imaging

In 2D imaging of the dental casts, the most important thing to consider is to avoid overexposing the cast. The exposure of an image is determined by three parameters: the aperture of the lens, exposure time and ISO Speed. The exposure time, or shutter speed, is the time the camera's shutter is open. The aperture size is the diameter of the hole through which the light can come to the image sensor. Aperture size is often specified with F-number. F-number is the ratio of camera's lens's focal length f to the aperture

diameter. Thus the aperture size decreases as the F-number increases. F-number also affects the resulting photo's depth of field, *i.e.* the range of distance over which the objects appear to be sharp. A high F-number (small aperture) results in a large depth of field, whereas a low F-number (big aperture) results in a narrow depth of field. The third parameter, ISO-Speed, controls the sensitivity of the camera's sensor to a given amount of light. High ISO-Speed values tend to cause noise, and thus usually the lowest possible ISO-Speed is selected. (Cambridge in Colour 2013).

In this study, two imaging sessions were performed. In the first session, the exposure time was set to 1/5 seconds and the F-number was set to f/5.6. In the second session, exposure time was set to 1/6 seconds and the F-number was set to f/36. ISO-Speed was ISO-100 in both sessions.

In the first session, images were taken from oblique angle above the cast. However, this frontal view did not seem to be useful for determining the movements of the teeth, because the front of the mouth was nearer the camera than the back of the mouth (Fig. 12). Thus in the second session the images were taken from occlusal view, *i.e.* straight above the cast (Fig. 13).

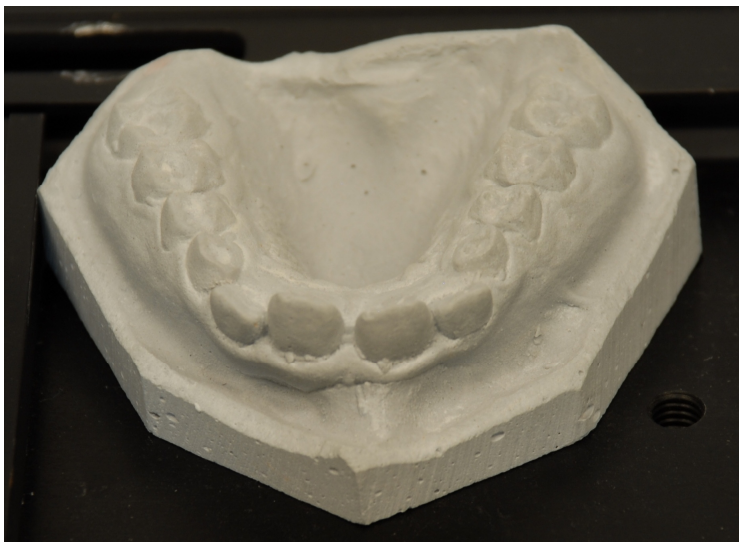

Figure 12. Upper jaw before treatment from frontal view and oblique angle.

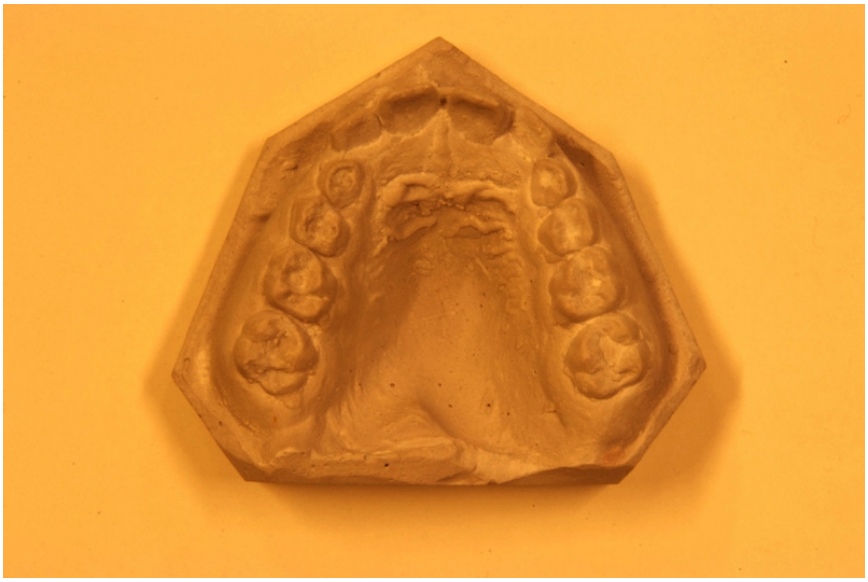

Figure 13. Upper jaw before treatment from occlusal view.

4.3. Preprocessing

Preprocessing of images was an important step in this study because good lightning equipment was not available. By preprocessing images it is possible to make important details more visible and to correct the overall quality of the image.

Some of the most popular image preprocessing techniques are contrast enhancing and edge detection. However, in this case those methods did not produce desired results. Better results were achieved with the help of local contrast enhancing algorithm. This tool performs contrast enhancing locally, unlike the usual contrast enhancing algorithm, which is performed globally. This tool is available in Fiji. The images were also converted from Red-Green-Blue (RGB) space to greyscale. They were first converted from RGB to 16-bit, and then to 8-bit. In this way the background was easily faded out.

4.4. 3D imaging

Three-dimensional imaging means capturing an object's shape in all three dimensions, *x, y* and *z*. Several different methods for this purpose can be used. Laser scanning, scanning with the help of physical contact and holography are some of the methods. However, this study concentrates on methods based on digital images. Several 3D construction and visualization software were tried or otherwise explored. These included e.g. Osirix, VTK, VV, Insight3D, Arius3D Pointstream Digital Imaging Software, DeVIDE, and Autodesk 123D Catch. Some of them are only targeted for visualization – like VTK – but some are capable of constructing a 3D image. Problem with many 3D construction programs is that they require slice images as input. This was not suitable for this project, because the casts could not be sliced. Only images from different viewpoints could be obtained. In this study, we consider two of them, namely imaging with structured light and stereophotogrammetric imaging.

4.4.1. Imaging with structured light

In methods based on structured light the object is illuminated with some light pattern. Light can be projected onto the object e.g. with the help of an ordinary video projector. A pixel's *z* coordinate can be then calculated from the pixel intensity. Several different strategies based on structured light exist. In this study we considered a method based on gray-level gradients.

One possible approach is to project gray-level gradients on a dental cast. This method is employed in Välisuo (2013). First the gradient goes from dark to bright, and this results in image I_{down}. Then the gradient is turned so that it goes from bright to dark, which results in image I_{up}. Because the intensity of each pixel depends not only on its position in *z* axel but also on the reflectance of the object, it is good to calibrate the image. The reflectance of the object can be calculated as the sum of the gradient images. Then the images can be calibrated by dividing them by that sum. (Välisuo 2013.)

For example, the calibrated version of I_{up} can be calculated in the following way

$$I = \frac{I_{up}}{I_{up} + I_{down}}. \tag{7}$$

Välisuo (2013) used the difference of the gradient images:

$$I = \frac{I_{up} - I_{down}}{I_{up} + I_{down}} \tag{8}$$

The z coordinate of a pixel can then be calculated with the help of the information of the imaging setup and the basic principle of triangulation. Fig. 14 illustrates the imaging setup and the angles and other parameters needed in triangulation. The change of colour dC and the change of coordinate y, dy can be calculated from the image and thus dz can be obtained as follows (Välisuo 2013):

$$dz = \frac{dC}{\cos(\varphi)} - dy \tag{9}$$

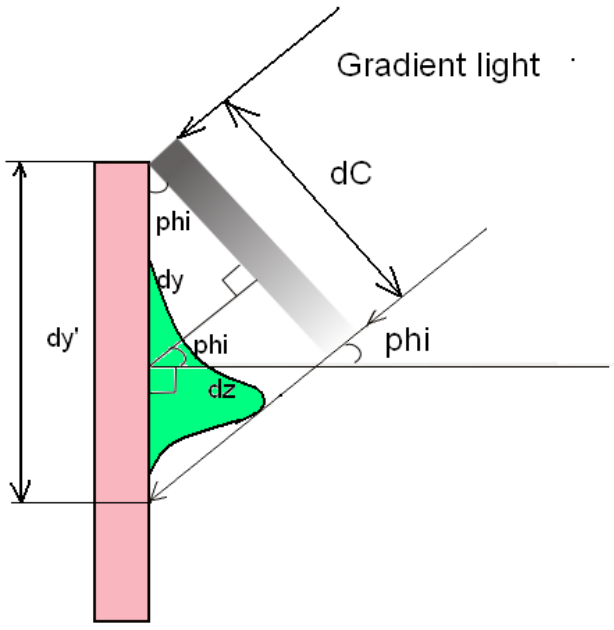

Figure 14. The imaging setup (Reproduction from Välisuo 2012).

The result of Välisuo's (2013) experiment showed that the shape of the cast was rather correct but the image contains also quite much noise and artifacts.

The advantage of using structured light is that the method is rather simple, and the z coordinate of a specific point can be calculated from one single image. However, if the object has such shape that all sides of it cannot fit into one image – which is also the case with dental casts – naturally several images must be taken. Then these images should be combined to get the whole 3D model, which would again require registration of the parts. Thus it can be concluded that the method based on structured light would perhaps be suitable for evaluating displacements in a small focused area of the cast instead of the whole arch.

4.4.2. Stereophotogrammetric imaging

Three-dimensional imaging based on stereophotogrammetry is a rather complex problem. Stereophotogrammetric imaging requires several images from different viewpoints. The process consists basically of two steps: determining camera parameters and computing a dense point cloud.

3D images can be obtained by photographing dental casts from several view angles and then photogrammetrically constructing a 3D surface from them. Images can be acquired by using one camera and rotating the object with the help of a turntable. In this study a turntable Thorlabs NR360S/M was used. Stereophotogrammetric imaging is suitable for especially capturing fine structures (Välisuo 2013), and thus it was worth testing.

An open-source package called Python Photogrammetry Toolbox was selected to be the 3D construction tool for this purpose. In Python Photogrammetry Toolbox (PPT) the 3D construction is done in two phases. First the so-called *Bundler* part is executed. Bundler is a Structure-from-Motion (SfM) system for unordered image collections. In Python Photogrammetry Toolbox, the main task of Bundler is to calculate the camera parameters and produce a sparse point cloud. Bundler uses the SIFT feature detection algorithm to do this. To obtain a denser point cloud, a tool called *CMVS/PMVS* is executed. This tool actually consists of two parts. Both CMVS and PMVS are multi-

view stereo (MVS) algorithms. Clustering views for Multi-view Stereo (CMVS) is a tool that helps to reduce the computation time by decomposing large sets of images into manageable-sized clusters. CMVS was actually developed for very large datasets, consisting of millions of images. Using it is not necessary when image sets are much smaller. However, it is important to perform the PVMS part. PMVS, Patch-based multi-view stereo algorithm is an algorithm that takes the outputs of Bundler as its inputs, and then computes the dense point cloud. (Furukawa *et al.* 2010a; Furukawa *et al.* 2010b).

PPT is a command-line driven program but it is also possible to install a graphical user interface, PPT-GUI, to facilitate its use. To be able to use PPT with PPT-GUI, some packages needed to be installed. First, Python 2.7 was installed to the computer. Next, PPT package was installed. Python Imaging Library (PIL) 1.1.7 was also installed. This was done easily with a Windows Installer obtained from the web. After this, SIP package (version 4.14.2), a tool for connecting C/C++ programs or libraries with Python, was installed. SIP configuration and installation was done with Visual Studio 2008 Command Prompt with commands `python configure.py`, `nmake` and `nmake install`. When SIP had been succesfully installed, PyQt4 package (version 4.9.6), which contains Python bindings for Qt GUI library, was installed. PyQt4 required a copy of Qt, which was also installed. Some DLL-related problems were encountered when trying to get PyQt4 working. Problems were apparently due to some conflict between files used by PyQt4 and MATLAB, and were fixed by editing the system *Path* variable.

Images were taken around the dental cast in every 30 degrees. First the images were taken with fluorescent tube light. The images were cropped so that only the dental casts were visible in them, because the non-rotating background makes the construction difficult. The *Bundler* part was executed first, and then the *PMVS* part. However, the construction was not successful, and the program gave an error message

```
sift.exe – Application Error
The instruction at “0x100100a0” referenced memory at “0x00000000”. The
memory could not be “written”.
```

This was probably because the PPT toolbox could not find enough features from the images. To get more details to the images, a different lightning was tried. A white screen was projected to the dental cast with a video projector (Fig. 15). This time the algorithm finished, but the result of the construction was very poor. Only some of the surroundings of the cast were included in the construction – not much of the cast itself (Fig. 16). The result was similar when a speckle pattern or a grid pattern was projected to the cast. Also the program gave a warning about not being able to determine the focal length of the cameras in pixels.

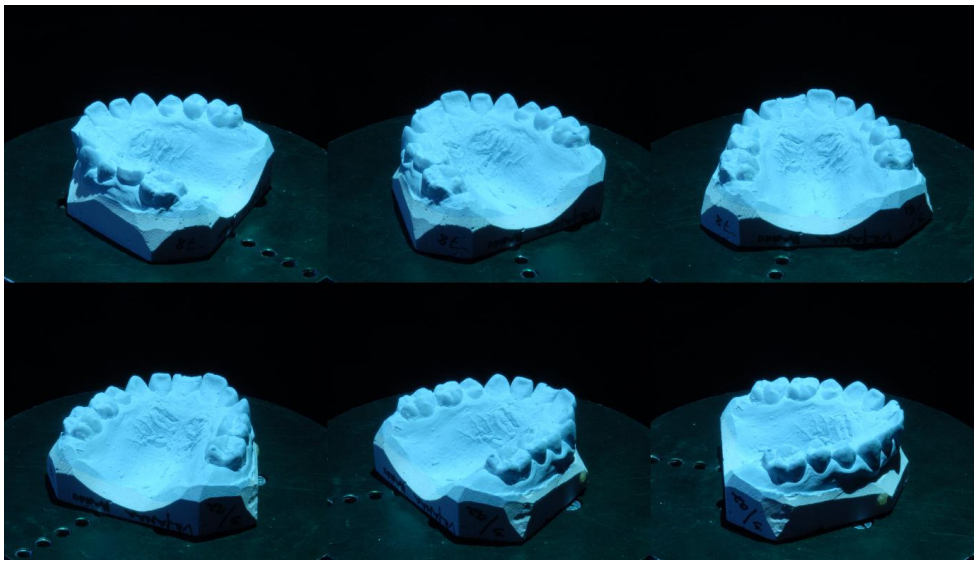

Figure 15. Six of the images used in the construction. The total number of images was 12.

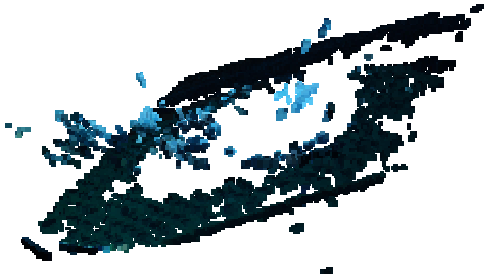

Figure 16. The result of 3D construction with Python Photogrammetry toolbox. Mostly background is included in the construction, not much of the cast itself.

5. MATERIALS AND IMPLEMENTED METHODS

In this chapter, the materials and methods used in this study are described. The material of the study consisted of eight dental casts from orthodontic patients. In the following sections the dental casts are described in a detailed way and the malocclusions of the patients are analyzed. Special challenges related to each cast pair are listed. The casts were imaged with the imaging methods described in Chapter 4. Then the registration and evaluation of changes were performed with methods described in this chapter.

5.1. Dental casts

In this study, dental casts from two patients were used. There were two cast pairs of both patients – one was taken of the lower jaw and one was taken of the upper jaw. In each pair one cast was taken before the treatment and the other was taken after the treatment. Both patients were at preadolescent age when the treatment was conducted. The casts of the patient #1 were relatively new and they had been stored at dental office. The casts of the patient #2 were much older and they had been stored at patient's home for about fifteen years, which had caused some wearing of the surface.

Accurate diagnoses of the types of the malocclusions were not available, so only some observations could be made. These observations are explained in the following.

5.1.1. Patient #1

The patient #1 had some crowding in his lower jaw, especially the lateral incisors were malposed. All the teeth of the lower jaw seemed to be lingually inclined. In his upper jaw the patient #1 had some gap between his central incisors, and both the central and lateral incisors were labially inclined. The whole right side of the upper arch seemed to be a bit buccally and labially inclined whereas the left side looked palatally inclined. The "before" and "after" pairs of the upper and lower jaw are shown in Fig. 17.

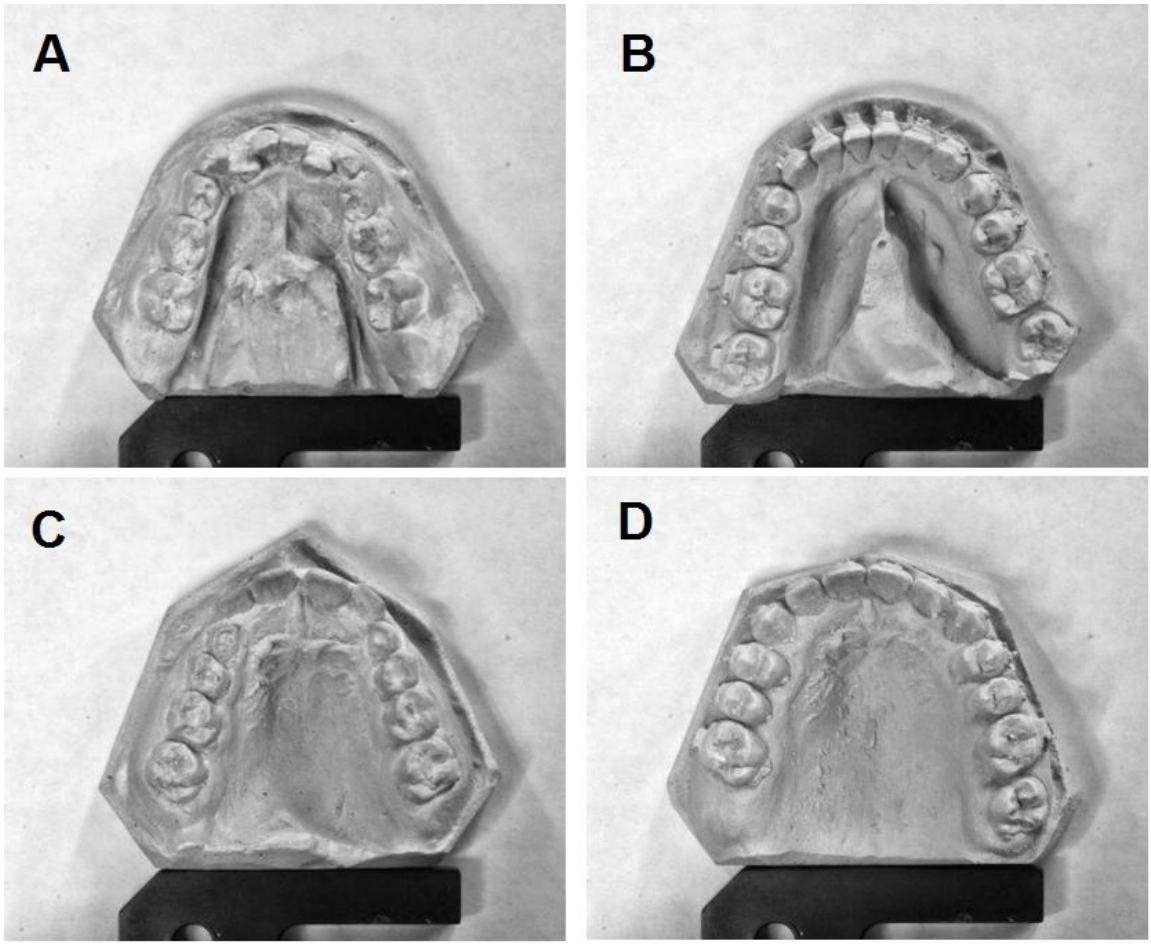

Figure 17. The casts of the patient #1. A. The lower jaw before the treatment. B. The lower jaw after the treatment. C. The upper jaw before the treatment. D. The upper jaw after the treatment.

In the lower jaw one tooth has erupted to both sides of the arch, so there are two more teeth in the "after" image than in the "before" image of the lower jaw. Those teeth are the second molars. Some of the other teeth have also changed from primary teeth to permanent teeth (Palo 2013). What makes the registration task even more challenging in the lower jaw is that the shape of the lingual area (the area of the tongue) is quite different between the two images. In the upper jaw again a second molar has erupted on the right side of the arch during the treatment, so there is one more teeth in the "after" image than in the "before" image of the upper jaw. Also the jaws of the patient have obviously grown during the treatment, which lasted for 2 years and 10 months. This maybe does not hamper the registration but makes the evaluation of the changes

challenging. Yet another issue is the colour of the casts: the “before” casts are different colour than the “after” casts. However, when the images were converted from RGB-scale to greyscale, the colour difference became almost invisible.

5.1.2. Patient #2

The patient #2 had a malposed left canine in his lower jaw. Also the left side of the lower arch was lingually inclined. The whole lower arch looks square-shaped and skew (asymmetric). In the upper jaw there are no clear malpositions of single teeth, but the whole arch is skew, too. Whereas the lower arch seems to be strongly tilted to the right, the upper jaw looks slightly tilted to the left. After the treatment both jaws look much more symmetric and also the squared form of the lower arch has changed to more oval form. The casts of the patient #2 are shown in Fig. 18.

As in the case of patient #1, the registration task included some special challenges. In the lower jaw a canine has erupted on the right side of the arch, so there is one more teeth in the “after” image than in the “before” image of the lower jaw. In the upper jaw no new teeth have erupted during the treatment. One tooth (right central incisor) has apparently fractured during the storage of the “after” cast of the upper jaw, but this probably does not pose a major problem to the registration since the fracture is quite small. Again also growth has taken place in both jaws, since the treatment lasted 2 years and 8 months. The colour of the casts is not a problem in this case since all the casts of this patient are of the same colour.

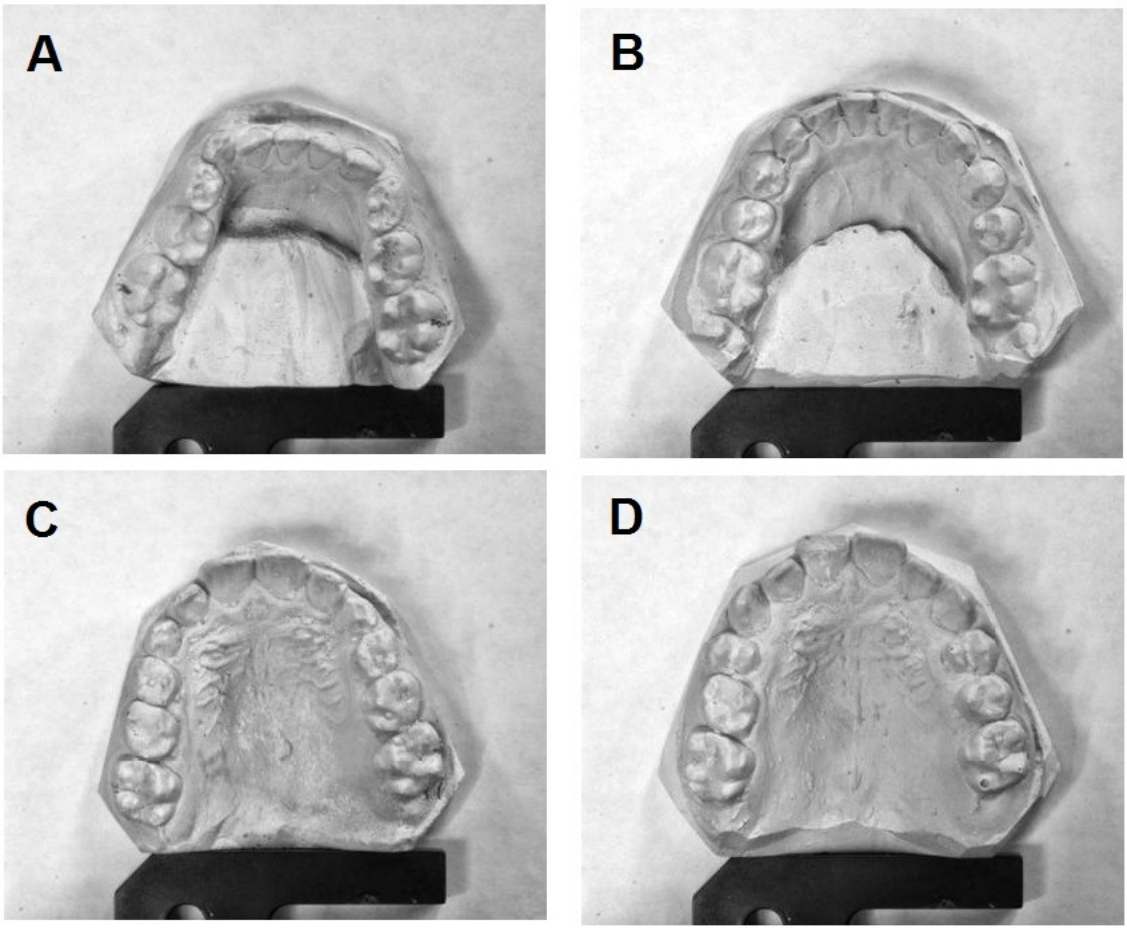

Figure 18. The casts of the patient #2. A. The lower jaw before the treatment. B. The lower jaw after the treatment. C. The upper jaw before the treatment. D. The upper jaw after the treatment.

5.2. Registration tools

The registration tool was Fiji plug-in *bUnwarpJ*, which is targeted for non-rigid and consistent registration of images. Consistency is achieved by using a consistency term in the energy function. In consistent registration both direct (source-to-target) and inverse (target-to-source) registrations are performed. The consistency term is then calculated by comparing how much an identity transform differs from a composed transform, which comprises of both direct and inverse transformation. (Arganda-Carreras *et. al.* 2006)

In addition to consistency term E_{cons}, there are four other terms in the function (Arganda-Carreras *et. al.* 2006):

$$E = w_i E_{img} + w_\mu E_\mu + (w_d E_{div} + w_r E_{rot}) + w_c E_{cons} . \quad (10)$$

In the *bUnwarpJ* user interface the user can give a weight for each term. E_{img} is the image term and w_i is its weight. This term represents the similarity between the source and target images. In this study, w_i was given value. E_u is the landmark weight which tells how strongly the algorithm tries to match the user-defined landmark locations of the image. The weight of this term, w_u, was given the value 3.0 in this study. The consistency term E_{cons} was given weight 10.0. The regularization terms, which were divergence (E_{div}) and curl (E_{rot}), were omitted from the function by giving them zero weight.

The registration tool accepts both user-defined landmark locations as well as features found by SIFT feature detection algorithm. The SIFT algorithm was applied to the images, but the results were not very good and thus only user-defined landmarks were used. We used one landmark per each tooth and one landmark next to each tooth (Fig. 19). No landmarks were placed to teeth which had erupted during the treatment.

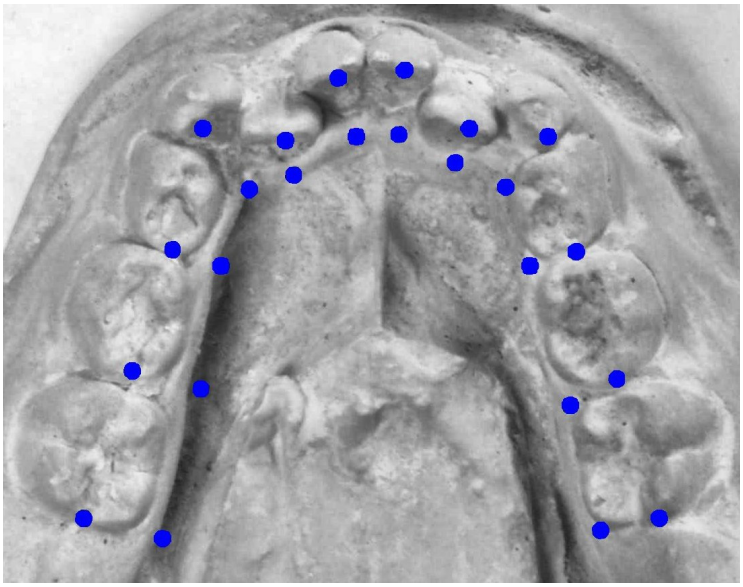

Figure 19. Registration landmarks.

B-splines and multiresolution strategy are utilized in *bUnwarpJ* in both image representations and deformation representations. The image taken before the orthodontic treatment was used as the source image, and the image taken after the treatment was used as the target image.

5.3. Estimation of the accuracy of the registration

As described in Section 3.6, there is no explicit, universal solution for determining the accuracy (correctness) of the registration. However, there are several simple, small tests which can be performed to get useful information of the correctness. These are *e.g.* manually inserted landmarks, consistency measure, difference image, Jacobian values, and – of course – eyeballing by naked eye.

From the methods listed above, the manually inserted landmarks provide the most detailed information of the accuracy of the registration. Thus it was chosen as the main method for accuracy estimation in this thesis. To make the landmark-based estimation easy, a Fiji macro was developed for that purpose (Appendix 1). A macro is a script which facilitates the use of some more complex framework – in this case Fiji. The macro language allows calling the functions of the registration tool – the Fiji plug-in *bUnwarpJ*. In this case the macro consisted of three main steps: drawing source landmarks with red color, applying the transformation, and drawing target landmarks with white color. This simple script facilitates the accuracy estimation process. The distances between the landmarks were measured with the ROI (region-of-interest) Manager of the Fiji software.

The validation landmarks were placed to different locations than in registration. This increases the reliability of the validation, since the registration algorithm tends to align the landmark locations more accurately than the other parts of the image. This way the registration was, in a way, cross-validated.

In addition to landmark-based accuracy estimation, another method based on lines and angles was utilized. This method utilizes the idea of Gulati, Kharbanda and Parkash's

(1998) approach, where the lines and angles of the dental casts are measured in order to determine the movements of teeth. In this study we use this approach to compare the proportions between the registered (deformed) source images and the original target images. This way we get information on how well the registration is able to match the distances and angles between certain points of the images. To separate this accuracy estimation method from landmark-based accuracy estimation, we call it parameter-based accuracy estimation. The measured error we call parameter-based error in contrast to landmark-based error.

In this study we used four linear, or distance-based, parameters, and two angular parameters for each cast. The linear parameters were measured from certain teeth to the central line of the cast. Those teeth were the first premolars and first molars. The angular parameters were measured as the angle between the central line and a line which passes through the distal surface of the first molar. All the parameters are shown and named in Fig. 20 and described in Table I. They were mostly the same as in Gulati *et al*. (1998).

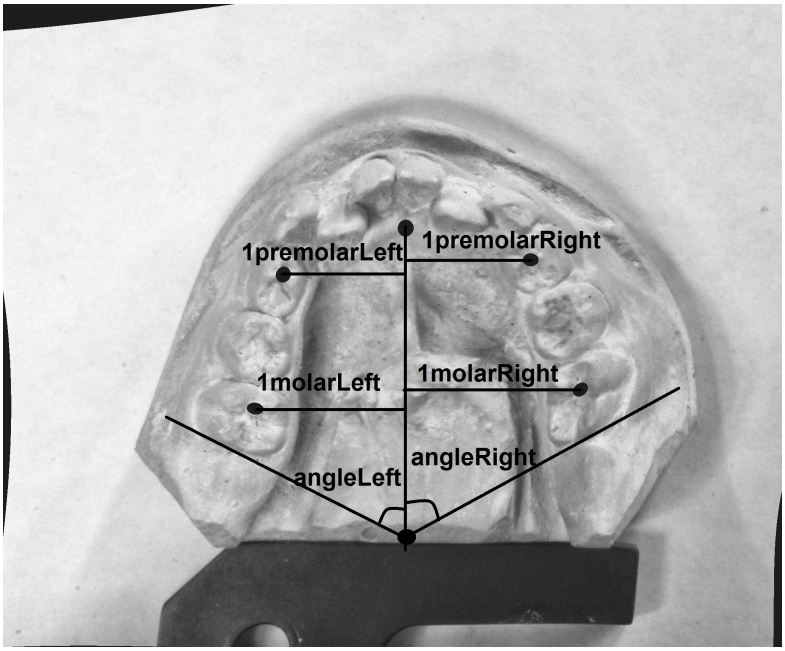

Figure 20. The linear and angular parameters

Table 1. The descriptions of the linear and angular parameters

Parameter	Description
1premolarLeft	Distance between the mesial pit of the left first premolar and the central line
1premolarRight	Distance between the mesial pit of the right first premolar and the central line
1molarLeft	Distance between the mesial pit of the left first molar and the central line
1molarRight	Distance between the mesial pit of the right first molar and the central line
angleLeft	Angle between the central line and a line which passes through the distal surface of the left first molar.
angleRight	Angle between the central line and a line which passes through the distal surface of the right first molar.

5.4. Estimation of changes

Estimation of the changes between images was done with the help of deformation-based morphometry. Deformation grid obtained from the registration was used for computing the ratio-of-areas estimates, which describe the amount of scale change between undistorted and distorted rectangles, as presented in Section 3.7. A grid image was combined with the deformed source image with the MIN (minimum) function of Fiji's Image calculator. This function calculates the minimum between two images.

A MATLAB program was developed for calculating the change estimates. A big part of the code was provided by researcher Vladimir Bochko. The code is presented in Appendix 2. The program binarized the grid image and segmented the cells chosen by the user. Number of pixels of each segmented cell was calculated and this number was then divided by the number of pixels in the original cell. The resulting ratio is the change estimate.

6. RESULTS

The aim of this study was to evaluate the accuracy of the registration and to find out how well the registration algorithm has managed to correct the malocclusions of the patients. The results were collected by calculating parameters and analyzing the deformations.

6.1. Patient #1

In the first experiment we used the images from the first imaging session, but soon it was discovered that the landmark errors were high, at most 150 pixels, and thus the registration was not as successful as hoped. We then continued the research by using the images taken during the second imaging session. Those images were taken from the occlusal view, *i.e*. straight from above the dental cast.

6.1.1. Accuracy of registration

With the new images, the registration of the lower jaw of the patient #1 was rather accurate (Fig. 21). The average landmark error was 41 pixels. The lowest errors were in the right central incisor, and in the right second premolar: 19 pixels. The poorest registration results were achieved at the right lateral incisor and right first molar, where the error was 74 pixels. Also in the left lateral incisor the error was high, 61 pixels. The reason for poor registration result in the lateral incisors is probably that some areas of them were not visible in the ‘after’ image of the cast. Thus the algorithm was not able to find the corresponding Y coordinates for those teeth. However, the algorithm did find the correct X coordinates quite well.

Although two teeth have erupted during the treatment, the last teeth of the ‘before’ image (the first molars) are vertically in correct positions. Probably the empty space behind them has helped in this.

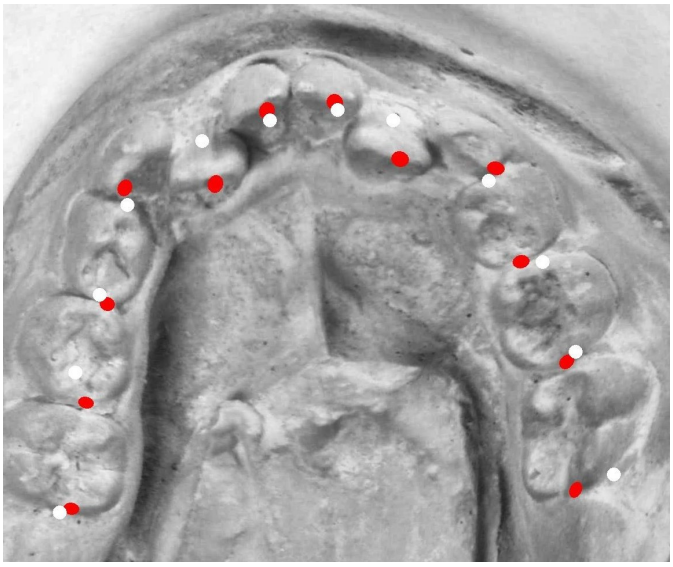

Figure 21. The landmark-based accuracy estimation of the registration of the lower jaw of patient #1. The white points are the landmark locations of the target image, *i.e.* they show the places where the algorithm *should* have shifted the landmark locations of the source image. The red points show where the registration algorithm *actually* shifted those locations. The distances between the red and white points thus tell the error of the registration.

The parameter-based accuracy evaluation was then performed to see how well the registration algorithm can estimate the distances and angles of the images. The original target image gives the correct values of the parameters, which we then compare to the values taken from the deformed source image. If the difference d_1 between those values is small, the registration has performed well. If the difference is big, the algorithm has failed to find the correct shape for the dental arch, or it has not estimated the growth of the tissues correctly. However, parameter measurements should not be used alone to analyze the accuracy of registration, but rather they should be interpreted together with the results of the landmark-based estimation.

In addition to this, the differences between the parameters of original target and original source were compared to the differences between the parameters of deformed source and original source. These differences were labelled d_2 and d_3, respectively. This comparison gave us the information if the registration algorithm found correctly the

direction of the change. If d_2 was positive and d_3 negative, the algorithm had shortened the distance when it should have increased it. On the contrary, if d_2 was negative and d_3 positive, the algorithm had increased the distance when it should have decreased it. Again if the both distances are positive or negative, the algorithm has taken the correct direction on that area.

The results of the parameter-based evaluation are listed in Table 2. The results show that some parts of the cast were better registered than others. The parameter-based error d_1 is low in other areas except in the left first premolar and right first molar. The error is negative, which means that the registration algorithm has not put enough space between the teeth and the central line. This is probably mostly due to the differences in the shape of the lingual and gingival area between the images. Near the right molar area, the plaster ridge in the lingual area is in totally different place in the 'before' image than in the 'after' image. The ridge produces a strong dark shadow, which the registration algorithm undoubtedly has detected and tried to match with the corresponding shadow in the target image. Also it seems that the algorithm has not corrected the orientation of the right first molar very well. In both the original source image and the deformed source image the molar is rotated. The rotation is even larger in the latter. The high landmark error in the right first molar supports this.

In the left premolar area the parameter-based error is high but the landmark-based error is low. From the images we can see that there is a rather big portion of gingiva which is visible in 'after' image but not in 'before' image. This probably has made the registration difficult in that area. Because the landmark-based evaluation shows that the tooth is approximately in a correct place, the central line must be in a wrong place. This suggests that in registration some landmarks should be placed also along the central line.

However, in most cases the algorithm has found the correct direction for the change. Only in two parameters the direction is wrong, but in those cases the error (d_1) is low.

Table 2. The parameter-based accuracy evaluation of the registration of the lower jaw of patient #1. Column d_1 gives the error of registration, *i.e*. the difference between the parameters of the original target image and the deformed source image. Column d_2 tells how much and to which direction the algorithm *should* have changed each parameter, and column d_3 tells how the registration *actually* changed each parameter.

Parameter	Deformed source image	Original target image	Original source image	d_1 (error)	d_2 (correct change)	d_3 (actual change)
1premolarLeft	324	357	285	-33	72	39
1premolarRight	336	333	255	3	78	81
1molarLeft	393	405	399	-12	6	-6
1molarRight	465	492	369	-27	123	96
angleLeft	60.3	64.5	62.7	-4.2	1.8	-2.4
angleRight	61.3	61.3	61.7	0	-0.4	-0.4

The accuracy of the registration was a bit better in the upper jaw than in the lower jaw. In the landmark-based evaluation of the registration (Fig. 22) the average error was 31 pixels. The lowest errors, 18 pixels, were located in the left lateral incisor, right central incisor and right first premolar. The highest error, 66 pixels, was located in the left canine, where the algorithm has failed to decrease the gap between the canine and left lateral incisor, which leaves the arch a bit square-shaped. Similar problem seems to have caused the error of 37 pixels also in the right lateral incisor. Also in the right second premolar the error was rather high, 42 pixels. The cause for this is probably local, or related to the failure to correct the squared shape of the arch. Surprisingly the right first molar is in a rather correct place although one more molar has erupted behind it.

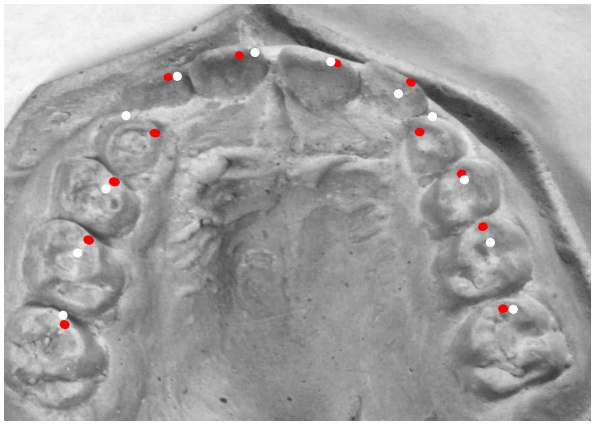

Figure 22. The landmark-based accuracy estimation of the registration of the upper jaw of patient #1.

The parameter-based accuracy evaluation showed that the registration had more difficulties in the right side than in the left side. Table 3 shows that in the right side the distances had an error of -26 and -30 pixels. The error is again negative, which means that the algorithm has made the distance between the central line and right-side teeth too short. Probably the algorithm has had difficulties to estimate the growth of the jaw in that area. The landmark-based estimation shows that the right-side teeth are in rather correct places, so the problem must be nearer the central line. However, the values of d_2 and d_3 show that the algorithm has found the correct direction for all the parameters.

Table 3. The parameter-based accuracy estimation of the registration of the upper jaw of patient #1.

Parameter	Deformed source image	Original target image	Original source image	d_1 (error)	d_2 (correct change)	d_3 (actual change)
1premolarLeft	411	419	327	-8	92	84
1premolarRight	399	425	363	-26	62	36
1molarLeft	501	509	441	-8	68	60
1molarRight	504	534	471	-30	63	33
angleLeft	61.2	61.6	62.3	-0.4	-0.7	-1.1
angleRight	59.9	59.8	62.7	0.1	-2.9	-2.8

6.1.2. Evaluation of changes

The evaluation of the changes was performed by inspecting the deformation grids. The grids were analyzed by calculating the change estimates *C*, *i.e.* the ratios between the areas of deformed and undeformed cells, as described in Section 3.7. Only the cells along the dental arch were chosen for evaluation.

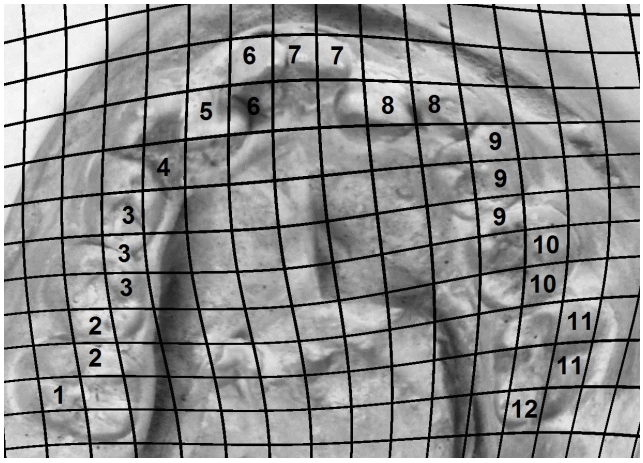

Figure 23. The cells chosen for evaluation of changes in the lower jaw of patient #1. The corresponding change estimates are shown in Fig. 24.

The cells chosen for evaluation of the lower jaw of patient #1 are shown in Figure 23. The corresponding change estimates are presented in Figure 24. The evaluation showed that in most of the cells the percentual change was over 0%. This means that those cells have expanded during the deformation. This suggests that the algorithm has tried to correct the lingual inclination of the teeth, which is good. The expansion was greatest in the area of incisors, especially in the lateral incisors. In that area the algorithm has tried to find the correct location for the malposed teeth.

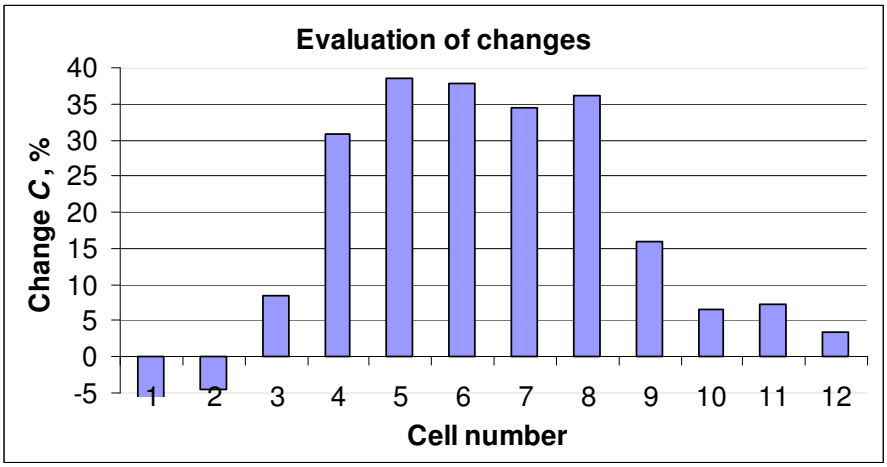

Figure 24. The change estimates of the lower jaw of patient #1. The cells are shown in Fig. 23.

The deformation grid of the upper jaw of patient #1 was quite different from the lower jaw grid (Fig. 25). Figure 26 shows that in most of the cells the change was less than 0%, which means that the cells have shrunk during the deformation. The shrinkage was greatest in the area of incisors, which shows that the algorithm has tried to correct the central gap and the labial inclination in that area. However, changes in the inclination of the entire right and left sides can not be clearly observed. The changes are only a bit bigger in the left side than in the right side. This is understandable because the inclination problem actually is not clearly visible from the viewpoint where the images have been taken.

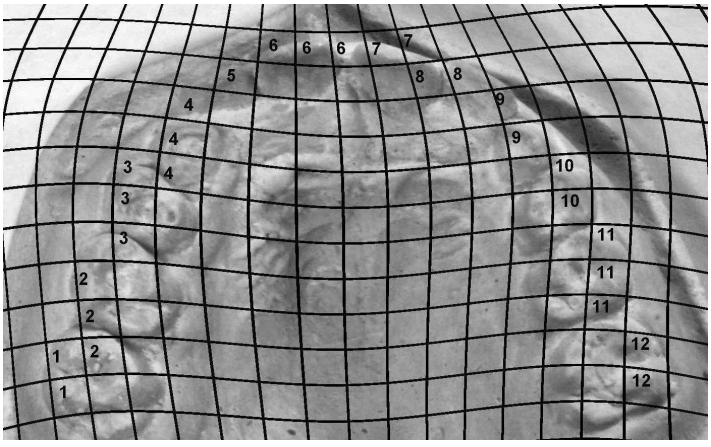

Figure 25. The deformation grid of the upper jaw of patient #1.

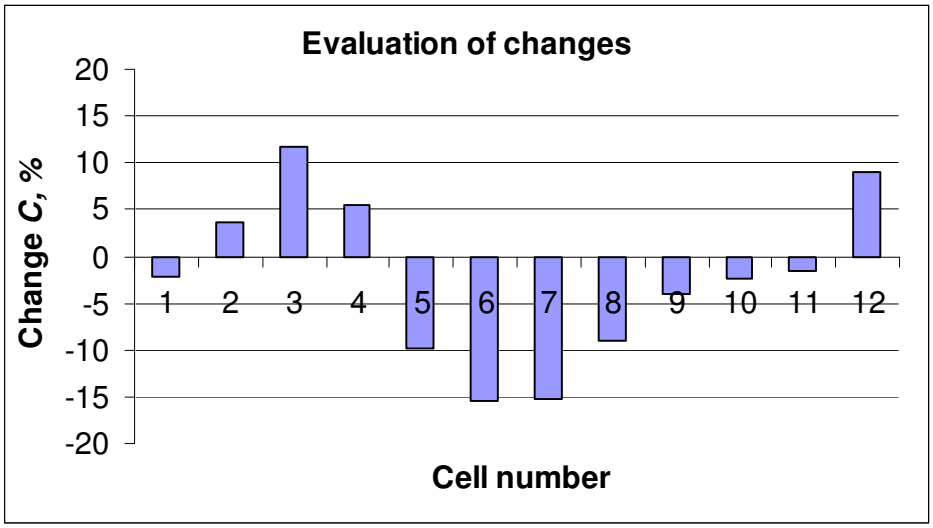

Figure 26. The change estimates of the upper jaw of patient #1.

6.2. Patient #2

6.2.1. Accuracy of registration

The landmark-based accuracy estimation of the lower jaw of the patient #2 showed that the registration was successful. The landmark-based errors are shown in Fig. 27. The average landmark-based error was 23 pixels. The lowest errors, 6 pixels, were located in the left central incisor and left canine. The highest error, 55 pixels, was located in the right lateral incisor. This error obviously is caused by failure to correct the distal inclination of that tooth. Also in the left first premolar as well as in the left first premolar the error was high, 30 pixels.

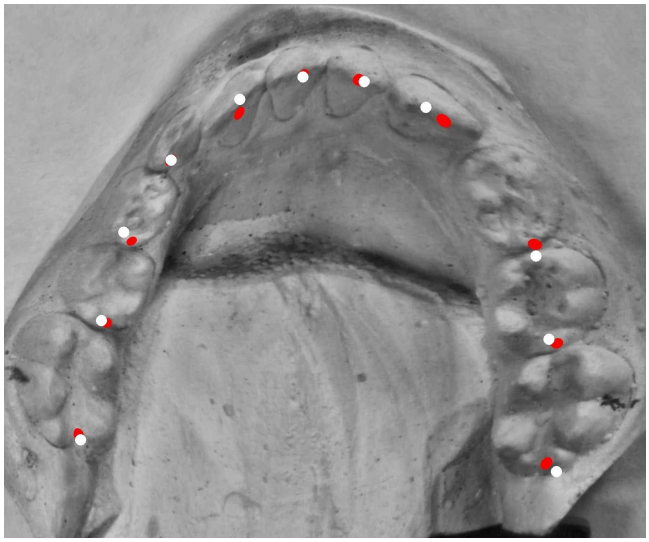

Figure 27. The landmark-based accuracy estimation of the registration of the lower jaw of patient #2.

The parameter-based accuracy evaluation gave also pretty good results. Table 4 shows that the error is rather low everywhere except in the left first premolar and right first molar. In the premolar area the error is negative, which means that the algorithm has made the distances too short in that area. Also the direction of change is wrong there. In the molar area the error is positive, which means that the distance is too large in that area. The visual inspection of the deformed image shows similar results: the anterior part of the palate seems to be too narrow, whereas the posterior part is too wide. Because the landmark-based error is low everywhere, the parameter-based error must be due to problems near the central line. This is understandable, because the 'before' cast was strongly skewed, and therefore it was difficult to place the central line correctly. Thus the parameter-based error does not necessarily tell only about inaccurate registration but also about inaccurate location of the central line.

Again also the lingual area probably causes problems for the registration. The shadow of the border of the plaster eminence has different shape and location in the two images.

Table 4. The parameter-based accuracy estimation of the registration of the lower jaw of patient #2.

Parameter	Deformed source image	Original target image	Original source image	d_1 (error)	d_2 (correct change)	d_3 (actual change)
1premolarLeft	348	387	351	-39	36	-3
1premolarRight	423	438	300	-15	138	123
1molarLeft	492	498	507	-6	-9	-15
1molarRight	507	468	375	39	93	132
angleLeft	64.6	65.3	69.4	-0.7	-4.1	-4.8
angleRight	67.4	66.7	79.4	0.7	-12.7	-12

The registration of the upper jaw of patient #2 showed good results in the landmark-based accuracy estimation. Figure 28 shows the landmarks. The average landmark-based error was only 25 pixels. The lowest error, 15 pixels, was located in the right lateral incisor. Also in the left canine the error was low, 18 pixels. The highest errors, which were 72 and 28 pixels, were detected at the right canine and left first premolar.

The parameter-based accuracy estimation showed that the registration performed badly in the right side of the cast. The error was positive, which means that the algorithm had made the distances between the central line and right-side teeth too long. The landmark-based errors in the right side are low, so again the problem must be nearer the central line. Probably the skewness of the jaw has made the registration difficult. The direction of the change was wrong in one parameter: in the right first premolar the distance should have been shortened by 23 pixels, but it was lengthened by one pixel.

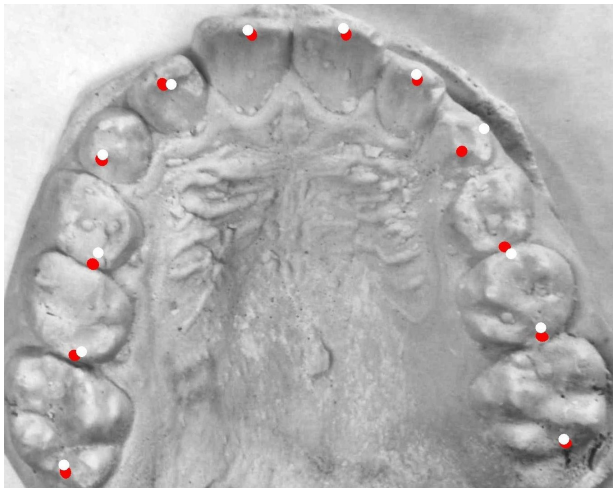

Figure 28. The landmark-based accuracy estimation of the registration of the upper jaw of patient #2.

Table 5. The parameter-based accuracy estimation of the registration of the upper jaw of patient #2.

Parameter	Deformed source image	Original target image	Original source image	d_1 (error)	d_2 (correct change)	d_3 (actual change)
1premolarLeft	453	447	374	6	73	79
1premolarRight	459	435	458	24	-23	1
1molarLeft	522	516	426	6	90	96
1molarRight	594	549	610	45	-61	-16
angleLeft	72.9	71.8	79.4	-1.1	-7.6	-6.5
angleRight	70.2	67	73.8	-3.2	-6.8	-3.6

6.2.2. Evaluation of changes

Again the deformation grids were analyzed to see if the algorithm has tried to correct the same things as the orthodontic treatment. The deformation grid of the lower jaw of patient #2 was somewhat difficult to interpret. The most obvious characteristic of the

change estimate diagram is that in the area of cells 8–10 the values are large when compared to the other cells. This is most probably due to the fact that during the treatment, a canine has erupted in that area, and thus the algorithm has created a lot of space there.

As noted in Section 5.1.2, the original untreated lower jaw was very asymmetric, and tilted to the right. Now from Figure 29 we can see that the deformation grid is rather skew, too. Figure 30 shows that in most cells of the left side there is shrinkage, whereas the right-side cells have expanded. Thus, any attempt to correct the lingual inclination of the left-side teeth can not be observed. Instead, the algorithm has tried to correct the overall skewness, as well as the squared shape of the arch.

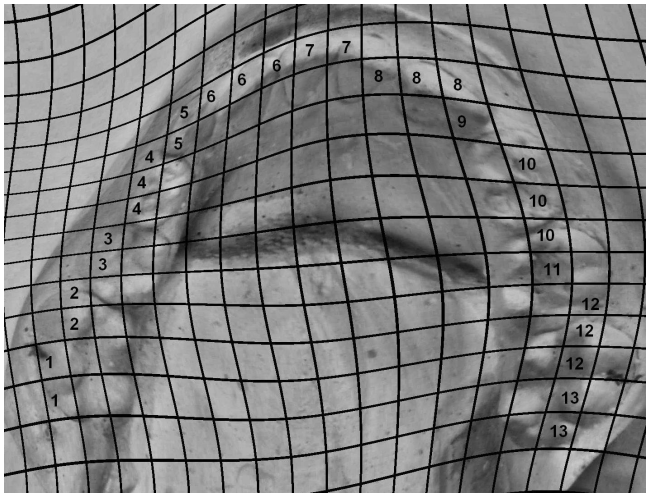

Figure 29. The deformation grid of the lower jaw of patient #2.

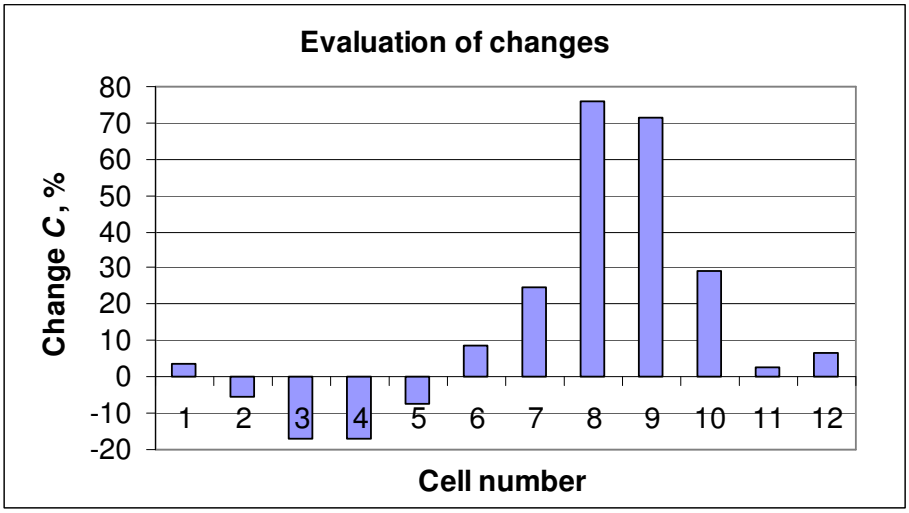

Figure 30. The change estimates of the lower jaw of patient #2.

The deformation grid of the upper jaw of patient #2 showed little change, as expected. The change estimate diagram in Figure 31 shows that all the cells have expanded – probably mostly due to the growth. The expansion is slightly bigger in the left side, where the average of the cells 1–6 is 23.6 %, than in the right side, where the average of the cells 9–14 is 15.7 %. However, the difference is small and does not tell much about the algorithm's attempt to correct the slight skewness of the arch. However, visual inspection of the deformation grid does show this attempt: the deformation field is clearly oriented to the left (Fig. 32).

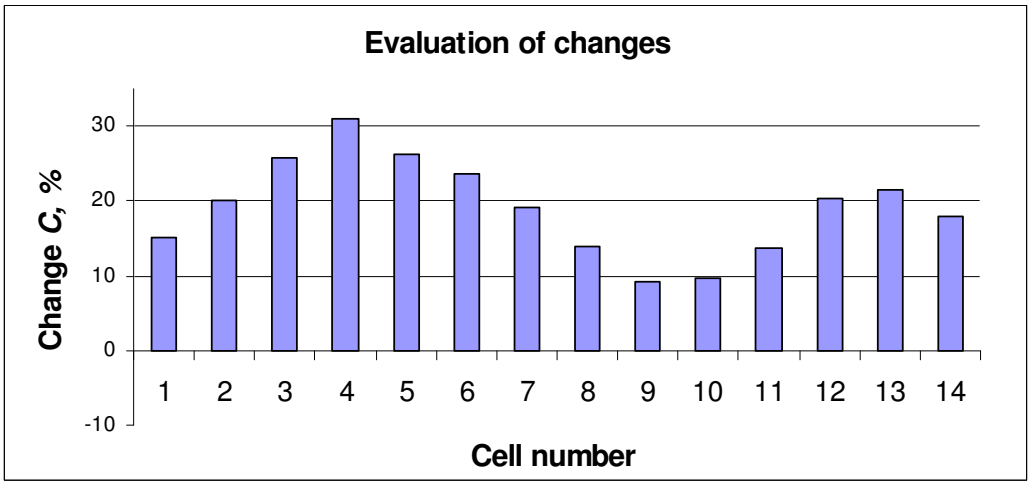

Figure 31. The change estimates of the upper jaw of patient #2.

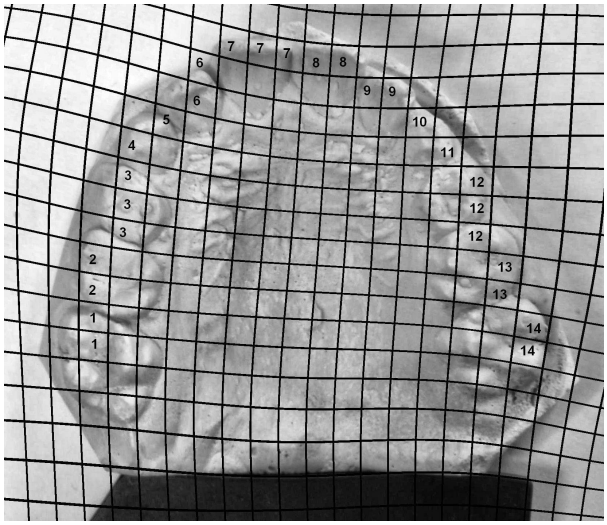

Figure 32. The deformation grid of the upper jaw of patient #2.

6.3. Attempts to improve the registration

The results showed that the cast images contain some parts which mislead the registration. Especially the lingual area of the lower jaw causes problems. This was tried to be fixed by removing the non-teeth area with a black mask with smooth edges. This was tested with the images of patient #1.

However, the accuracy of the registration was not better than without mask. In the lower jaw the landmark-based error even increased: the average error was 99 pixels. The highest error, 230 pixels, was located in the left second premolar and the lowest error, 24 pixels, in the right canine (Fig. 33). In the upper jaw the results were not much different than without mask. The average landmark-based error was 39 pixels. Like in the first registration, the highest error, 81 pixels, was located in the left canine, and the lowest, 12 pixels, in the left lateral incisor (Fig. 34). The parameter-based accuracy also did not show clear improvements. The parameter-based errors were only slightly lower with mask than without mask.

Also similar masked images with white and gray mask were tested, but the result was not better. The errors were especially high in the left side, as in the images with black mask.

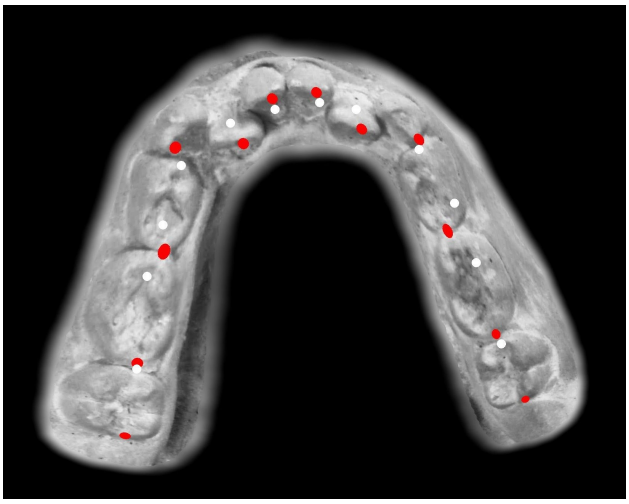

Figure 33. The landmark-based accuracy estimation of the masked lower jaw of patient #1.

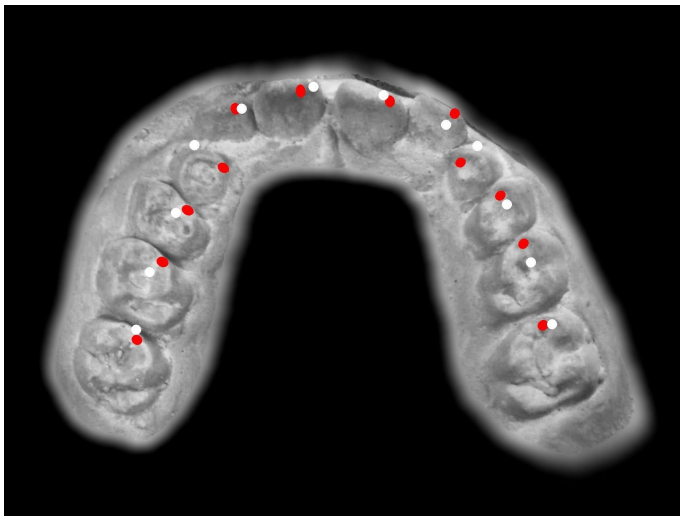

Figure 34. The landmark-based accuracy estimation of the masked upper jaw of patient #1.

Another approach to improve the registration is to concentrate on larger areas than landmarks. The teeth are rigid objects, but they move relative to each other and the jaw

during the treatment. To achieve better results it might be good to first register the individual teeth with a rigid or affine transformation and then perform the final non-rigid registration. In this study, an affine registration of individual teeth was experimented with a Fiji tool, but the result was poor. Probably better images and a more sophisticated algorithm is required for this purpose.

6.4. Simplifying registration

The registration process is also a bit tedious process. The insertion of the landmarks is a time-consuming task. We experimented, if it is possible to leave some of the registration landmarks away. We registered the lower jaw of patient #2 with landmarks only on teeth – not at all in the palate.

The landmark-based accuracy estimation showed that the registration was rather successful. The landmark-based errors are shown in Fig. 35. The average landmark-based error was 28 pixels. The lowest errors, 9 pixels, were located in the central incisors. The highest error, 64 pixels, was located in the right lateral incisor.

The parameter-based accuracy evaluation gave worse results than the landmark-based accuracy estimation. The error was rather high for all other parameters except for the left first molar and for the angular parameters. From the visual inspection we can also see that the correction of the squared shape of the arch has succeeded worse than in the registration with more landmarks.

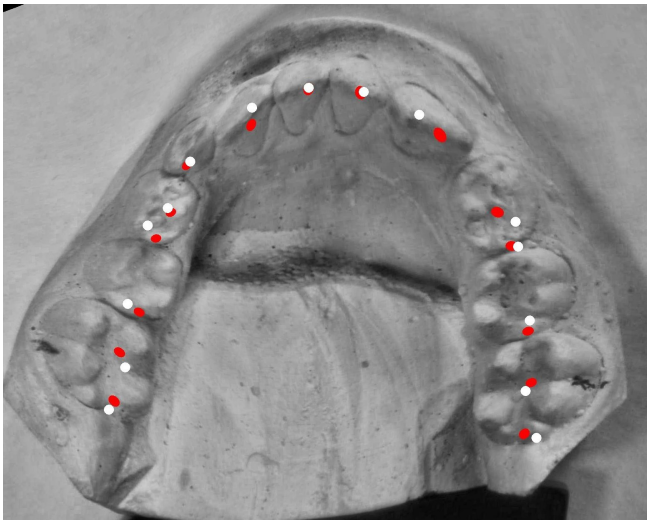

Figure 35. Landmark-based accuracy estimation of the registration of lower jaw of patient #2. The registration was performed with landmarks only on teeth.

Table 6. The parameter-based accuracy estimation of the registration of the lower jaw of patient #2. The registration was performed with landmarks only on teeth.

Parameter	Deformed source image	Original target image	Original source image	d_1 (error)	d_2 (correct change)	d_3 (actual change)
1premolarLeft	363	387	351	-24	36	12
1premolarRight	405	438	300	-33	138	105
1molarLeft	507	498	507	9	-9	0
1molarRight	492	468	375	24	93	117
angleLeft	64.7	65.3	69.4	-0.6	-4.1	-4.7
angleRight	70.1	66.7	79.4	3.4	-12.7	-9.3

7. CONCLUSIONS AND FUTURE WORK

The goal of this research was to find out, whether the non-rigid registration of dental casts can be used in the evaluation of orthodontic treatment and to design a program, which would at least partially automatize the evaluation process of 2D images and be easy to use. The aim was also to experiment the evaluation with 3D models of the casts. This research was delimited to cover only evaluation of malocclusions within one dental arch.

The accuracy of non-rigid registration of dental casts is within reasonable error limits, if the algorithm is performed with landmarks inserted by a human, and the images are taken from the occlusal view, *i.e.* straight from above. The landmarks can be placed to each teeth and to palatal/lingual area near each teeth. With these arrangements, the average landmark-based error varied between 25–41 pixels. Without these arrangements the registration had difficulties to find the correct solution, and the average landmark-based error was at most 150 pixels. The parameter-based accuracy estimation showed that to achieve even better results, some landmarks should also be placed along the central line of the cast.

Deformation-based morphometrical measurements showed that the movement of teeth can be coarsely detected by using Jacobian-like measures of change. The registration had corrected especially the overall skewness of the arch, some gaps and inclinations and some of the shape of the arch. However, to achieve even better results, the algorithm seemingly needs more landmarks in registration. One different approach is to change the whole registration so that the algorithm concentrates on individual teeth instead of landmarks. This approach was quickly tested with an affine registration tool of Fiji, but was not successful. Perhaps some different algorithm might work better. For example the focused mutual information (FMI) algorithm developed by Jacquet, Nyssen, Bottenberg, Truyen and de Groen (2009) might give better results.

The eruption of teeth during the treatment does not seem to hamper the registration seriously, if the teeth have erupted behind the last teeth – which was the case in the casts of patient #1. If the extra teeth have erupted between other teeth, like in the lower jaw of

patient #2, it affects more negatively to the accuracy of registration. From other challenges especially the differences in the lingual area of the lower jaw causes some errors in registration.

Automatic and semi-automatic tools for treatment evaluation were produced during the project. Estimation of the accuracy of the registration was facilitated by writing a Fiji macro, which automatizes the drawing of landmarks and deforms the source image. A MATLAB program for computing the change estimates from the deformation grid cells was also written.

Three-dimensional imaging of the casts was unsuccessful, and thus the development of 3D evaluation system was left as a future research topic. The registration of 3D images might be performed by using Elastix toolbox, which registers 3D raster images which are e.g. in .DICOM format. However, before registration the point cloud files of the 3D models must be transformed into raster format. This should be performed with some voxelisation algorithm (Jones & Satherley 1996). Further, because 3D model contains only the surface of the object, the inside of the cast should be filled with some constructive solid geometry algorithm. On the other hand, if only point cloud files can be obtained, one alternative is to register point cloud files by Coherent Point Drift algorithm, which is non-rigid version of Iterative Closest Point algorithm (Myronenko & Song 2009).

In the future, more dental casts are needed for testing the evaluation system developed in this thesis. Also more accurate diagnosis need to be done to find out how the treatment of different types of malpositions can be evaluated. Developing a 3D evaluation system is also an interesting research topic.

REFERENCES

Andrilli, Stephen & David Hecker (2010). *Change of Variables and the Jacobian* [online]. [Cited 30.11.2013]. Available: <URL: http://booksite.academicpress.com/andrilli/elementary/content/jacobian.pdf>. Supplementary material for a textbook named *Elementary Linear Algebra* (Andrilli & Hecker 2010).

Andronache, A., M. von Siebenthal, G. Székely & Ph. Cattin (2007). Non-rigid registration of multi-modal images using both mutual information and cross-correlation. *Medical Image Analysis* 12:1, 3–15. doi:10.1016/j.media.2007.06.005.

Arganda-Carreras, I., C. O. S. Sorzano, R. Marabini, J. M. Carazo, C. Ortiz-de-Solorzano & J. Kybic (2006). Consistent and elastic registration of histological sections using vector-spline regularization. In: *CVAMIA: Computer Vision Approaches to Medical Image Analysis*, volume 4241/2006, 85–95. Berlin: Springer-Verlag Berlin Heidelberg. doi:10.1007/11889762_8

Arganda-Carreras, I., C. O. S. Sorzano, P. Thévenaz, A. Muñoz-Barrutia, J. Kybic, R. Marabini, J. M. Carazo & C. Ortiz-de-Solorzano (2010). Non-rigid consistent registration of 2D image sequences. *Physics in Medicine and Biology*, 55:20, 6215-6242. doi:10.1088/0031-9155/55/20/012

Aung, Min S. H., John Y. Goulermas, Shaheen Hamdy & Maxine Power (2010). Spatiotemporal visualizations for the measurement of oropharyngeal transit time from videofluoroscopy. *IEEE Transactions on Biomedical Engineering* 57:2, 432–441. doi:10.1109/TBME.2009.2019828.

Avery, James K. (1992). Periodontium: Alveolar Process and Cementum. In: *Essentials of Oral Histology and Embryology*: *A Clinical Approach.* Ed. Pauline F. Steele. St. Louis: Mosby. 224 p. ISBN: 0-8016-5868.

Badshah, Amir, Paul O'Leary, Matthew Harker & Christian Sallinger (2011). Non-rigid registration for qualitiy control of printed materials. In: *Proceedings of the SPIE 8000*, Volume 8000, pp. 80000J–80000J-10. Ed. Jean-Charles Pinoli, Johan Debayle, Yann Gavet, Frédéric Gruy, Claude Lambert. Saint-Etienne: . doi:10.1117/12.890901

Battezzato, Alessandro, Laura Gastaldi & Stefano Pastorelli (2011). Evaluation of the factors affecting the optimal fiducial configurations calculated through a genetic-algorithm-based methodology in image-guided neurosurgery. *The International Journal of Medical Robotics and Computer Assisted Surgery* 7:4, 441–451. doi: 10.2319/041709-219.1

Bholsithi, Wiharut, Natthawadee Phichitchaiphan & Chanjira Sinthanayothin (2010). Online dental information database for dental identification system. In: *ICCIT 2010, 5th International Conference on Computer Sciences and Convergence Information Technology*, 278-283. Seoul: IEEE Xplore. doi: 10.1109/ICCIT.2010.5711071. ISBN 978-1-4244-8567-3.

Bro-Nielsen, Morten, Claus Gramkov & Sven Kreiborg (1997). Non-rigid image registration using bone growth model. In: *CVRMed-MRCAS'97*, 1–12. Ed. Jocelyne Troccaz, Eric Grimson & Ralph Mösges. Berlin: Springer-Verlag Berlin Heidelberg. doi: 10.1007/BFb0029219

Buder, Maximilian (2012). Dense realtime stereo matching using a memory efficient Semi-Global-Matching variant based on FPGAs. In: *Proceedings Real-Time Hardware*. Brussels: EuroSPIE. doi:10.1117/12.921147

Cambridge in Colour (2013). *Camera exposure* [online]. [Cited 30.1.2013]. Available: <URL: http://www.cambridgeincolour.com/tutorials/camera-exposure.htm>.

Chung, M.K., K.J. Worsley, T. Paus, C. Cherif, D.L. Collins, J.N. Giedd, J.L. Rapoport & A.C. Evans (2001). A unified statistical approach to deformation-based morphometry. *NeuroImage* 14:3, 595-606. ISSN 1053–8119, doi:10.1006/nimg.2001.0862.

Crum, W. R., T. Hartkens & D. L. G. Hill (2004). Non-rigid image registration: theory and practice. *The British Journal of Radiology* 77:2004, 140–153 doi: 10.1259/bjr/25329214.

Cong, Jason, Muhuan Huang & Yi Zou (2011). Accelerating fluid registration algorithm on multi-FPGA platforms. In: *Proceedings of the 2011 21st International Conference on Field Programmable Logic and Applications*, p. 50–57. doi:10.1109/FPL.2011.20

Damas S., O. Cordón & J. Santamaria (2011). Medical image registration using evolutionary computation: An experimental survey. *IEEE Computational Intelligence Magazine* 6:4 [Cited 26.9.2012], 26–42.

Dandekar, Omkar & Raj Shekhah (2007). FPGA-accelerated deformable image registration for improved target-delineation during CT-guided interventions. *IEEE Transactions on Biomedical Circuits and Systems* 1:2, 117–127. doi:10.1109/TBCAS.2007.909023

de Boor, Carl (1978). *A Practical Guide to Splines*. In: Applied Mathematical Sciences 27. New York: Springer-Verlag. ISBN: 978-0-387-95366-3. 392 p.

Dosselmann, Richard & Yang, Xue (2011). A comprehensive assessment of the structural similarity index. *Signal, Image and Video Processing* 5:1, 81–91. ISSN: 1863-1703 Doi: 10.1007/s11760-009-0144-1

Fookes, C. & A. Maeder (2003). Comparison of popular non-rigid image registration techniques and a new hybrid mutual information-based fluid algorithm. In: *2003 APRS Workshop on Digital Image Computing*. Brisbane: WDIC.

Furukawa, Yasutaka & Jean Ponce (2010a). Accurate, dense, and robust multi-view stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32:8, 1362-1376.

Furukawa, Yasutaka, Brian Curless, Steven M. Seitz & Richard Szeliski (2010b). Towards internet-scale multi-view stereo. In: *CVPR 2010*, p. 1434-1441. San Fransisco: IEEE.

Gibbs, J.W. (1899). Fourier Series. *Nature* 59, 606.

Gulati, Sumit, O.P. Kharbanda & Hari Parkash (1998). Dental and skeletal changes after intraoral molar distalization with sectional jig assembly. *American Journal of Orthodontics and Dentofacial Orthopedics* 114:3, 319-327. doi: 10.1016/S0889-5406(98)70215-X, ISSN 0889-5406.

Haavikko, Kaarina (1985). Development of dentition. In: *Introduction to Orthodontics*, p.45-60. Ed. Birgit Thilander & Olli Rönning. 5th edition. Stockholm: Tandläkarförlaget. 295 p. ISBN: 95-85174-17-3.

Hearn, Donald & Pauline M. Baker (1997). *Computer Graphics, C version*. New Jersey: Prentice Hall. ISBN: 0-13-530924-7. 652 p.

Horton, Heather M. I., James R. Miller, Philippe R. Gaillard & Brent E. Larson: Technique comparison for efficient orthodontic tooth measurements using digital models. *Angle Orthodontist* 2010 80:2, 254–261.

Ibáñez, Oscar, Oscar Cordón, Sergio Damas & José Santamaría (2011). An advanced scatter search design for skull-face overlay in craniofacial superimposition. *Expert Systems with Applications* 39:2012, 1459–1473. doi:10.1016/j.eswa.2011.08.034

Insight Software Consortium (2003). *ITK Software Guide.* [online] [cited 26.9.2012]. Available: <URL: http://www.itk.org/ItkSoftwareGuide.pdf>.

Jacquet, W., E. Nyssen, P. Bottenberg, B. Truyen & P. de Groen (2009). 2D image registration using focused mutual information for application in dentistry. *Computers in biology and medicine* 39:6, 545-553.

Jones, Mark W. & Richard Satherley (1996). Voxelisation: Modelling for Volume Graphics. *Computer Graphics Forum* 15:5, 311–318.

Klein, Stefan & Martin Staring (2012). *Elastix – the manual* [online] [Cited 26.9.2012]. Available: <URL: http://elastix.isi.uu.nl>.

Knisley, Jeff & Kevin Shirley (2001). *The Jacobian* [online]. [Cited 1.12.2013]. Available: <URL: http://math.etsu.edu/multicalc/prealpha/Chap3/Chap3-3/printversion.pdf>. Supplementary material for a textbook named *Calculus: A Modern Approach* (Knisley & Shirley 2001).

Koljonen, Janne (2010). *Computer Vision and Optimization Methods Applied to the Measurements of In-Plane Deformations*. Vaasa, Finland: University of Vaasa, Faculty of Technology. PhD thesis. ISBN: 978-952-476-295-3. 209 p.

Loza, A., L. Mihaylova, L., N. Canagarajah, N. & D. Bull. (2006) Structural similarity-based object tracking in video sequence. In: *Information Fusion, 2006 9th International Conference on*. ISBN 1-4244-0953-5 doi: 10.1109/ICIF.2006.301574.

Loi, Gianfranco, Marco Dominietto, Irene Manfredda, Eleonora Mones, Alessandro Carriero, Eugenio Inglese, Marco Krengli & Marco Brambilla (2008). Acceptance test of a commercially available software for automatic image registration of computed tomography (CT), magnetic resonance imaging (MRI) And ^{99m}Tc-methoxyisobutylisonitrile (MIBI) single-photon emission computed tomography (SPECT) brain images. *Journal of Digital Imaging* 21:3, 329–337. doi: 10.1007/s10278-007-9042-7

Lundström, Anders (1985). Malocclusion of the teeth: classification, prevalence and treatment need. In: *Introduction to Orthodontics*, p.85-112. Ed. Birgit Thilander & Olli Rönning. 5th edition. Stockholm: Tandläkarförlaget. 295 p. ISBN: 95-85174-17-3.

May, Michael, Martin Turner & Tim Morris (2011). Analysing false positives and 3D structure to create intelligent thresholding and weighting functions for SIFT features. In: *PSIVT'11 Proceedings of the 5th Pacific Rim conference on Advances in Image and Video Technology – Volume Part I*, 190-201. Berlin: Springer-Verlag Berlin Heidelberg. doi:10.1007/978-3-642-25367-6_17

Myronenko, Andriy & Xubo Song (2009). Point Set Registration: Coherent Point Drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32:12, 2262–2275.

Nassar, Diaa Eldin, Mythili Ogirala, Donald Adjeroh & Hany Ammar (2006). An efficient multi-resolution GA approach to dental image alignment. In: *Proceedings of SPIE-IS&T Electronic Imaging*, Vol. 6064.

Netter, Frank H. (1989). *Atlas of Human Anatomy*. Ciba-Ceigy Corporation. ISBN: 0-914168-19-3.

Okada, Keita & Fumihiko Saitoh (2011). High-speed image matching using partial template consisting of multiple rectangular areas extracted by genetic algorithm. *Electronics and Communications in Japan* 94:10, 696–703.

Palo, Katri (2013). Acting Chief Dental Officer, City of Vaasa. Interview 25.4.2013. Interviewer: Suvi Karhu. Notes available from interviewer.

Proffit, William R., Henry W. Fields, James L. Ackerman, Peter M. Sinclair, Paul M. Thomas & J.F. Camilla Tulloch (2004). *Contemporary Orthodontics*. 2nd edition. St Louis: Mosby Year Book. 668 p. ISBN: 0-8016-6393-8.

Riddle, William R., Rui Li, Michael Fitzpatrick, Susan DonLevy, Benoit M. Dawant & Ronald R. Price (2004). Characterizing changes in MR images with color-coded Jacobians. *Magnetic Resonance Imaging* 22:6, 769–777. ISSN: 0730-725X, doi:10.1016/j.mri.2004.01.078.

Rueckert, D., L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach & D. J. Hawkes (1999). Nonrigid registration using free-form deformations: application to breast MR images. *IEEE Transactions on Medical Imaging*, 18:8, 712–721. doi: 10.1109/42.796284

Russakoff, Daniel B., Carlo Tomasi, Torsten Rohlfing & Calvin R. Maurer, Jr. (2004). Image similarity using mutual information of regions. In: *Computer Vision - ECCV 2004,* p. 596–607. Ed. Pajdla, Tomáš & Jiří Matas. Berlin: Springer-Verlag Berlin Heidelberg. doi:10.1007/978-3-540-24672-5_47

Rönning, Olli (1985). Development of the occlusion. In: *Introduction to Orthodontics*, p.63-83. Ed. Birgit Thilander & Olli Rönning. 5th edition. Stockholm: Tandläkarförlaget. 295 p. ISBN: 95-85174-17-3.

Santamaría J., O. Cordón & S. Damas (2011). A comparative study of state-of-the-art evolutionary image registration methods for 3D modeling. *Computer Vision and Image Understanding* 115:9, 1340–1354. doi:10.1016/j.cviu.2011.05.006

Savitzky, A. & M. J. E Golay. (1964). Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry* 36:8, 1627–1639. doi:10.1021/ac60214a047.

Simplyteeth (2012). *The Anatomy of Teeth and Jaws* [online]. [Cited 15.3.2013]. Available: <URL: http://www.simplyteeth.com/category/sections/adult/aboutteeth/anatomy.asp?category=adult&section=1&page=1>.

Sinthanayothin, Chanjira, Natthawadee Phichitchaiphan, Nonlapas Wongwaen & Wisarut Bholsithi (2010). System for archiving, communication and analyzing of 3D dental cast model. In: *ICEIE 2010, International Conference On Electronics and Information Engineering*, p. V1-258–V1-262. Kyoto: IEEE Xplore. doi: 10.1109/ICEIE.2010.5559881, ISBN: 978-1-4244-7679-4.

Välisuo, Petri (2013). Experimentation report, preliminary version. Vaasa: University of Vaasa. Cited 31.1.2013.

Wang, Z., A. C. Bovik, H. R. Sheikh & E. P. Simoncelli (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13:4, 600–612. ISSN:1057-7149. doi: 10.1109/TIP.2003.819861

Yamamoto, K., S. Hayashi, H. Nishikawa, S. Nakamura & T. Mikami (1991). Measurements of dental cast profile and three-dimensional tooth movement during orthodontic treatment. *IEEE Transactions on Biomedical Engineering*, 38:4, 360–365. doi: 10.1109/10.133232.

Zhao, Jia-qing, Pan Zeng, Li-Ping Lei & Yuan Ma (2011). Initial guess by improved population-based intelligent algorithms for large inter-frame deformation measurement using digital image correlation. *Optics and Lasers in Engineering* 50:2012, 473–490. doi:10.1016/j.optlaseng.2011.10.005

APPENDICES

APPENDIX 1. The program code of landmark-based accuracy estimation

This appendix contains main parts of the code which was used in the landmark-based accuracy estimation. First the Fiji macro code is presented. This macro calls two self-implemented plugins: `DrawSourceLandmarks_red_v3_.java` and `DrawSourceLandmarks_white_v3_.java`. The `DrawSourceLandmarks_red_v3_` plugin is presented in Appendix 1.2. The `DrawSourceLandmarks_white_v3_` is not presented since it is very similar to the Appendix 1.2 code, except it uses white color and target landmarks instead of red color and source landmarks.

1.1. The Fiji macro code

This Fiji macro code allows the user to select a source image, a transform file and a landmark file. Then it calls three Fiji plugins which perform three tasks: first draw the source landmarks to the source image, then deform the source image and finally draw the target landmarks to the source image.

```
path_source = File.openDialog("Select the source image");
open(path_source); // open the file
dir = File.getParent(path_source);

path_transform = File.openDialog("Select the transform file");

path_landmark = File.openDialog("Select the landmark file");

circleRadius="12";

path_red = dir+File.separator+"red_landmarks_FROM_DODRAWING.jpg";
call("draw_sourceLandmarks.DrawSourceLandmarks_red_v3_.drawSourceLandm
arks_red_Macro", path_source, path_landmark, path_red, circleRadius);

path_output=dir+File.separator+"red_landmarks_deformed_FROM_MACRO.jpg"
;

call("bunwarpj.bUnwarpJ_.elasticTransformImageMacro", path_red,
path_red, path_transform, path_output);

path_white = dir+File.separator+"white_landmarks_FROM_DODRAWING.jpg";
```

```
call("draw_targetLandmarks.DrawTargetLandmarks_white_v3_.drawTargetLan
dmarks_white_Macro", path_output, path_landmark, path_white,
circleRadius);
```

1.2. The source landmark drawing code DrawSourceLandmarks_red_v3_.java

The following Fiji plugin reads landmarks from a file and then draws them to an image. Plugin can be used through the Fiji user interface, from command line or through a Fiji macro.

```
package draw_sourceLandmarks;
import ij.IJ;
import ij.ImagePlus;
import ij.io.OpenDialog;
import ij.plugin.filter.PlugInFilter;
import ij.process.ImageProcessor;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.InputStreamReader;
import java.util.StringTokenizer;

public class DrawSourceLandmarks_red_v3_ implements PlugInFilter {
          ImagePlus imp;

          public int setup(String arg, ImagePlus imp) {
                    this.imp = imp;
                    return DOES_ALL;
          }

          public void run(ImageProcessor ip) {
                    ip.invert();
                    imp.updateAndDraw();
                    IJ.wait(500);
                    ip.invert();
                    imp.updateAndDraw();
                    OpenDialog od=new OpenDialog("Choose landmark
file - Valitse Landmark-tiedosto",
"C:\\Users\\Suvi2\\Documents\\dityo\\ orthoped\\", null);
                    int[][] xy_data=new int[30][2];
                    String path=od.getDirectory()+od.getFileName();
                    IJ.showMessage(path);

                    xy_data=LueKoordinaattiTiedosto(path, 1,false);

                    ip.setColor(java.awt.Color.red);
                    for(int i=0;i<xy_data.length;i++)
```

```
                        {

                        ip.fillOval(xy_data[i][0],xy_data[i][1],20,20);
                        }
                        imp.updateAndDraw();
            }

            public static String doDrawingCommandLine(String[] args)
            {
                        String imagePath=args[1];
                        String landmarkPath=args[2];
                        String newFilePath=args[3];
                        int rad=Integer.parseInt(args[4]);

                        ImagePlus imp_commandLine=new
ImagePlus(imagePath);
                        ImageProcessor
ip_commandLine=imp_commandLine.getProcessor();
                        int[][] xy_data_commandLine=new int[30][2];

            xy_data_commandLine=LueKoordinaattiTiedosto(landmarkPath,
1,false);
                        ip_commandLine.setColor(java.awt.Color.red);
                        for(int i=0;i<xy_data_commandLine.length;i++)
                        {
                                    if(xy_data_commandLine[i][0]==0)
                                                break;

            ip_commandLine.fillOval(xy_data_commandLine[i][0],xy_data_c
ommandLine[i][1],rad*2,rad*2);
                        }
                        imp_commandLine.setProcessor(ip_commandLine);
                        File imageFile = new File(imagePath);

                        IJ.save(imp_commandLine, newFilePath);
                        return newFilePath;
            }
public static int[][] LueKoordinaattiTiedosto(String path, int
numberOfTitleRows, boolean target)
{
   int[][] xy_data=new int[30][2];
   try
   {
     FileInputStream stream = new FileInputStream(path);
     InputStreamReader reader = new InputStreamReader(stream);
     BufferedReader br= new BufferedReader(reader);
     for(int i=0;i<numberOfTitleRows;i++)
     {
            br.readLine();
     }

     String riviS=null;
     int rivi=0;
     do
     {
            riviS=br.readLine();
            if(riviS!=null)
            {
```

```
                StringTokenizer st=new StringTokenizer(riviS,"\t",false);
                st.nextToken();
                if(target)
                {
                   st.nextToken();
                   st.nextToken();

                   int xkoordinaatti=-1;
                   int ykoordinaatti=-1;
             xkoordinaatti=Integer.parseInt(xkoordinaattitoken.trim());
             ykoordinaatti=Integer.parseInt(ykoordinaattitoken.trim());
             xy_data[rivi][0]=xkoordinaatti;
             xy_data[rivi][1]=ykoordinaatti;
                }
             rivi++;
             }
      while(riviS!=null);
      }
      catch (Exception ex)
     {
             ex.printStackTrace();
             return null;
      }

      return xy_data;

}
public static void drawSourceLandmarks_red_Macro(String imagePath,
String landmarkPath, String newFilePath, String circleRadius)
{
             String[] args = {"DrawSourceLandmarks_red_", imagePath,
landmarkPath, newFilePath, circleRadius};
             doDrawingCommandLine(args);
}


}//end of class
```

APPENDIX 2. The program code for calculating the change estimates of the dental casts

This MATLAB program was used to calculate the change estimates of the dental casts. The program binarizes the grid image and segments the cells chosen by the user. Number of pixels of each segmented cell is calculated, and this number is then divided by 4900, which is the number of pixels in the original, non-deformed grid cell. This division gives the ratio *R*. Then 1 is subtracted from that ratio, and the result of the subtraction is multiplied by 100 %. This gives the change estimate, *C*. The values *R* and *C* are described also in Section 3.7.

A big part of the code was provided by researcher Vladimir Bochko.

```
%Program takes the inputs (points and labels)first and then calculates
the results.
%If two cells are horizontally or vertically adjacent, they should
have
%the same label.
%When calculating the results, the areas of cells with same labels are
%averaged.
%Plots results in a bar plot in a separate figure.

function suvi_code2
clc;
dxdy = 4900;

rgb_img = imread('def_grid.jpg');
rgb_img2 = imread('overlap.jpg');

figure(1)
imshow(rgb_img2);

figure(2)
imshow(rgb_img);
figure(1)

hhh = gca;

point_xys=ones(1,2)
labels=ones(1)
teethareas=ones(1)

ind = 1;

while(1)
```

```
    n = 1;
    point_xy = round(ginput(n));

    point_xys(ind,:) = point_xy;

    reply_label = input('Give label: ');
    %reply_label_mat=cell2mat(reply_label)
    %size(reply_label_mat)
    labels(ind) = reply_label; %(char(reply_label));

    %put a rectangular mark to the clicked point in visible image
    rgb_img2(point_xy(2)-2:point_xy(2)+2,point_xy(1)-2:point_xy(1)+2,
:) = 0;


    figure(1)
    imshow(rgb_img2);
    hhh = gca;
    axes(hhh);

    text(point_xy(1),point_xy(2),num2str(reply_label));
    imshow(rgb_img2);

    reply_continue = input('Do you want continue? Y/N [Y]: ','s');
    if isempty(reply_continue)
        reply_continue='y';
%    else if strcmp(reply_continue, 'y')
%        reply_continue='y';
%    else if strcmp(reply_continue, 'Y')
%        reply_continue='y';
    else
        break;
    end

    ind = ind+1;

end
%labels = labels -48
img_filled_double_old = zeros(size(rgb2gray(rgb_img)));
length_xys=length(point_xys)
for index=1:size(point_xys,1),

LabelsDisp = im2double(rgb2gray(rgb_img));

image_mouse_point = zeros(size(LabelsDisp));
image_mouse_point(point_xys(index,2),point_xys(index,1)) = 1;

[AllArea2 img_labels2] = dealing_with_regions(LabelsDisp);%,...
                         %handles.axes2,handles.axes4,...
                         %holesVal, holesMax);
comp_lbl2 = zeros(size(LabelsDisp));
%_____________________________________________________________
% Similarity between regions:
% intersection (i.e dot product for binary images)
% intersection between region and mask
% if any intersect then it is desirable region
% this is alternative to intersection using SVM boundary
```

```
% shown in comments below  starting with a line 188

for i = 1:length(AllArea2)
    comp_lbl2(img_labels2 == i) = 1;
       if AllArea2(i) > 0
           res_comp(i) = comp_lbl2(:)'*image_mouse_point(:);
           if res_comp(i) > 0
               break;
           end
       end
    comp_lbl2(img_labels2 == i) = 0;
end
[max_val_not_used, max_ind] = max(res_comp);

LabelsDisp(img_labels2~=max_ind)=0;
%_____________________________________________________________

img_filled_double_old = img_filled_double_old + LabelsDisp;

img_filled = im2bw(LabelsDisp);

img_filled_show = im2bw(img_filled_double_old);

figure(3)
imshow(img_filled_show);
hold off

area = blob_parameters(img_filled) % blob parameters measurument
myareas = [area];
teethareas(index) = myareas(end);

end

%labels_num = ones(size(labels));

%for i=1:length(labels)
%    labels_num(i) = str2num(labels(i));
%end
labels
teethareas
labels_and_teethareas = [labels' teethareas']


i2=1;
i3=1;
while(i2<=size(labels,2))
    current_label = labels(i2);
    ind2 = i2;
    sum = 0;
    count_same_group = 0;
    while (labels(ind2) == current_label)
       count_same_group = count_same_group + 1;
        sum = sum + teethareas(ind2);
        if(length(labels)>ind2)
            ind2 = ind2 +1;
        else
            break;
        end
```

```
    end
    sum
    count_same_group
    average = (sum / count_same_group)
    R = (average/dxdy)
    ratios(i3, 1) = i3;
    ratios(i3, 2) = R;
    i2 = i2 + count_same_group;
    i3 = i3+1;
end
C=(ratios-1)*100

figure
hold on
bar(C(:,2), 0.4)
set(gca,'XTickLabel',int2str(C(:,1)))

%reply_write_to_file = input('Write all areas to a file? Y/N [Y]:
','s');
%if isempty(reply_write_to_file)
%    reply_filename = input('Give filename: ','s');
%    dlmwrite(reply_filename, labels_and_teethareas, '\t')
%end

%reply_write_to_file2 = input('Write change estimates to a file? Y/N
[Y]: ','s');
%if isempty(reply_write_to_file2)
%    reply_filename = input('Give filename: ','s');
%    dlmwrite(reply_filename, C, '\t')
%end


function [AllArea img_labels] = dealing_with_regions(LabelsDisp)
%,axes4,...
                                %holesVal, holesMax)

img_filled = im2bw(LabelsDisp);

[img_boundaries, img_labels] = bwboundaries(img_filled);

img_labels(img_filled==0) = 0;
stats = regionprops(img_labels, 'Area');
AllArea = [stats.Area];


function [area, sum_area] = blob_parameters(img_filled)
% Blob parameters measurement
% all time consuming parameters excluded (convex-hull etc.)

[img,num] = bwlabel(img_filled);

s1 = regionprops(img, 'Area');
area = s1.Area;

sum_area = sum([s1.Area]);
```