UNIVERSITY OF GLASGOW

# Statistical modelling of local features of three-dimensional shapes

by

Yinuo Liu

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

College of Science and Engineering

School of Mathematics and Statistics

December 2020

# Declaration of Authorship

I, Yinuo Liu, declare that this thesis titled, 'Statistical modelling of local features of three-dimensional shapes' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

*"It ain't what you don't know that gets you into trouble. It's what you know for sure that just ain't so."*

Mark Twain

# *Abstract*

The rapid development of 3D imaging technology allows data to be collected directly in three-dimensional space. The high accuracy of the images requires further investigations on digitised objects, especially of local features. In the last decade, 3D Local features have played an important role in recognising and modelling real-world 3D objects. This thesis introduces a series of methods for 3D local features, including automatic keypoints detection, 3D model construction with curves, local region detection and statistical analysis of local features. Those methods are not only to build 3D local feature descriptors, but also have a wide range of applications, such as shape comparison in medical facial treatments and evolutionary researches in biology.

Conventional shape analysis, limited by the data-collection technology, project 3D objects into 2D space to analyse, or focus on 3D discrete points which are not close to each other. Those points of anatomical meanings are called landmarks. Researchers used to manually place the landmarks on 2D or 3D images by eyes, but it generates the operator error which is not of interest but has a large influence on shape analysis. This thesis introduces a novel method to automatically estimate the landmarks on 3D models using Bayesian statistics. The Procrustes matching of the landmark sets shows that the variation of Bayesian placements is much smaller than the manual placements. Local shapes like "ridges" and "valleys", which are considered to contain rich geometric information, can be estimated based on landmarks. Existing methods rely heavily on landmarks, but in most cases, the number of landmarks is not enough and adding extra ones are time-and-labour consuming. A flexible and user-customisable method is introduced in this thesis to deal with complex surfaces marked with as few landmarks as possible. A simulation study is conducted, and the result shows that the method is stable and efficient in terms of local feature description.

After the 3D curve is estimated, methods to analyse the local features using the curves are discussed. An algorithm to flexibly dissect the surface along the estimated curve is developed for extracting local pieces or divide the surface into

pieces. The novelty of this method is that it applies directly on 3D shapes and dissects the shape along any 3D curves, such as the lip edge on a human facial model. Besides the novel method for 3D shapes, curvatures, which reflect the bending amount along the curves, are calculated. The curvatures of the same local feature on different individuals are aligned to analyse the average shape difference of groups, such as gender and age. A reconstruction procedure from the curvatures is discussed and the effect of noise on choosing the degree of freedom in smoothing is investigated. Another application of the estimated curves is in benchmarking the performance of different 3D camera systems. A new camera system developed by NCTech®, Edinburgh, is assessed using the evaluation outcome of facial deformity surgeries in Brazil. It is designed to be child-friendly, portable and low-cost. Validation studies are carried out at three stages of the development, and both landmarks and curves are used to evaluate the performance of the new camera system on estimating local features in comparison with mature products from DI4D® and Artec®.

# *Acknowledgements*

I would like to express my sincere gratitude to my supervisor, Professor Adrian Bowman, for his invaluable and considerate supervision. He has been always full of kindness, patience and insight, and provided me with the strongest support during all the time of my research. His enthusiasm about knowledge, life and work will be the guidance of my life. I would like to extend my sincere thanks to my co-supervisor, Dr Liberty Vittert, for her invaluable suggestions and unwavering encouragement. It was a great privilege and honour to work and study under their guidance. I am deeply indebted their valuable time.

I would like to pay my special regards to my manager and also my friend Mrs Shazia Ahmed for her kind guidance and encouragement during those years; to my best colleagues and also my friends Ruth and George in LEADS. Their smiles will shine in my life. Besides, many thanks to my colleagues in the school for their kind and patient support; to every participant for their passion for research.

I am also extremely grateful to my family. Without their encouragement to pursue the research and unwavering support during those years, this work would not have been possible. I have been missing them all the time far away from home.

Finally, I wish to acknowledge the support of Mr Hao Xu who accompanied me during those years and his insightful suggestions in my research.

# Contents

# List of Figures

# List of Tables

# Abbreviations

| | |
|---|---|
| **AAM** | **A**ctive **A**appearance **M**odel |
| **AR** | **A**ugmented **R**eality |
| **CMM** | **C**oordinate **M**easuring **M**achine |
| **CNN** | **C**onvolutional **N**eural **N**etwork |
| **CP** | **C**oordinate **P**ath |
| **DCNN** | **D**eep **C**onvolutional **N**eural **N**etwork |
| **DNN** | **D**eep **N**eural **N**etwork |
| **ECT** | **E**stimation **C**orrection **T**uning |
| **FFPD** | **F**acial **F**eature **P**oint **D**etection |
| **GPA** | **G**eneralised **P**rocrustes **A**nalysis |
| **GSS** | **G**eneralised **S**um of **S**quares |
| **IBAHCM** | **I**nstitute of **B**iodiversity, **A**nimal **H**ealth and **C**omparative **M**edicine |
| **LDA** | **L**inear **D**iscriminant **A**nalysis |
| **LOESS** | **L**ocally **E**stimated **S**catterplot **S**moothing |
| **MOSFET** | **M**etal **O**xide **S**emiconductor **F**ield **E**ffect **T**ransistor |
| **OLS** | **O**rdinary **L**east **S**quares |
| **OPA** | **O**rdinary **P**rocrustes **A**nalysis |
| **OSS** | **O**rdinary **S**um **S**quares |
| **PCA** | **P**rincipal **C**omponent **A**nalysis |
| **PCs** | **P**rincipal **C**omponents |
| **PDM** | **P**oint **D**istribution **M**odel |
| **RMLS** | **R**obust **M**oving **L**east **S**quares |
| **rmsd** | **r**oot **m**ean **s**quare **d**eviation |
| **RSS** | **R**esidual **S**um of **S**quares |

| | |
|---|---|
| **SIFT** | **S**cale **I**nvariant **F**eature **T**ransform |
| **SRVF** | **S**quare **R**oot **V**elocity **F**unction |

# Symbols

| | |
|---|---|
| $\vec{\bullet}$ | Vector Notation |
| $\times$ | power Cross Product |
| $\lvert\bullet\rvert$ | Modulus Operator |
| $\langle\rangle$ | Inner Product |

# Chapter 1

# Introduction

The rapid development of 3D imaging technology and processing methodology greets the modern civilisation on a broad new stage. This evolution brings applications including built-in facial recognition system (e.g., Face ID) in mobile phones, digitally-scanned virtual actors in films, inside-body detector and monitor in medical studies, 3D digital preservation of cultural heritage etc. A more colourful world for human beings has been constructed.

## 1.1 A brief history of 3D imaging technology

Most of the objects in the earth exist in 3D form, but do not directly appear in the digital world, unless it is captured or constructed as 3D models. Analysis of 3D models requires accurate reconstruction of the object in the form of meshes, which is widely used in the industry.

Six decades ago, the invention of the MOSFET (MOS field-effect transistor) [9] by Mohamed M. Atalla and Dawon Kahng at Bell Labs triggered the development of the digital camera image sensor. The first fully digital camera, FUJIX DS-1P was produced by Fujifilm in 1988 [10]. The output, a 2D digital image, is composed of pixels each with its intensity recorded over two-dimensional spatial coordinates [11]. Since then, digital photography has come to life. After 1999 [12], digital

images which were captured by phones with built-in image sensors have been sent and shared by people all around the world in seconds.

However, even after digital photography was invented, it could not fulfil the desire of capturing real-world objects with more than two-dimensional imagery. As mentioned in Wade [13], the first 3D display appeared in the 1830s, achieved by the earliest stereoscopes invented by Sir Shares Wheatstone. But at that time, stereoscopy only showed a perception of 3D depth which was processed inside a human brain when the two eyes were watching two different 2D images. An actual 3D image was first captured in the 1960s when the first 3D scanning technology was invented [14].

Similar to a 2D image, a 3D image contains a 3D point cloud which is the scanner's estimate of the surface [14] of a real-world object. The 3D object reconstructed in an image and displayed on a computer is called a 3D shape or a 3D mesh, while a 3D model or a 3D configuration is a mathematical representation of the 3D shape. Modern 3D photography produces 3D images with texture, but it is not of interest in this thesis.

FIGURE 1.1: Contact CMM [1] (left) and non-contact 3D scanner [2] (right)

3D imaging technology has developed rapidly with different types of scanner, referred to Leo and Manimegalai [15]. A 3D scanner can be categorised by surface-contact and non-surface-contact. As shown in Figure 1.1, reliable contact measurement of a complex 3D surface in manufacturing is created by a coordinate measuring machine (CMM) [16]. It produces extremely precise 3D point clouds, but there is also a risk of changing or even damaging the surface of the measured

object, such as a human face or a fragile cultural relics. Thus, as summarised by Leo and Manimegalai [15], the common 3D reconstruction methods are the non-contact scanners, with three common technologies namely: passive and active 3D stereo camera systems, time-of-flight sensors and structured-light sensors. The principle of those four methods is visualised in Figure 1.2. They can be grouped into passive and active technologies.

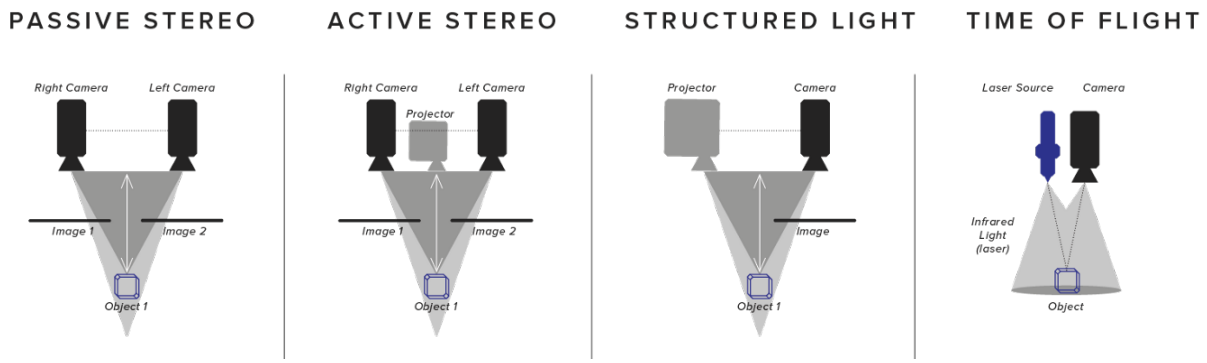



FIGURE 1.2: Examples of different non-contact 3D scanners from DAQRI [3], where "Projector" is the light emitter.

The non-contact passive stereo on the first left is a passive type whose development is inspired by the principle of human stereoscopic vision. Normally there are digital cameras (left and right) fixed slightly apart, each capturing a photo of the object and the 2D images (image 1 and 2) are registered and analysed by software to reconstruct a 3D image. The name "passive" comes from the nature of human eyes seeing an object, which receives the natural visible light reflected by the surface of the object and directed towards the eyes. However, if there is not enough natural light, the active stereo scanner on the second left which has a projector actively emitting light to the object may outperform the passive stereo scanner.

The last three technologies in Figure 1.2 all actively project the light to the object by itself and measures the distance by receivers (sensors, normally digital cameras). As Leo and Manimegalai [15] mentioned, the two most popular active technologies are structured light scanning (the third left) and 3D laser scanning (the fourth left).

Time-of-flight, or laser scanning, uses a beam emitter to project from the laser source to the surface. A detector inside the camera system receives the reflected light and records the corresponding time. With the known speed of light, the measured distance should be twice the distance between the scanner and the object, at the point of the surface which reflects this beam of light. All the distances at the points are measured by repeating this procedure so that the point cloud of a 3D image can be created based on those distances. It performs better in large objects measuring, such as buildings 3D reconstruction. But it also suffers from inaccuracy because of the difficulty to catch the light. Thus, it is more useful when scanning smaller objects.

The latest invention for scanning small objects is to project structured-light. It performs well for both static and dynamic object. Instead of dots and stripes, this type of scanner projects a pattern to the surface. As it captures the shape from the deformation of the projected pattern on the surface by sensors near the emitter, it also can record dynamic high-speed movements.

The data involved in this thesis are 3D shapes, including human faces and fish mandible bones. Facial data are normally scanned with high-accuracy by high-speed 3D scanners [17]. For example, Vittert et al. [7] used non-contact passive stereo camera designed by DI4D®. However, most state-of-art 3D scanners are very expensive. In a project introduced in Chapter 5, we cooperated with NCTech® (based in Edinburgh) to develop a low-cost non-contact active structured-light scanner. We designed several validation studies to benchmark the performance of each stage of the development and compared the outputs with the products from DI4D® and Artec®, with details in Chapter 5. Compared with facial data, fish mandibles are much smaller (1 centimetre each). The data were scanned by Dr Kirsty McWhinnie using non-contact active laser scanners.

Instead of pixels in a 2D image, a 3D photo composed of a set of points, each with its depth perception, is recorded by three-dimensional spatial coordinates.

The photos that we are familiar with are converted from camera vector graphics. However, the data in this research are point clouds connected by Delaunay triangulation, the details of which are given in Section 2.2.

## 1.2 Motivation for curve identification

Nature has provided the common features that can be seen among every group of creatures. The commonalities are defining elements of each species, whereas the subtle differentiation makes the individuals. Curves, which represents the bending size and direction of the surface, enable the first grasp of impression, then the identification.

This thesis examines the curves of everyone's fundamental feature, the face, and establish its entity in the digital realm, in which the curve can be identified and analysed. Besides the human face, the fish mandible contains the secret of evolution. 3D shape analysis helps biologists to reveal the truth by showing the statistical results between species.

The computational toolbox used in this thesis is the R language R Core Team [18]. The algorithm of the methods was developed upon an unpublished R package Adrian and Katina [19] in our lab which is called "face3d". The new methods in Chapter 3 and 4 are conducted by the algorithms written by myself.

### 1.2.1 Motivation for local feature extraction

Morphological modularity is a popular topic in Biology, where shape analysis is applied based on its definition that the integration in a structure is compartmentalized with strong within-module and weak between-module integration, as introduced in Klingenberg [20]. Previous studies, such as the genomics study, aiming to find the gene which conducts adaptive divergence on shapes, analyse key shape features through two-dimensional landmarks. For example, Parsons et al.

[4] captures the shape to investigate morphological modularity by placing 15 landmarks and 56 semi-landmarks which are in-between points of the landmarks on the lateral left-side two-dimensional images shown in Figure 1.3.

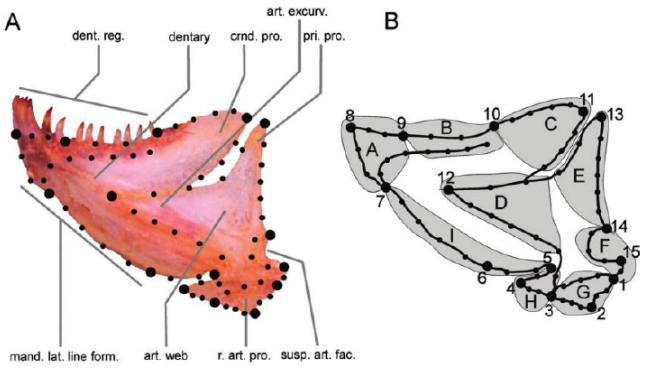



FIGURE 1.3: The left 2D lateral view of a cichlid mandible with landmarks (black larger dots) and semi-landmarks (black smaller dots). The figures are taken from Parsons et al. [4].

In the anatomical area of interest, landmark-only shape analysis with the reduction of one dimension loses much information to identify the key local feature of the shape. Thus, this thesis introduces a novel approach to identify three-dimensional curves on the shape which has a wide application in different disciplines. Here we mainly focus on two: morphometrics analysis of Lake Malawi cichlid mandible and shape analysis on the surgery of treatable facial deformity (cleft lip and palate).

Leng et al. [21] gave a comprehensive summary of methods to describe local features and register images according to the local feature descriptors. In the early stage of this topic, Moravec [22] introduced the Harris corner detector to match images by a set of local points identified by a corner detector. The method was evolved from the Moravec detector to the Harris corner detector, and extended from matching images with small variations to a large range of images.

However, as pointed by Lowe [23], the Harris corner detector is not stable across scale changes in the images. Although this problem has been solved, a fully affine-invariant, including scale, rotation and translation, local feature descriptor is in high demand. For example, Lowe [23] introduced a widely-used method of image

matching and recognition, called Scale Invariant Feature Transform (SIFT), which detect key points to match the images. Others include detecting regions, edges, image contours and so on.

Chapter 3 introduces a novel approach to the estimation of 3D ridge and valley curves as local feature descriptors, as ridge and valley curves contain rich information of the surface which can be used in a variety of studies, such as shape comparison, modelling and recognition. With those local feature descriptors, a local region can be dissected with the contour and even the whole shape feature can be analysed, as discussed in Chapter 4.

### 1.2.2 3D facial curve identification

Faces, the front of the head that hosts the eyes, nose and mouth, are crucial for individuals' identities. The modelling of a single face can be achieved by describing the face as a manifold represented by discrete points. Identifying a curve on a manifold can be a great challenge due to complex facial shapes and sophisticated solid geometry relationships.

Applications of curve identification on facial shapes are widely used by industries, but more specifically, in medical practice. The curve is especially useful to characterise specific diseases or illness, or evaluate the effect of facial surgery.

Curves on the face may not be as obvious as colour, but they are very informative for shape analysis. This thesis explores the quantitative measurement of facial surgery using a statistical method, by comparing pre-surgical and post-surgical changes. Besides, a control group is introduced to compare with post-surgical facial shapes to assess the success of the surgery. Both steps are required to fully assess the success of the surgical operation.

The idea of analysing cleft-lip or cleft-palate patients is investigated, with particular focus on new-born infants in less developed countries. Quantitative measurement based on curves can be used to establish a comprehensive database and enable the analysis of the shapes and lead to better treatment.

## 1.3 Thesis organisation

This thesis mainly introduces local feature extraction methods of 3D shapes. Before explaining the methods, Chapter 1 explains how the data used in this thesis computationally created from real-world 3D objects and why 3D local feature extraction is important for modern shape analysis; Chapter 2 describes all the fundamental concepts used through the thesis including data structure, statistics, differential geometry, shape analysis and algebra.

As the first part of the key findings, Chapter 3 introduces two local feature extraction methods with 3D curves as the descriptor. The first is a further development of an existing method Vittert et al. [24], where the geodesic distance is measured instead of the Euclidean distance approximation. It works well on less complex surfaces, while the second method is a novel approach which can deal with fluctuate shapes, even with holes.

Chapter 4 and 5 includes different aspects of curves applications. The novel methods include the dissection of 3D shapes along the estimated curves and the assessment of imaging performance based on curves in developing a new camera system. Others are applications of the curves on traditional shape analysis, such as principal component analysis to compare shape difference between groups, curve alignment and reconstruction. They are typical applications of the extracted local features.

Chapter 6 introduces a novel approach to automatically localise landmarks on 3D shapes by a Bayesian model. It meets the requirement from the difficulty to manually place landmarks in all the chapters above.

Chapter 7 contains a summary of the thesis and the limitations that the study should consider. It also suggests future work to further investigate the methods in this thesis and their possible applications.

# Chapter 2

# Background concepts

## 2.1 Data collection

In general, the data involved in this thesis are three-dimensional images of real-world objects, with an example shown in Figure 2.1 (left). The image consists of discrete points with three-dimensional (3D) coordinates captured by stereo cameras, as the small dots shown in Figure 2.1 (middle).

In this thesis, a "point" refers to the mathematical representation of a three-dimensional coordinates, and a "dots" represents the visualisation of a "point". The coordinates of the points correspond to an arbitrary coordinate system, denoted by $x$, $y$, $z$, created by the camera system. These can be stored as a matrix, where each row contains the coordinates of each 3D point and each column refers to a dimension. The row number of each point is used as its index. For example, if the data consist of 10 points, the matrix will have 3 columns and 10 rows, and the indices of the points are from 1 to 10.

The imaging system also processes the 3D points to become vertices of triangles based on Delaunay triangulation [25]. As can be seen in Figure 2.1 (right), the surface of the object is displayed by a set of triangles. The number of triangles will increase with a higher resolution of the image. The triangles are stored with the indices of their vertices as groups of three.

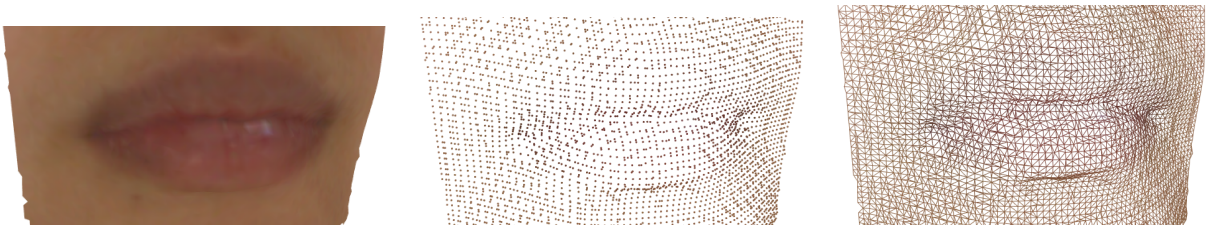


FIGURE 2.1: Left: A 3D image of a human lip which is captured and processed by a 3D Artec® imaging system; Middle: The point cloud in the image; Right: The image is displayed by triangles whose vertices form the point cloud.

Landmarks are also 3D points. Referring to Katina et al. [26], they represent well-defined correspondence across objects. Another data structure involved in this research is a curve on a 3D image. Curves are characterised by a set of successive 3D points on the image. Landmarks and the points on the curve can either be the observations (triangle vertices) captured by the camera system, or the estimates in-between the observations, but they are all considered to be on the surface.

Figure 2.2 shows an example of a mid-line lip curve (black and red dots) on a 3D image. To highlight the curve on the image, the points which lie on the curve are plotted with a much larger size than the point cloud in Figure 2.1 (middle). The curve is identified with the guidance of two anatomical landmarks (black larger dots) by the method in Chapter 3. The landmarks lying on the corners of the lip are manually placed in the ©IDAV system *Landmark*. The definitions of landmarks and curves will be illustrated further in Section 2.2.1.

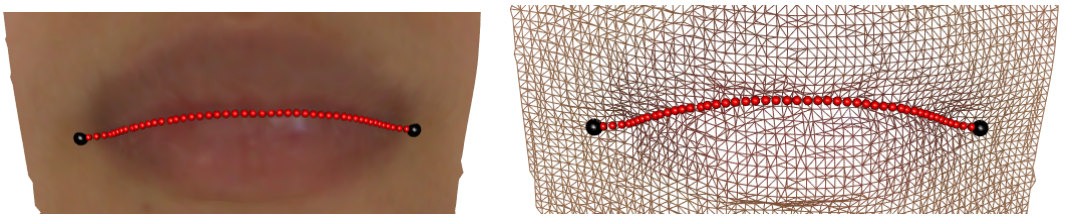


FIGURE 2.2: A mid-line lip curve (red large dots) with landmarks (black larger dots) which is identified on a 3D lip image.

## 2.2 Fundamental materials in use

Figure 2.3 shows an example of a 3D triangulated local surface.

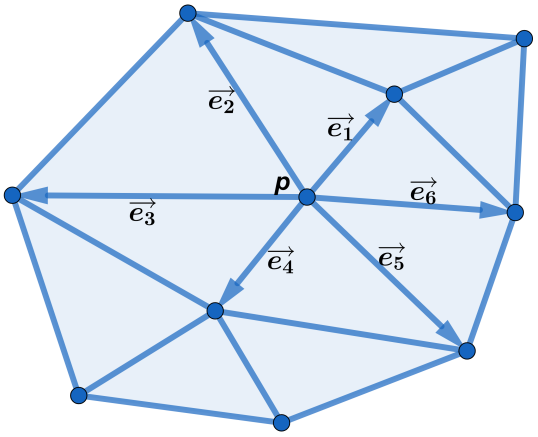



FIGURE 2.3: A 3D triangulated surface. The vertices, including $p$, are the points observed by the 3D camera system. Adjacent points processed in the same triangle are connected by triangle edges.

Shape index and principal curvature are two key concepts in this research.

Principal curvatures measure the maximum and minimum bending of a 3D surface at any point, along the directions of two principal vectors on its tangent plane. To evaluate the principal curvatures of a point (e.g., $p$ in Figure 2.3), its adjacent points are first found by which share the same edges. The shared edges are denoted by $\overrightarrow{e}_i = (e_{i1}, e_{i2}, e_{i3})$, as shown in Figure 2.3.

A normal vector of a point is the vector which is orthogonal to the tangent plane at the point. In this thesis, define the normal vector pointing out of the surface, e.g., pointing to the front on a facial image The normal vector $\overrightarrow{n}_p$ of the surface at point $p$ is obtained by averaging the cross products of paired adjacent $\overrightarrow{e}_i$. In detail, if there are $N$ edges which share point $p$ ($N = 6$ in Figure 2.3), the normal

vector at $p$ to the surface will be

$$\begin{aligned}\overrightarrow{n}_p =& \frac{1}{N}\sum_{i=1}^{N}\overrightarrow{e}_i \times \overrightarrow{e}_{i+1} \\ =& \left(\frac{1}{N}\sum_{i=1}^{N}(e_{i2}e_{i+1,3} - e_{i3}e_{i+1,2}), \frac{1}{N}\sum_{i=1}^{N}(e_{i3}e_{i+1,1} - e_{i1}e_{i+1,3})\right. \\ & \left., \frac{1}{N}\sum_{i=1}^{N}(e_{i1}e_{i+1,2} - e_{i2}e_{i+1,1})\right),\end{aligned} \tag{2.1}$$

where $\overrightarrow{e}_{N+1} = \overrightarrow{e}_1$. The unit normal vector of $p$ is $\overrightarrow{N}_p = \frac{\overrightarrow{n}_p}{|\overrightarrow{n}_p|}$.

A local coordinate system at point $p$ can be constructed by: its unit normal vector $\overrightarrow{N}_p$, one of the edge vectors ($\overrightarrow{e}_i$ in Figure 2.3) denoted by $\overrightarrow{x}_u(p) = (p_{u1}, p_{u2}, p_{u3})$ and the cross product of $\overrightarrow{N}_p$ and $\overrightarrow{x}_u(p)$ denoted by $\overrightarrow{x}_v(p) = (p_{v1}, p_{v2}, p_{v3})$. Used as three axes of the local coordinate system, those three vectors are all standardised by their modulus. For convenience, denote the three local axes by $X_u(p)$, $X_v(p)$ and $N_p$, where $N_p$ is the normal to the surface and $X_u(p)$, $X_v(p)$ denote orthogonal axes in the tangent plane of $p$.

Any point $q_i = (x_{qi}, y_{qi}, z_{qi})$ in the neighbourhood of $p$ can be projected to this local coordinate system. Note that the axes $x$, $y$ and $z$ in this local coordinate system are $X_u(p)$, $X_v(p)$ and $N_p$. The projected coordinates are as follows.

$$\begin{aligned}q_{i,projected} &= (aX_u(p), bX_v(p), cN_p) \\ &= (\langle\overrightarrow{pq_i}, X_u(p)\rangle X_u(p), \langle\overrightarrow{pq_i}, X_v(p)\rangle X_v(p), \langle\overrightarrow{pq_i}, N_p\rangle N_p)\end{aligned} \tag{2.2}$$

A local shape of $p$ which consists of $q_{i,projected}$s can be fitted the model

$$z = f(x,y) = \frac{A}{2}x^2 + Bxy + \frac{C}{2}y^2 + Dx^3 + Ex^2y + Fxy^2 + Gy^3. \tag{2.3}$$

The Weingarten matrix [27] at the point $p = (x_p, y_p, z_p)$ is

$$W = \begin{pmatrix} A & B \\ B & C \end{pmatrix}. \tag{2.4}$$

The principal curvatures of $p$ are the eigenvalues of its Weingarten matrix $W$. The two principal curvatures describe bending along and across a ridge or a valley curve. Referring to Ohtake et al. [28][1], a ridge curve or a valley curve can be defined by principle curvatures. Denote the maximal and minimal principal curvatures of the surface by $\kappa_{max}$ and $\kappa_{min}$ ($\kappa_{max} \geqslant \kappa_{min}$); the corresponding principal directions by $\overrightarrow{d_{max}}$ and $\overrightarrow{d_{min}}$; the first derivatives of the principal curvatures in their corresponding principal directions by $\kappa'_{max} = \frac{\partial \kappa_{max}}{\partial \overrightarrow{d_{max}}}$ and $\kappa'_{min} = \frac{\partial \kappa_{min}}{\partial \overrightarrow{d_{min}}}$. A ridge curve satisfies

$$\kappa'_{min} = 0, \quad \frac{\partial \kappa'_{min}}{\partial \overrightarrow{d_{min}}} > 0, \quad \kappa_{min} < -|\kappa_{max}| \tag{2.5}$$

and a valley curve satisfies

$$\kappa'_{max} = 0, \quad \frac{\partial \kappa'_{max}}{\partial \overrightarrow{d_{max}}} < 0, \quad \kappa_{max} > |\kappa_{min}| \tag{2.6}$$

The bending across a ridge is described by $\kappa_{min}$ which is negative and the bending across a valley is described by $\kappa_{max}$ which is positive. Thus, a more convex ridge shape has smaller $\kappa_{min}$, while a more concave valley shape has larger $\kappa_{max}$. For convenience, a more convex or concave shape is regarded as having a stronger principle curvature.

These two principal curvatures can also be denoted by the first principal curvature $\kappa_1 = \kappa_{max}$ and the second principal curvature $\kappa_2 = \kappa_{min}$, where $\kappa_1 \geqslant \kappa_2$. For example, to highlight the ridges by yellow, the 3D image of a human lip in Figure 2.5 (left) is coloured by $1 - \kappa_{min}$ with its colour indicator on its right. The range of $\kappa_{min}$ is [-0.18,0.09], so the range of $1 - \kappa_{min}$ is [0.91,1.18].

While curvatures give the bending strength, shape index (SI)[5] represents the shape category at any point $p$. In other words, the shape at any point can be

[1] Ohtake et al. [28] uses an opposite definition of ridge and valley curves, because the normal vector is defined pointing into the surface, while here it is pointing out of the surface.

categorised by SI based on the value of principal curvatures.

$$SI = \frac{2}{\pi} arctan\left(\frac{\kappa_{min} + \kappa_{max}}{\kappa_{min} - \kappa_{max}}\right). \tag{2.7}$$

It can be regarded as a mapping of the shape type to the range [-1,1]. Figure 2.4 adapted from Vittert et al. [24] gives some examples of the value of SI. It can be used to recognise local shape with a similar type of the surface, such as the middle horizontal valley of human lips. The difference between principle curvatures and SI is shown in Figure 2.5, where the figure on the right shows the same image but coloured by SI with its colour indicator on its right ($-0.99 < \text{SI} < 0.98$). The vertical middle of the lip shows a convex shape so that it has SI close to 1 with red colour, and the strong curvature $\kappa_{min}$ with brown colour.

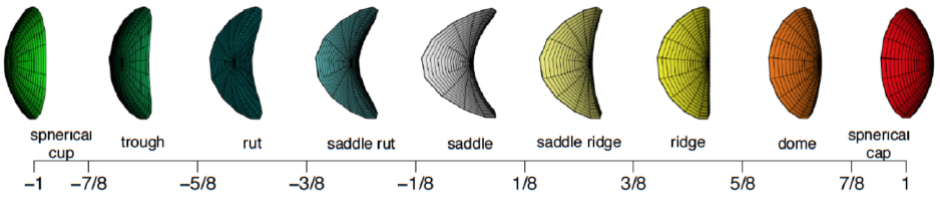



FIGURE 2.4: Local surfaces associated with the shape index on the scale from -1 to 1, with colour coding to identify each shape category [5].

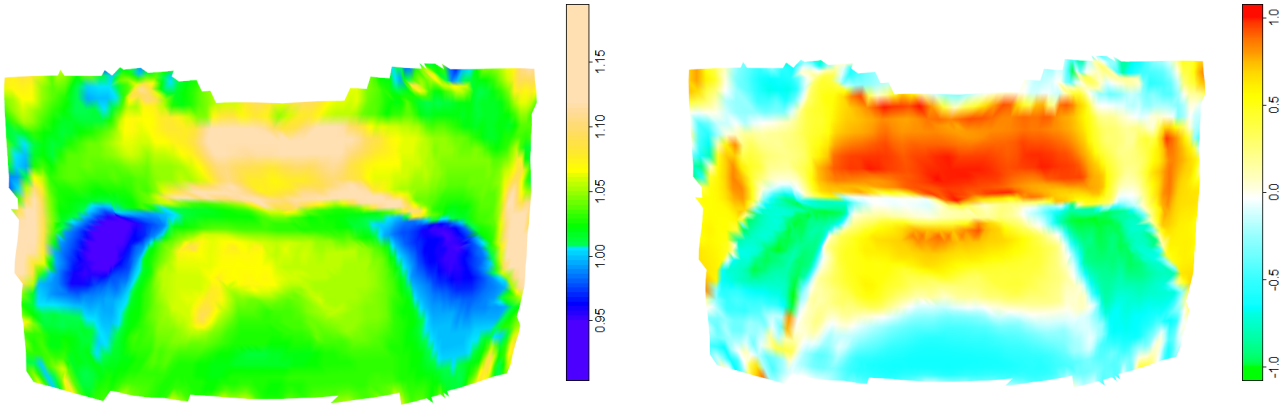



FIGURE 2.5: Left: a 3D image which is coloured by $1 - \kappa_{min}$ of each point; Right: coloured by the shape index of each point. Both have their colour indicator (bar) by their right-hand side.

### 2.2.1 Local features identification

Local features on a face can be represented by landmarks and curves. The captured image can be loaded into the ©IDAV system *Landmark*, which allows landmarks to be manually placed. The surface can be displayed in a 3D space and the landmarks can be placed anywhere on the surface in the system with the normal vector shown to assist the placements. Figure 2.6 shows an example of a facial surface displayed in the system with 23 anatomical landmarks defined by Katina et al. [26].

In addition to landmarks, curves which consist of successive 3D points are another form of data with geometrical information. Arc length is an important measurement of curves. The discrete points on a 3D curve are ordered and denoted by $p_0, p_1, p_2, \ldots$. The arc length of the $n$th point $p_n = (x_n, y_n, z_n)$ on the curve in the space expanded by three dimensions $x, y$ and $z$ is defined by

$$s_n = \sum_{i=1}^{n} \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 + (z_i - z_{i-1})^2}. \tag{2.8}$$

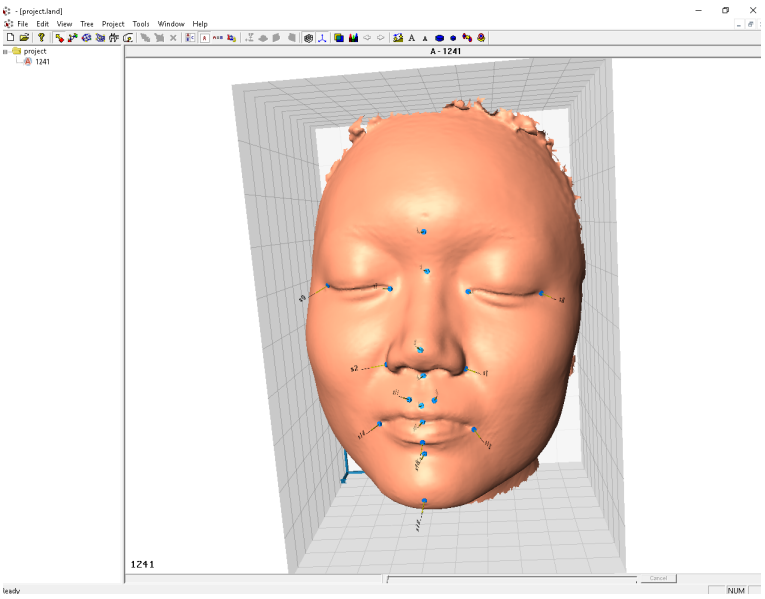



FIGURE 2.6: An example of manually placing the landmarks on a human face.

### 2.2.2 OPA, GPA, PCA and LDA in shape analysis

Dryden and Mardia [29] illustrated methods to analyse two or more configurations. Generalised Procrustes Analysis (GPA) is a popular method to match a set of configurations. It is derived from Ordinary Procrustes analysis (OPA) which matches two configurations by minimising their distances. The principle of OPA is to remove location ($\gamma$), rotation ($\Gamma$) and scale ($\beta$) differences between two shapes. Denote two sample configurations by $X_1$ and $X_2$. OPA seeks to minimise the squared Euclidean norm $\|X_2 - \beta X_1\Gamma - 1_k\gamma^{\mathrm{T}}\|^2$ which is called the Procrustes distance between $X_1$ and $X_2$; while GPA seeks to minimise the total squared Euclidean norm of $n$ samples as follows ($n > 2$).

$$\mathrm{GSS} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=i+1}^{n}\|(\beta_i X_i\Gamma_i + 1_k{\gamma_i}^{\mathrm{T}}) - (\beta_j X_j\Gamma_j + 1_k{\gamma_j}^{\mathrm{T}})\|^2 \tag{2.9}$$

where GSS denotes the Generalised Sum of Squares in GPA. However, in some studies, subjects of different ages are sampled and the size difference is also of interest. So Ordinary partial Procrustes and Generalized partial Procrustes are applied instead of full OPA and GPA. Without removing scale ($\beta$) differences, Ordinary partial Procrustes minimises the squared Euclidean norm $\|X_2 - X_1\Gamma - 1_k\gamma^{\mathrm{T}}\|^2$, while partial GPA minimises the total squared Euclidean norm of n configurations as follows ($n > 2$).

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{j=i+1}^{n}\|(X_i\Gamma_i + 1_k{\gamma_i}^{\mathrm{T}}) - (X_j\Gamma_j + 1_k{\gamma_j}^{\mathrm{T}})\|^2 \tag{2.10}$$

The population mean shape, or average configuration, after full GPA can be estimated by the sample average configuration $\bar{X} = \frac{1}{n}\sum_{i=1}^{n}(\beta_i X_i\Gamma_i + 1_k{\gamma_i}^{\mathrm{T}})$, while the partial GPA version is $\bar{X} = \frac{1}{n}\sum_{i=1}^{n}(X_i\Gamma_i + 1_k{\gamma_i}^{\mathrm{T}})$.

Dryden and Mardia [30] described a principal component model for shape which is developed by Cootes et al. [31]. A configuration $X$ whose coordinates are stored

in a matrix can be vectorised as

$$x = \text{vec}(X), \tag{2.11}$$

where $X$ is a matrix with $m$ columns and $n$ rows, and $x$ is a vector with length $mn$ constructed by stacking the columns of $X$. Following the notation in Dryden and Mardia [29], the inverse operator of $\text{vec}(\cdot)$ is ${\text{vec}_m}^{-1}(\cdot)$

$$X = {\text{vec}_m}^{-1}(x). \tag{2.12}$$

In a principal component model, the shape difference is decomposed by principal components which are obtained from the sample covariance matrix of a set of vectorised configurations. However, for configurations of more than two dimensions, the distribution is too complex to calculate the variation of the shapes. Thus, the shape variations are approximated using tangent coordinates in the tangent space, instead of the original coordinates of a configuration $X_i$. In Procrustes tangent space, the pole is first chosen, for example, to be the average configuration. The configuration $X_i$ is transformed by translating, rotating and rescaling by Procrustes analysis so that the transformed ${X_i}^P$ is as close as possible to the pole. The Procrustes tangent coordinates are the Procrustes residuals

$$r_i = \text{vec}({X_i}^P) - \text{vec}(\bar{X}) = {x_i}^P - \bar{x}. \tag{2.13}$$

The sample covariance matrix of Procrustes residuals is

$$s_r = \frac{1}{n}\sum_{i=1}^{n}(r_i - \bar{r})(r_i - \bar{r})^{\mathrm{T}}. \tag{2.14}$$

where $\bar{r} = \frac{1}{n}\sum_{i=1}^{n} r_i$. The principal components (PCs) are calculated by the eigenvectors of $s_r$ and denoted by $v_i$. The corresponding eigenvalues $\lambda_i$ is the shape variation along the $i$th PC. The principal component score of the $i$th subject on

the $j$th PC is calculated by

$$s_{ij} = v_j^{\mathrm{T}}(r_i - \bar{r}). \tag{2.15}$$

As the $i$th eigenvalue $\lambda_i$ of $s_r$ represents the variance of Procrustes residuals along the $i$th PC, the typical shape with two standard deviations from the pole along the $i$th PC is characterised by

$$X_t = \mathrm{vec}_m{}^{-1}(x_t) \tag{2.16}$$

$$= \mathrm{vec}_m{}^{-1}\left(\bar{x} + \bar{r} + 2\lambda_i{}^{\frac{1}{2}} v_i\right). \tag{2.17}$$

If the Procrustes residuals $r$ come from different groups, they can be used to build classification rules to predict the group of a new set of residuals. This thesis will only consider the two-class case of Linear Discriminant Analysis (LDA).

Following the notations in Lindo [32], LDA assumes that each group $C_g$ is normally distributed with a group-specific mean $\mu_g$ and a common covariance matrix $\Sigma$. Denote the probability that randomly chosen object belongs to the $g$th class by $\{\pi_g\}$=P$(C_g)$. Based on the Bayes' Theorem, the posterior probability used to indicate the "confidence" in a given classification is

$$\mathrm{P}(C_g|r) = \frac{\mathrm{P}(r|C_g)\mathrm{P}(C_g)}{\sum_{i=1}^{2} \mathrm{P}(r|C_i)\mathrm{P}(C_i)}, \qquad g = 1, 2 \tag{2.18}$$

The discriminant rule is that an object should be allocated to the class which has the largest posterior probability. Suppose the length of the Procrutes residual of an individual is $p$, the likelihood will be

$$\mathrm{P}(r|C_g) = \frac{1}{(2\pi)^{\frac{p}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2} d_g(r)\right\}, \qquad g = 1, 2 \tag{2.19}$$

where $d_g(r) = (r - \mu_g)^{\mathrm{T}}\Sigma^{-1}(r - \mu_g)$. Then the posterior probabilities

$$\mathrm{P}(C_g|r) = \frac{\pi_g \exp\left\{-\frac{1}{2}d_g(r)\right\}}{\sum_{i=1}^{2} \pi_i \exp\left\{-\frac{1}{2}d_i(r)\right\}}, \qquad g = 1, 2 \tag{2.20}$$

where $\pi_i = \frac{1}{2}$. $\mu_g$ is estimated by the average Procrustes residuals of each group and $\Sigma$ is estimated by the pooled covariance matrix

$$S = \frac{1}{n-2}\sum_{g=1}^{2}(n_g - 1)S_g, \tag{2.21}$$

where $n$ is the total sample size, $n_g$ is the sample size of each group and $S_g$ is the sample variance of each group. Then the object with observation $r$ would be classified into class 1 if and only if $\mathrm{P}(C_1|r) > \mathrm{P}(C_2|r)$.

### 2.2.3 Nonparametric smoothing methods

The statistical models, such as curve estimation, of local features, are non-parametric, so smoothing is essential. Chapter 3 and 4 discuss the estimation and application of curve estimates, where piecewise smoothing methods are involved, including local linear regression, natural (cubic) smoothing and penalty spline smoothing. Those smoothing methods use the combination of piecewise polynomials to fit discrete observations and evaluate using the fitted models.

#### Local linear regression

The ordinary least squares (OLS) regression uses the residual sum of squares (RSS) to measure the overall model fit. Suppose we have a set of observations $\{x_i, y_i\}$, $i = 1, \ldots, n$. The coefficients of the regression are chosen by minimising the squared difference between each $y_i$ and its fitted on the regression line.

$$\mathrm{RSS} = \sum_{i=1}^{n}(y_i - x_i{}^{\intercal}\beta)^2 = (y - X\beta)^{\intercal}(y - X\beta) \tag{2.22}$$

So the coefficient estimate which minimises Equation 2.22 can be derived as $\hat{\beta} = (X^\intercal X)^{-1} X^\intercal y$. Define a hat (smoothing) matrix by $S = X(X^\intercal X)^{-1} X^\intercal$ and the degree of freedom is defined by the trace of the hat matrix $df = \text{trace}(S)$ which equals to the sample size minus the number of predictors or columns of $X$. However, the data in this thesis vary heavily and cannot be fitted well by a straight line. As introduced by DeVeaux et al. [33], the local linear regression takes the local average of the response variable and estimates the coefficients at $x$ by minimising

$$\text{RSS}_l = \sum_{i=1}^{n} w(x_i - x; h)(y_i - x_i - x^\intercal \beta)^2 = W(y - X\beta)^\intercal (y - X\beta) \tag{2.23}$$

where $w(\cdot)$ is a weight function which gives more weights to the observations close to $x$ and $h$ determines the range of the local average. Different from the OLS regression, the locally estimated scatterplot smoothing (LOESS) uses the local average of the observations to fit in the model and presents a smoother regression curve. The smoothing matrix becomes $S_l = W^\intercal X(X^\intercal W X)^{-1} X^\intercal W$ and the degree of freedom is still defined by its trace. In Chapter 4, the smoothness of the fit will be adjusted by changing the degree of freedom. The smoothness will increase when $\text{df}_l$ decreases and the optimal $\text{df}_l$ for smoothness will be found by experiments.

### Natural (cubic) spline smoothing with penalty

Another popular nonparametric smoothing method is to fit piecewise cubic polynomials and add a penalty to RSS which controls the smoothness of the fit. The penalised residual sum of squares is

$$\text{RSS}_c = \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \int f''(x)\, dx = (y - \hat{m})^\intercal (y - \hat{m}) + \lambda \hat{m}^\intercal A \tag{2.24}$$

In other words, following the notation in Hyndman [34], the piecewise cubic polynomials $h_1(x) = 1, h_2(x) = x, h_3(x) = x^2, h_4(x) = x^3, h_j(x) = x - \kappa_{j-4}^3$ for $j = 5, \ldots, k+4$ are the predictors and determined by a set of $k$ knots $\kappa_1 < \kappa_2 < \cdots < \kappa_k$. Suppose $B_{ij} = h_j(x_i)$ is the basis matrix, then the RSS

becomes

$$\mathrm{RSS}_c = (y - B\beta)^\intercal (y - B\beta) + \lambda \beta^\intercal \Omega \beta \tag{2.25}$$

where $\Omega_{jk} = \int h_i''(x) h_j''(x)$. The smoothing matrix is $S_c = B(B^\intercal B + \lambda\Omega)^{-1} B^\intercal$. In order to be comparable with other methods, the smoothness is still adjusted by the degree of freedom, which is the trace of $S_c$.

**Basis spline smoothing with penalty**

Equation 3.5 in Section 3.2.2 uses a P-spline smoothing technique which is B-spline smoothing with a penalty. A basis spline, or B-spline, was introduced by de Boor [35] as a function which is a piecewise polynomial with order $q + 1$ or degree $q$ of a variable or variables (for a high-dimensional basis spline function). For a one-dimensional basis spline function, the pieces are defined on the intervals whose start and end values are called knots.

The left of Figure 2.7 (a) shows a basis spline consisting of two polynomials of order 2 whose non-zero parts are defined on intervals $[t_1, t_2]$ and $[t_2, t_3]$, where the $t_i$s are values of the variable $T$. The left of Figure 2.7 (b) shows a basis spline consisting of three polynomials of order 3 whose non-zero parts are defined on intervals $[t_1, t_2]$, $[t_2, t_3]$ and $[t_3, t_4]$. On the right of Figure 2.7 shows more basis splines of order 2 in (a) and order 3 in (b).

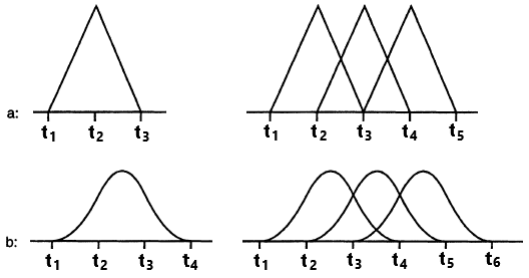



FIGURE 2.7: Visualisation of basis splines of degree 1 (a) and degree 2 (b) from Eilers et al. [6]

The use of basis splines is to fit a smoothing curve to a set of discrete observations whose coordinates are denoted by $(x_i, y_i)$, where the $x_i$s are observations of $T$ .

Following the notation in Eilers et al. [6], the fitted smooth function is a linear combination of $n$ basis splines.

$$\hat{y}(x) = \sum_{j=1}^{n} \hat{\alpha}_j B_j(x; q) \tag{2.26}$$

where $B_j(x; q)$ denotes the $j$th basis spline function of variable $x$ with degree $q$; $\hat{\alpha}_j$ is called a parameter in this fitting. The fitting is determined by the number of knots (or $n$) and the degree $q$.

However, the variation of the fitted curve will increase with a larger number of knots. O'Sullivan [36] and Eilers et al. [6] introduced some solutions that added in another constraint to control the variation. The latter is an approximation of the former, but has the advantage that it is more applicable. Thus, we use B-spline smoothing with a penalty on (higher-order) finite differences of the coefficients of adjacent B-splines, developed by Eilers et al. [6].

The RSS to minimise in regression is the deviance between the observation and the fitted value

$$\text{RSS}_b = \sum_{i=1}^{m} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{m} \left( y_i - \sum_{j=1}^{n} \hat{\alpha}_j B_j(x; q) \right)^2 \tag{2.27}$$

Adding the $k$th difference penalty, the RSS becomes

$$\text{RSS}_{bp} = \sum_{i=1}^{m} \left( y_i - \sum_{j=1}^{n} \hat{\alpha}_j B_j(x; q) \right)^2 + \lambda \sum_{j=k+1}^{n} (\delta^k \alpha_j)^2 \tag{2.28}$$

where $\lambda$ is a parameter and $\delta$ is a difference operator ($\delta^{(1)}\alpha_j = \alpha_j - \alpha_{j-1}$ and $\delta^{(k)}\alpha_j = \delta(\delta^{(k-1)}\alpha_j)$).

Based on one-dimensional basis splines, Eilers et al. [37] extended their work to create two-dimentional basis splines by the tensor products of $n_1 + n_2$ one-dimensional

basis splines. The fitted function becomes a surface as follows.

$$\hat{y}(x) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \hat{\alpha}_{ij} B_i(x_1; q) B_j(x_2; q) \tag{2.29}$$

Two-dimensional penalty spline smoothing will be described and implemented in Section 3.2.2.

# Chapter 3

# Identifying 3D curves with landmarks

In local feature extraction, 3D curves can provide much richer geometry information than 3D landmarks. With the rapid development of high-resolution 3D camera systems, 3D curves can be estimated with much higher accuracy. This chapter discusses two approaches to estimate curves on 3D shapes. The first method is a further development of Vittert et al. [7] where the geodesic distance is approximated by the Euclidean distance. The second is a brand-new approach to estimate curves on relatively complex surfaces without the guidance of large number of landmarks. A simulation study is included to test the performance of this new method under the pressure of random noise.

## 3.1 A brief history of 3D curve estimation

In the last few decades, 3D object recognition has become a popular research topic. As Guo et al. [38] mentioned, there are three procedures of 3D object recognition: detecting 3D keypoint (or landmarks), describing local features of the object and comparing objects. Guo et al. [38] provided a comprehensive summary of efficient

methods before 2015. However, the local feature of an object can be better described by curves which are prominent with rich geometric information and can be used to recognise or compare objects. Thus, many researchers including biologists have started to analyse shapes with curves instead of conventional landmark-based methods.

Curve estimation can be grouped into two categories: estimating from meshes and point clouds, as summarised by Daniels et al. [39]. Much research is based solely on 3D point clouds, instead of on a 3D triangulated mesh. For example, Daniels et al. [39] used and improved Robust Moving Least Squares (RMLS) to extract points on the surface where it bends sharply. Curves are identified by connecting subsets of those points and can also be used to build triangulations for the surface. This method works well on objects whose surface bends more than 90 degrees within a small range, such as creases, but might be challenging where the surface has a mild bump to estimate.

Others were interested in estimating curves directly on 3D triangulated meshes. Lin et al. [40] summarised a set of surface fitting-based methods. For instance, Ohtake et al. [28] developed a method to detect ridges and valleys which have efficient power to describe shapes. The detected ridges and valleys lack connectivity, but their principle can be a good guide to future researches. In fact, our method described in Section 3.4 is inspired by Ohtake et al. [28]. Hildebrandt et al. [41] studied the triangles individually and examined on the principal curvatures of its vertices to select candidate edges and smooth the result. This method does well in extracting feature lines on a 3D object. However, it can be problematic when it is used to estimate a contour of a closed area, as the ridges (or valleys) are estimated together and it lacks connectivity with nearby curves.

There are also many effective methods in computer science with the help of machine learning. Zhang et al. [42] extracted lines as the contours of large objects, based on Deep Neural Network (DNN). Similar to other methods based on Convolutional Neural Network (CNN), this requires a huge training data set. Davies et al. [43] finds curve parameterisation by optimising an objective function which

is used to build a measurement of shape qualities. They parameterised each point on a 2D curve by its proportional distance to the left and right neighbours. The parameterisation is extended to 3D by the proportional distance to the vertices of the triangle where the point on the curve lies. Because the curve is identified by the optimal objective function, this method is time-consuming and also infeasible when the data set is large.

This chapter aims to introduce some methods which are user-friendly on automatic 3D curve estimation. The methods work with 3D triangulated meshes and do not require training data sets. They are all implemented in R, but the algorithm can be programmed in other software environments, such as Matlab. The estimation can be adjusted easily by changing parameters, to satisfy the user's need. There are two contributions in this chapter. Section 3.2.2 describes a further development of an existing method which is introduced in Section 3.2.1. This method is designed to deal with normal surfaces without gaps or holes, while the second method introduced in Section 3.4 is a brand-new approach which focuses on complex shapes where there is not enough information or landmarks to guide the curve estimation.

## 3.2 Statistics principles

3D curves were introduced in Section 2.2, consisting of successive discrete points. Because the image (observed manifold) from a 3D camera system is an estimate of the true manifold of the photographed object, the identified curve on the image is also an estimate of a curve on the observed manifold. Vittert et al. [24] discussed a parameterised curve over a true manifold, defined by the functions $x$, $y$ and $z$ of the arc length $s$, which can be denoted by $p(s) = ((x(s), y(s), z(s))$. They also emphasised that if the image error reduced (e.g. after repeated sampling), the identified curves would converge to the true curve.

## 3.3 Plane cuts and subsequent flexible adjustments

This section will first introduce an existing method to estimate curves Vittert et al. [7]. It estimates curves by adjusting the intersect of a 2D plane and the surface. Within this adjustment, there is one step which approximates the distance along the surface by the Euclidean distance. This method works very well on relatively "smooth" surfaces, such as human faces. However, it may fail on complex surfaces with gaps or holes where this approximation has large deviance. The contribution of this thesis in Section 3.2.2 is to calculate the actual geodesic distance without large computational pressure.

### 3.3.1 Identify a "plane cut" on the shape

To identify anatomical curves on human faces, a methodology has been developed by Vittert et al. [7] which starts by creating a "plane cut" between two landmarks on a 3D surface. As there is no indication where the curve lies, the initial step is to identify the location where a plane in 3D intersects with the edges of the triangulation. As shown in the red dashed curve in Figure 3.1, this "plane cut" lying on the surface consists of 3D red crosses, starts at landmark $l_1$ and ends at landmark $l_2$. So the aim is to identify the coordinates of the red crosses.

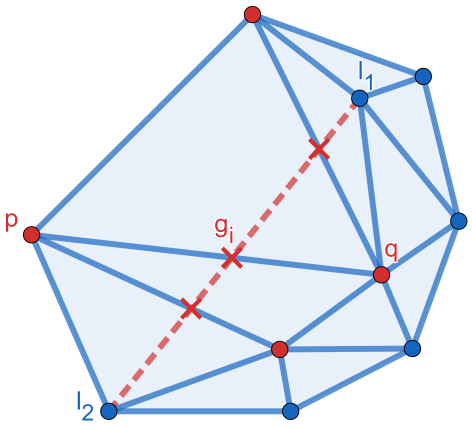



FIGURE 3.1: Interpolate a "plane cut" on the surface

Firstly, if the cross product of the normal vector at $l_1$ and vector $\overrightarrow{l_1 l_2}$ is denoted by $\overrightarrow{c}$, then $\overrightarrow{c}$ and $\overrightarrow{l_1 l_2}$ will define a plane going through $l_1$ and $l_2$, whose normal vector is $\overrightarrow{c}$, .

Secondly, the points on the surface which lie on either side of the plane can be distinguished by comparing the lengths of their projections on $\overrightarrow{c}$, the normal vector of the plane. In detail, suppose that the coordinates of $l_1$ are $(l_{11}, l_{12}, l_{13})$. The length of the projection of $\overrightarrow{l_1}$ on $\overrightarrow{c}$ can be calculated by the inner product of $\overrightarrow{l_1}$ and $\overrightarrow{c} = (c_1, c_2, c_3)$, which is $l_{1,projected} = \langle \overrightarrow{l_1}, \overrightarrow{c} \rangle = \sum_{i=1}^{3} c_i l_{1i}$. For any point $p = (p_1, p_2, p_3)$, the length of its projection on $\overrightarrow{c}$ can be evaluated in this way, so that $p_{projected} = \sum_{i=1}^{3} c_i p_i$. Thus, the points on the surface will be divided into two groups of larger lengths than $l_{1,projected}$ or smaller, corresponding to left or right of the plane.

Thus, a dummy variable $r$ which has only two values: 0 (left) and 1 (right), can be used to characterise the position of a point with respect to the plane. Because the image data consists of the indices of the triangles and the coordinates of their vertices, the following procedure starts from the triangles instead of the edges. For a triangle (indexed by the camera system) with three vertices all on one side of the plane, it will be either $\sum_{i=1}^{3} r_i = 0$ or $\sum_{i=1}^{3} r_i = 3$. So the triangles whose intersections with the plane are not empty will be selected by $0 < \sum_{i=1}^{3} r_i < 3$. For each selected triangle, the edges whose two vertices (red dots in Figure 3.1) have different values of $r$ will then be marked. As an edge is shared by two triangles, these marked edges can be sorted one-by-one in order.

Lastly, the coordinates of the red crosses ($g_i = (g_{i1}, g_{i2}, g_{i3})$ in Figure 3.1) can be obtained based on the coordinates of the vertices ($p = (p_1, p_2, p_3)$ and $q = (q_1, q_2, q_3)$ in Figure 3.1) which are on the same edge and $\overrightarrow{c} = (c_1, c_2, c_3)$ as following.

$$\overrightarrow{g_i} = \overrightarrow{p} + \lambda \overrightarrow{pq}, \tag{3.1}$$

where $\lambda$ is the proportion of the length of $\overrightarrow{pg_i}$ over the length of $\overrightarrow{pq}$. Because $\overrightarrow{c}$ is orthogonal to the plane which $g_i$ and $l_1$ lie on, $\lambda$ can be obtained by the projections of $p$, $q$ and $g_i$ on the vector $\overrightarrow{c}$.

$$\lambda = \frac{\langle \overrightarrow{c}, \overrightarrow{l_1} \rangle - \langle \overrightarrow{c}, \overrightarrow{p} \rangle}{\langle \overrightarrow{c}, \overrightarrow{q} \rangle - \langle \overrightarrow{c}, \overrightarrow{p} \rangle} \tag{3.2}$$

$$= \frac{\sum_{i=1}^{3} c_i l_{1i} - \sum_{i=1}^{3} c_i p_i}{\sum_{i=1}^{3} c_i q_i - \sum_{i=1}^{3} c_i p_i}. \tag{3.3}$$

Therefore, a series of ordered discrete points $(l_1, g_1, g_2, \ldots, g_N, l_2)$ can be obtained, by which a curve is constructed.

### 3.3.2 A flexible adjustment of the "plane cut"

A series of "plane cuts" can be obtained on the plane containing $\overrightarrow{l_1 l_2}$, by rotating $\overrightarrow{c}$ with different angles $\theta$ around the unit vector of $\overrightarrow{l_1 l_2}$ $\left(\text{denoted by } \overrightarrow{L} = \frac{\overrightarrow{l_1 l_2}}{|\overrightarrow{l_1 l_2}|}\right)$. The rotation matrix $R$ is from Rodrigues' rotation formula [44].

$$R = \overrightarrow{c}\cos\theta + (\overrightarrow{L} \times \overrightarrow{c})\sin\theta + \overrightarrow{L}\langle \overrightarrow{L}, \overrightarrow{c} \rangle (1 - \cos\theta). \tag{3.4}$$

The black plane path on a human lip shown in Figure 3.2 illustrates this. The optimal path was selected by maximising the integral of curvatures along the path. However, we can see that the initial "plane cuts" do not quite capture the valley curve of the lip. So a more flexible estimate can be investigated by an adjustment procedure, where initial "plane cuts" will be used as an initial curve estimate, called the *reference* path.

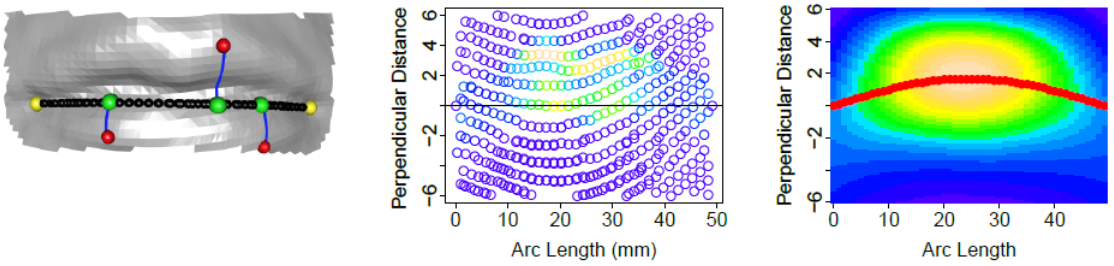



FIGURE 3.2: An example of the adjustment procedure [7]. Left: yellow are two landmarks; the black curve is the *reference* path; red points are ones on the related local area with negative SI; green points are red points' the closest points on the black curve; Middle: all points on the plot are ones on the related local area, such as the red points on the left figure, and are coloured according to $\kappa_1$. Right: the points on the middle plot are fitted. The path with a maximum $\kappa_1$ on this related local area is shown in red.

This procedure first projects three-dimensional data into two dimensions. At first, an appropriate subset of the shape can be selected with appropriate values of shape index (for a valley, all points with negative SI). For any 3D (red) point on the shape, its Euclidean distance from the *reference* path is computed (vertical axis in Figure 3.2). This distance is measured to the closest point (green) on the *reference* path and denoted by $d_p$. The other dimension (horizontal axis in Figure 3.2) contains the arc length of the closest (green) point along the *reference* path and is denoted by $s_p$. The middle plot in Figure 3.2 is the result of this projection. Each point on this plot refers to a 3D point on the lip shape.

Besides, suppose that the maximum (negative for a ridge and positive for a valley) principal curvature is $\kappa_1$. All the points on the plot were coloured according to their $\kappa_1$ values the same with Figure 2.5 (left), which is used for visualisation of the size of the bending degree for each point. The colour will be closer to yellow when its curvature is larger and closer to blue when its curvature is smaller.

As shown in Figure 3.2 (middle and right), the horizontal middle lip curve has maximal curvature in the lip valley. According to this characteristic, for each arc

length $s_p$ on the middle plot in Figure 3.2, we seek to identify the perpendicular distances at which $\kappa_1$ is maximised. Then a curve consisting of a set of discrete 2D points would be found and smoothed by spline smoothing with a penalty on the second-order difference as in the following procedure.

For convenience, the investigated principal curvature $\kappa_1$ is simplified by $\kappa$. As shown in the middle plot in Figure 3.2, each point is coloured by its curvature $\kappa$. In order to estimate the location at any arc length $s$, $\kappa$ is fitted by two independent variables, $s_p$ and $d_p$. Basis function constructed as the product of two one-dimensional cubic splines were used, with a penalty the second-order difference on the fitting coefficients $\beta_{ij}$. In other words, $\kappa$ was fitted by the tensor product of two set of basic cubic spline functions, $\phi(s)$ and $\phi(d)$. Then the objective function to minimise for the fitting of curvatures $\kappa$ is

$$S = \sum_p \sum_i \sum_j \{\kappa(s_p, d_p) - \beta_{ij}\phi_i(s_p)\phi_j(d_p)\}^2 + \lambda \left( \sum_i (\Delta^2 \beta_{ij})^2 + \sum_j (\Delta^2 \beta_{ij})^2 \right) \tag{3.5}$$

where $\lambda$ is the penalty parameter and $\Delta^2$ denotes the second order difference. For an adequate flexibility for the fitted surface, the coefficient estimates $\hat{\beta}_{ij}$ were obtained by fixing the penalty parameter $\lambda$ to be equivalent to 12 degrees of freedoms ($df$ in Equation 3.6).

$$df(\lambda) = trace(\Phi(\Phi^{\mathrm{T}}\Phi + \lambda D)^{-1}\Phi^{\mathrm{T}}). \tag{3.6}$$

In this way of constructing the curvatures $\kappa$ and the estimation of coefficients $\hat{\beta}_{ij}$, the red curve on the third plot in Figure 3.2 will be obtained by an iteration of Newton's method. The aim is to find a series of points with maximum total curvature by maximising the following function, where the integration of curvatures along the arc length is calculated by a discrete approximation.

$$M = \frac{1}{N} \sum_{p=1}^{N} \kappa(s_p, \alpha_p) - \lambda \alpha^{\mathrm{T}} P \alpha, \tag{3.7}$$

where $N$ is the total number of pairs of $s_i$ and $\alpha_p$ on a particular curve $\alpha$. Here $\alpha$ is a function of arc length $s$, and $P = D^{\mathrm{T}}D$ where $D$ is the second order difference matrix [45]

$$D = \begin{bmatrix} 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & -2 & 1 \end{bmatrix}.$$

The penalty parameter $\lambda$ in Equation 3.7 is user-defined. For example, 0.5 is an experimentally efficient value for the balance between deviance and smoothness on human facial curves. The starting point of the iteration is to set $\alpha = 0$. The iteration equation is

$$\alpha^{(m+1)} = \alpha^{(m)} - H^{-1}(\alpha^{(m)})f(\alpha^{(m)}), \tag{3.8}$$

where $f$ represents the first derivatives of $\alpha_p$ and $H$ denotes the second derivatives matrix. The convergence consisting of a set of discrete 2D points will finally be obtained where Equation 3.7 is maximised.

$$\frac{\partial M}{\partial \alpha_p} = \sum_i \sum_j \hat{\beta}_{ij}\phi_i(s_p)\phi'_j(\alpha_p) - 2\lambda(P\alpha)_p \tag{3.9}$$

$$\frac{\partial^2 M}{\partial \alpha_p^2} = \sum_i \sum_j \hat{\beta}_{ij}\phi_i(s_p)\phi''_j(\alpha_p) - 2\lambda(P)_{p,p} \tag{3.10}$$

$$\frac{\partial^2 M}{\partial \alpha_p \partial \alpha_q} = -2\lambda(P)_{p,q}. \tag{3.11}$$

To transfer back to 3D, for each point $p_c = (s_c, d_c)$ on the final curve, the three surrounding points in the Delaunay triangulation and three Barycentric coordinates $\delta_i$, i=1,2,3, were calculated, so that $p_c = (s_c, d_c) = (\sum_{i=1}^{3} \delta_i s_i, \sum_{i=1}^{3} \delta_i d_i)$. Then the transferred 3D coordinates would be $(x_c, y_c, z_c) = (\sum_{i=1}^{3} \delta_i x_i, \sum_{i=1}^{3} \delta_i y_i, \sum_{i=1}^{3} \delta_i z_i)$, where $(x_i, y_i, z_i)$ are the coordinates of the surrounding vertices in the corresponding 3D triangulation. Figure 4.17 shows an example of identified human facial curves using this procedure.

In some cases, the *reference* path consists of several "plane cuts" and the following

smoothing procedure will be the same. This situation results from when there is a sudden change of direction, and there exist semi-landmarks used to provide specific positions on the curves between two anatomical landmarks.

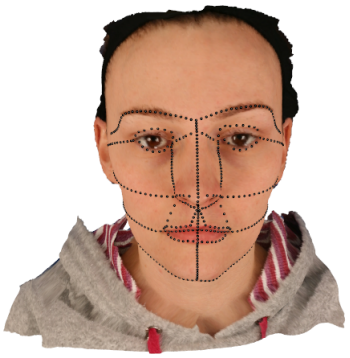


FIGURE 3.3: An example of identified anatomical curves on a human face [8]

### 3.3.3 An example of estimating curves on fish mandibles

Research on the shape of fish mandibles was carried out in cooperation with Dr Kirsty McWhinnie in the Institute of Biodiversity, Animal Health and Comparative Medicine (IBAHCM). As traditional biological methods are based solely on landmarks, this project investigated a novel approach which identified anatomical curves on three-dimensional shapes.

The data consist of 177 fish jaws, as shown in Figure 3.4, which are the descendants of two kinds of fish. The units of the data have not been given, but the size of the jaws is about 1 or 2 centimetres. There are in total 34 landmarks on each fish jaw, including anatomical ones and semi-landmarks. Because of the complexity of the rugged shape, only five principal anatomical curves can be identified satisfactorily using the method by Vittert et al. [7]. Finally, five curves and five pairs of single landmarks (red in Figure 3.4) were used as a representation of the shape of the fish mandible for further analysis.

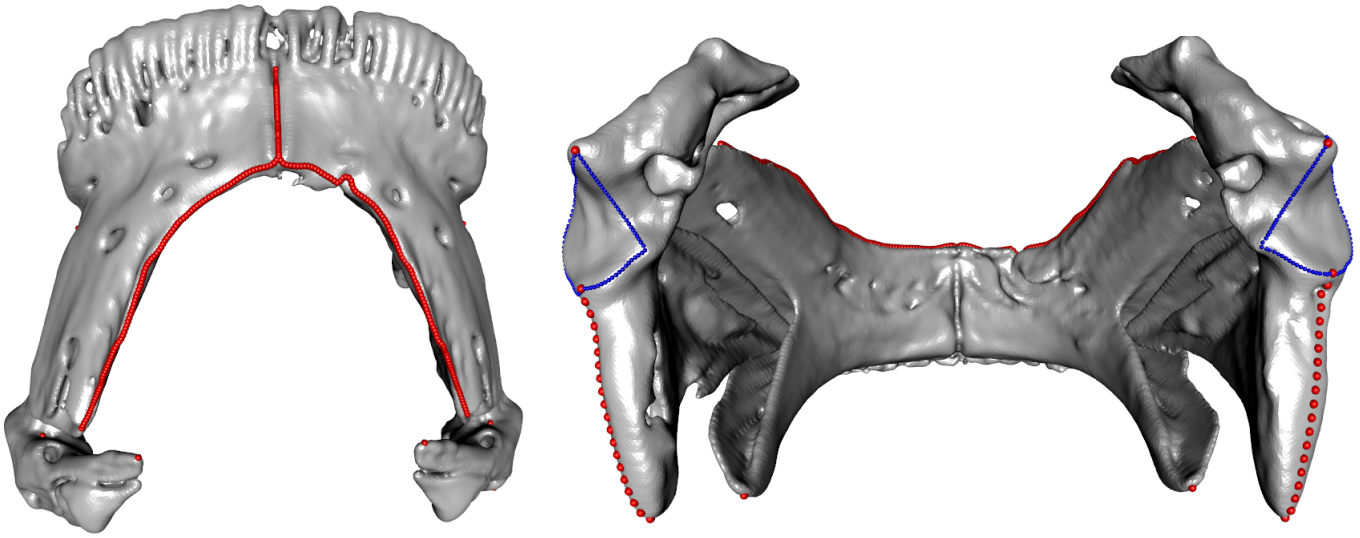


FIGURE 3.4: An example of a fish mandible with identified curves

## 3.4 Problems with the existing method

The smoothing procedure introduced in Section 3.2.2 has been very effective in estimating curves on three-dimensional objects. However, there is a potential danger when the distance of any point to the *reference* path is not well approximated by Euclidean distance. There may not be any marked deviation for objects such as lips on human faces, as the surface of the shape is flat enough for the approximation to be good.

However, shapes may have a folded surface where there is a curve to estimate nearby. Figure 3.5 shows an example of a shape with its top view on the left and bottom view on the right. The black dots are the identified curve which is estimated by the smoothing procedure in Section 3.2.2. After interpolating back to three-dimensional space, some points on the created 3D curve will jump into the hole, or jump far away from other curve points. The missing points of the curve in the left picture are located under the surface indicated by red circles. This results from the large hole near the curves of interest, which may provide the same Euclidean distance from points outside and inside the hole to their closest points on the *reference* path (not shown in Figure 3.5).

In other words, using Euclidean distance to approximate the perpendicular distance in Figure 3.2 sometimes causes the estimation to fail. Thus, an improvement on the smoothing procedure is required to estimate the curves, using the actual distances across the surface instead of the Euclidean distance.

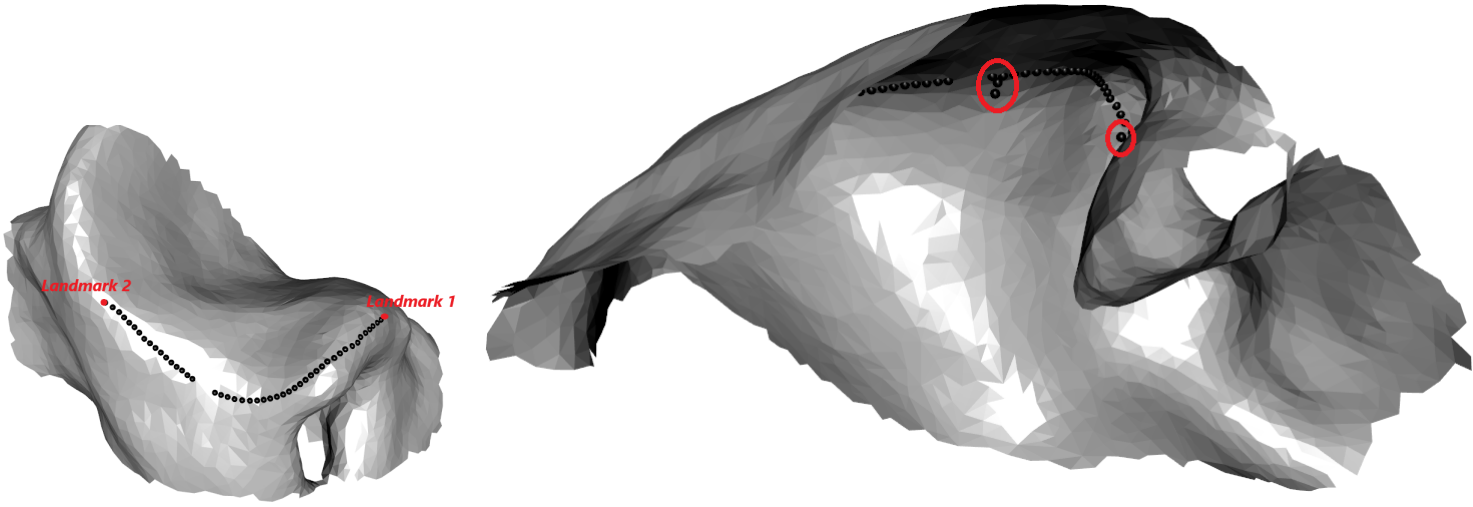



FIGURE 3.5: An example of a failure to identify a curve on a fish bone. The left is the top view and the right is the upward view

### 3.4.1 Distance along the surface

In this section, fish mandibles will still be used as an application to illustrate the improvement of the adjustment procedure. The aim is to estimate a quarter edge of the socket area (blue in Figure 3.4) which is an important part of the mandible to investigate shape differences.

Given the *reference* path (black dots in Figure 3.6) created between landmark 1 and 2, the adjustment in Section 3.2.2 is first to evaluate the distance along the surface for each point nearby to the *reference* path. It would be ideal to produce a "plane cut" from each point to the nearest point on the *reference* path and calculate the total arc length on this "plane cut" as its surface distance, but this is too computationally expensive. So a novel interpolation method has been used to approximate it. The interpolation is based on a fixed small number of "plane cuts" to approximate the distances at all locations. The algorithm is as follows.

Step 1: Create a *reference* path (black dots in Figure 3.6) by the method in Section 3.2.1;

Step 2: Create a series of "plane cuts" containing each point on the *reference* path which follows the perpendicular direction of the point to the path. All "plane cuts" are shown as a series of red and green lines in Figure 3.6 (right). The colour difference represents different sides to the *reference* path. We name this series of created path by "coordinate paths";

Step 3: Calculate the surface distance from any point on the shape (red and green dots in Figure 3.6 left) to the *reference* path by interpolation;

Step 4: Use those surface distances as the perpendicular distances in Section 3.2.2 (or the vertical values in Figure 3.2 right). Then the following procedures to estimate the curve are the same as in Section 3.2.2.

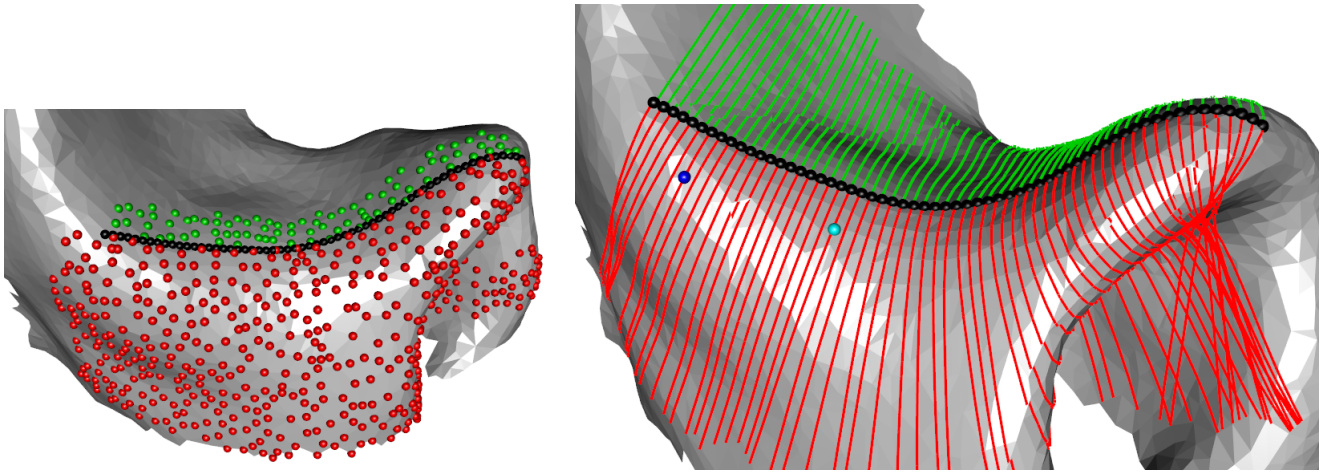


FIGURE 3.6: Improved smoothing procedure. Left: red points have positive distances to the *reference* path (black dotted curve) and green points have negative distances; Right: series of coordinate paths created to calculate surface distance of any point to the *reference* path (black dotted curve) on the shape.

**Details in Step 3: Calculate surface distances by interpolation**

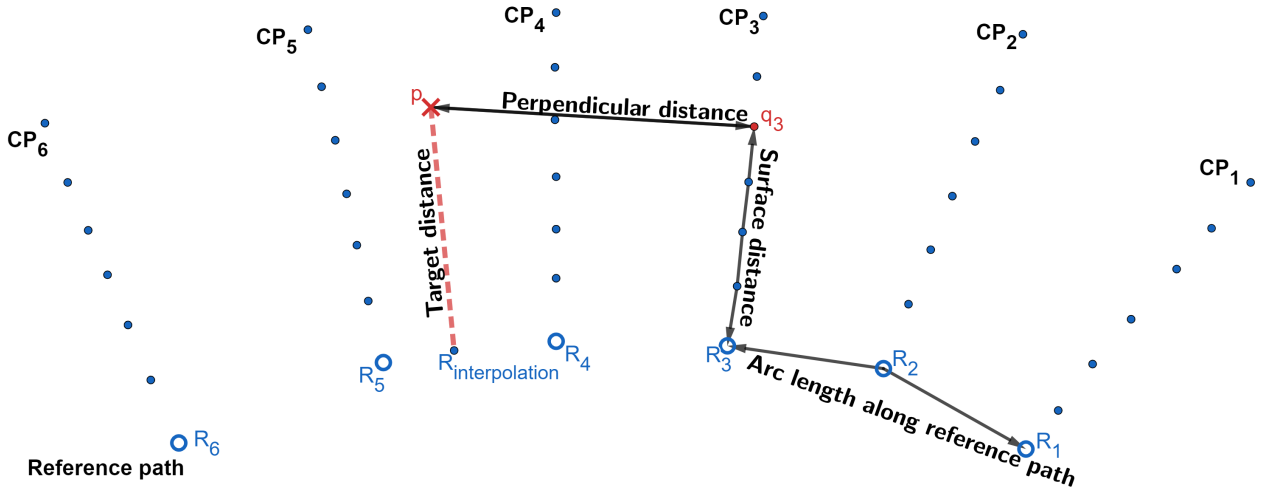
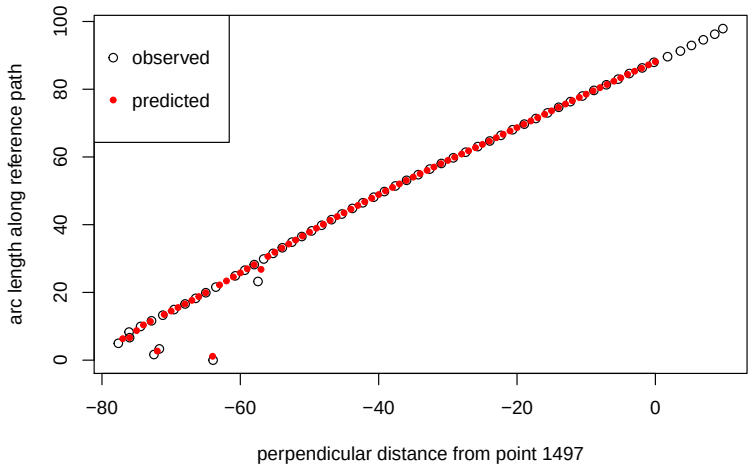



FIGURE 3.7: Illustration of Step 3

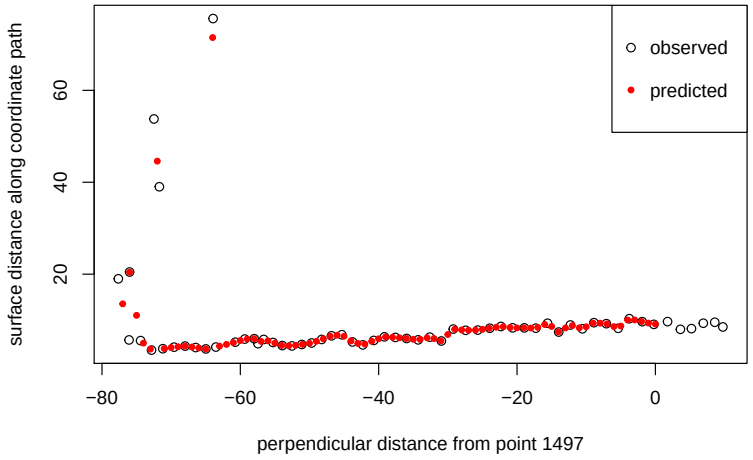
Figure 3.7 shows an illustration of Step 3 in the algorithm. Blue circles $R_1, R_2, \ldots, R_6$ are estimated points on the *reference* path (black dots in Figure 3.6). Blue solid dots $CP_1, CP_2, \ldots, CP_6$ are estimated points on the coordinate paths (red or green lines in Figure 3.6 right) created from $R_1, R_2, \ldots, R_6$. Red cross $p$ stands for any point in the neighbourhood of the *reference* path (red or green points in Figure 3.6 left).

The aim is to calculate the distance from $p$ to the *reference* path across the surface (red dotted lines in Figure 3.7). It is equivalent to interpolating a coordinate path or a point $R_{interpolation}$ on the *reference* path and obtaining the arc length from $p$ to $R_{interpolation}$. However, interpolating a coordinate path for each point nearby is too computationally expensive. So the following procedures can reduce the computing pressure.

For any point $p$ on the shape, its closest point on each coordinate path can be found. In Figure 3.6, for the red cross $p$, the closest point on the 3rd coordinate path is red solid dot $q_3$, with respect to the Euclidean distance. The closest point $q_3$ should be on the same side to the *reference* path with point $p$ (red side or green side in Figure 3.6).




(a)




(b)

FIGURE 3.8: Interpolation to calculate surface distances of any point on the shape. (a) the distance of any point to each coordinate path, against the arc length of the point along the *reference* path which created the corresponding coordinate path; (b) the distance of any point to each coordinate path, against the surface distance along the coordinate path. Red points are the estimated from interpolation

Then the corresponding Euclidean distances from point $p$ to its closest $q_i$s are

recorded and used as the "perpendicular distance" in Figure 3.8. Note that the definition of perpendicular distance in Figure 3.2 is different from the "perpendicular distance" in Figure 3.8. In Figure 3.8, each black circle stands for each $p$.

The arc length from $q_i$ to the *reference* path along its coordinate path $CP_i$ can be calculated and used as "surface distance" on the vertical axis in Figure 3.8 (b).

The arc length of the point $R_i$ on the *reference* path is regarded as the arc length of the coordinate path $CP_i$, correspondingly, the arc length of each $q_i$ along its $CP_i$. Those arc lengths are used as "arc length" in Figure 3.8 (a). Note that $R_1$ has arc length equal to zero.

Each circle in Figure 3.8 is plotted with the above data of each closest $q_i$. As the left and right side $q_i$ and $R_i$ to $p$ use positive and negative values, the aim is to find the corresponding "surface distance" and "arc length" when "perpendicular distance" is zero. This "surface distance" is used as "Perpendicular distance" on vertical axis and "arc length" is used as "Arc length" on horizontal axis in Figure 3.2. In other words, it is to find the "interpolated" coordinate path passing exactly through the point $p$ on the shape, so that its arc length along the "interpolated" coordinate path can be used as its surface distance to the *reference* path.

### Possible inaccuracy of interpolation

This idea of interpolation comes from a grid on a planar shape, which is a simplified version where all points in Figure 3.8 (b) should lie on a perfectly straight and equally-spaced line. However, most of the shapes have a rugged surface, such as the fish mandible. The points in the hole create some crossing coordinate paths, like the red paths in the hole shown in Figure 3.6 (right). The surface distance of a point on the shape to the *reference* path can lead to inaccurate estimation of those paths.

For example, the crossing would make values on the vertical axis in Figure 3.9 non-monotonic. In other words, coordinate paths which are further away can be regarded as the closer, which places the circles in Figure 3.9 at the wrong heights.

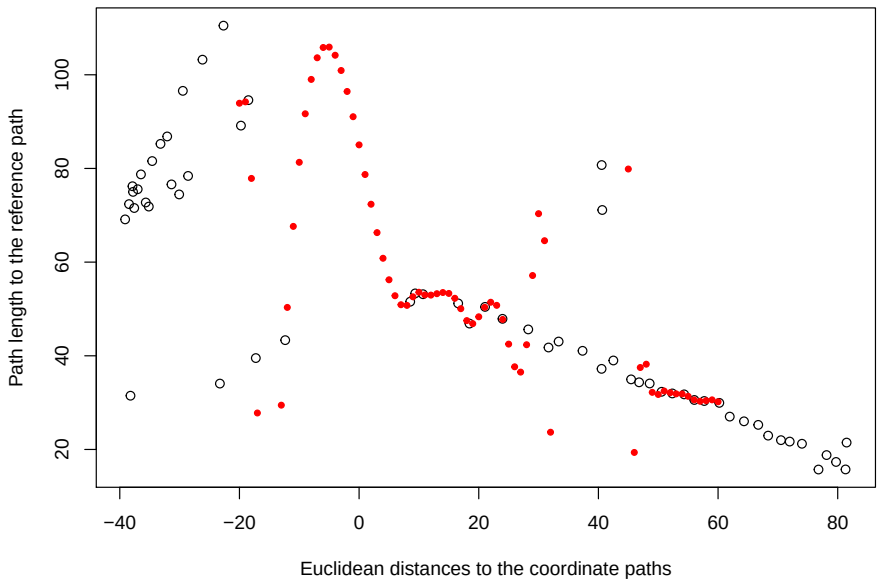



FIGURE 3.9: Wrong estimate at zero perpendicular distance (zero on the vertical axis)

Another issue is shown in Figure 3.10 as an example, where the first coordinate path goes too deep inside the hole and the closest point $q_i$ on the path to the red point $p$ is the blue one. Because the blue point has a long inside-hole surface distance and it is close to the target red point $p$, there would be a signal that the red point is far away from the *reference* path. This can be the reason for the points whose distances are far over 60 in Figure 3.9, while a reasonable range of distances is up to 60.

As shown in Figure 3.9, as a result of the large value, the estimated surface distance (coordinates of red solid points on the vertical axis) at zero perpendicular distance (on the horizontal axis) is much larger (>100) than the expected value (<60).

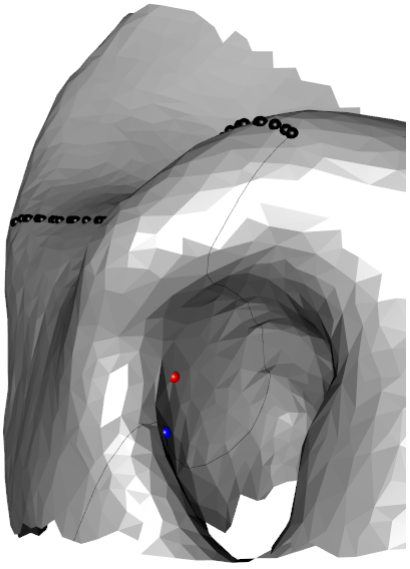


FIGURE 3.10: Too large estimated surface distance of a point (red) on the shape to *reference* path (black dots), which misled by the coordinate path (black line) inside the hole. Blue point is the closest point on the coordinate path to the red point.

Some constraints have been applied to deal with this issue. In Figure 3.9 as an example, points under 60 (vertical) would give a reasonable estimate at 0 (horizontal). So points whose vertical values are far from their neighbour's would be deleted by calculating the differences of adjacent points. Besides, points in the hole would be deleted using the direction of their normal vectors (pointing out of the surface).

### 3.4.2 Problem of penalty parameter after iteration

After recreating the axes in Figure 3.2, the iteration result in Equation 3.8 is very sensitive to the penalty parameter $\lambda$ in Equation 3.7. As shown in Figure 3.11, the red curve is created with $\lambda = 0.2$ and the black is with $\lambda = 0.1$. Although it seems $\lambda = 0.1$ is reasonable for the edge of this jaw, other jaws would have different optimal $\lambda$. This issue can be overcome by repeating the estimation with a fixed conservatively large value of $\lambda$. Each time uses the last estimated curve as a new *reference* path so that it slowly approaches the target position.

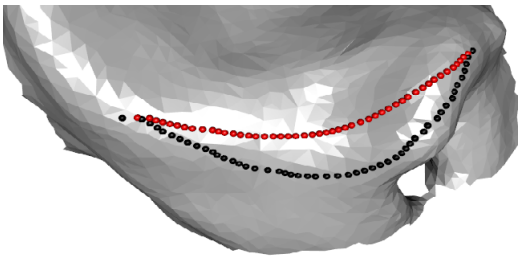


FIGURE 3.11: Two resulting curves from the improved smoothing procedure: a red curve with $\lambda = 0.2$ and black curve with $\lambda = 0.1$

For example, repeating with the same $\lambda = 0.5$ to create a series of *reference* paths has been tried, as shown in Figure 3.12. Shapes in this figure are coloured by the second principal curvature whose maximums (brown in the figure) indicate ridges, as defined in Equation 2.5.

It can be seen that there is a convergence of the repeats on the left shape, but it failed on the right. The reason may be that the curvature signal is strong enough for the convergence on the left shape, but too weak on the right to obtain a convergence on such flatter surface with a large yellow area. Besides, it can be noticed that there is a "ring" area with a strong ridge (yellow) signal. As the repeats are looking for a ridge for the edge, it can be heavily influenced by this and keeps tracking the "ring" which goes far away from the expected edge.

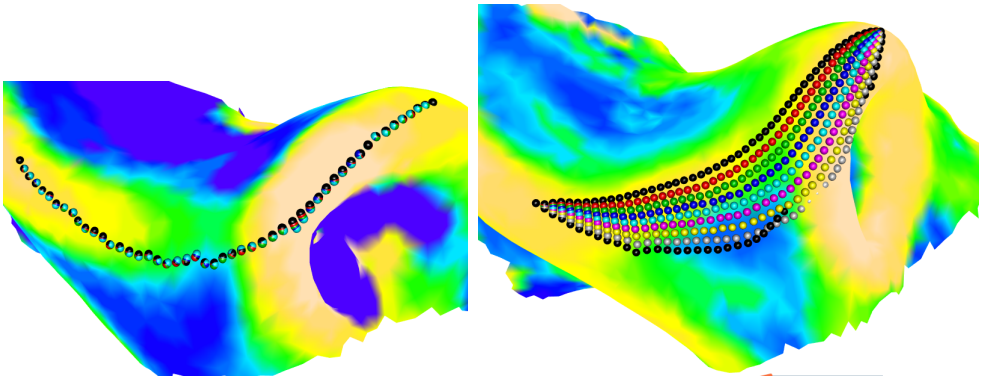


FIGURE 3.12: Shape coloured by the second principal curvature. Left: convergence achieved by updating the *reference* path with $\lambda = 0.5$; Right: fail to obtain a convergence due to strong ridge information in the neighbourhood

## 3.5 Identify rings with only one landmark

Landmarks play an important role in identifying curves on three-dimensional objects. They may guide the composition of a curve (with several landmarks), or be used only to define the start and end of the curve. However, in most cases, the landmarks used to guide the curves are very few. Adding extra ones can be labour-and-time consuming that sometimes it is difficult to accomplish.

The method illustrated in previous sections works well on relatively "simple" surfaces, but it has a limitation that it relies on the guidance of landmarks. Thus, this section describes a novel computational-geometry method which estimates a smooth curve provided only one landmark. It first identifies an initial curve and smooths the curve by adjusting each point on the curve based on the positions of its neighbours. An example of a ring-like ridge identification is used to visualise the algorithm. The shape is shown in Figure 3.13 and is coloured according to the principal curvature $\kappa_2$, with yellow for larger bending of a ridge. It uses only one landmark (black point in Figure 3.13) which is manually placed on (or near) the ring (the brown circle area in Figure 3.13).

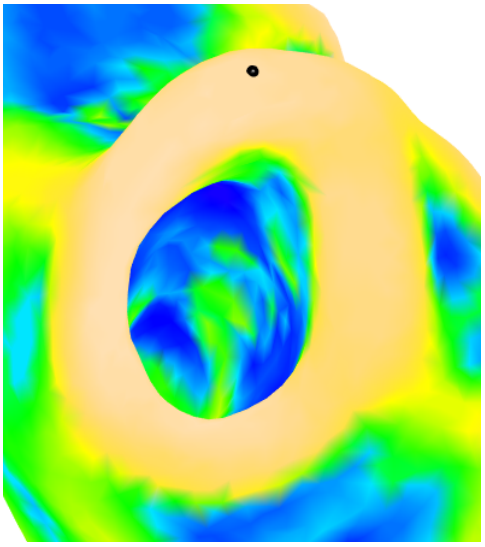


FIGURE 3.13: An example of a ring-like ridge surface

### 3.5.1 Methodology

#### Identification of an initial curve

For any point on the three-dimensional surface, there are two principal directions which determine the directions of the maximum and minimum bending (curvature) of the surface at that point. As introduced in Section 2.2, in terms of a ridge, the first principal curvature ($\kappa_1 = \kappa_{max}$) is in the tangent direction along the ridge and the second ($\kappa_2 = \kappa_{min}$) is in the orthogonal direction.

Surface creases are defined by Ohtake et al. [28]: " The loci of points where the largest in absolute value principal curvature takes a positive maximum (negative minimum) along its corresponding curvature line". Inspired by this, this section proposes a novel but instinctive method to find the loci of the points. One of the principal curvatures, whose principal direction is across the curve direction, is used to locate the points where it takes local positive maximum (or negative minimum for a ridge). The other is used to identify the points one-by-one with equal distance.

The rationale to identify the ridge curve in Figure 3.13 is that it should follow the direction of the ridge and consist of points which have the strongest bending size in the direction across the curve. As the second principal curvature ($\kappa_2$) is negative for a ridge, the *bending depth* is represented by its absolute value ($\kappa = |\kappa_2|$) in their neighbourhood. Those points will be an initial approximation of the curve and then smoothed based on their local neighbours. In the beginning, an initial landmark (previously identified) is used as a starting point.

#### Algorithm

The algorithm and the details are as follow.

Step 1: From a given landmark as a starting point, create a one-step length of a path following the first principal direction of the surface at the landmark;

Step 2: Find the closest vertex on the surface to the end of the path and create a two-sided path following the second principal direction of the vertex;

Step 3: Find the point which has the maximum curvature $\kappa = |\kappa_2|$ on the two-sided path as a new identification on the curve; use it as the next starting point;

Step 4: Repeat Step 1, 2 and 3, until some conditions. For example, reaching a position which is close enough to the original landmark, for a ring-like curve, or reaching a user-defined total arc length of the curve.

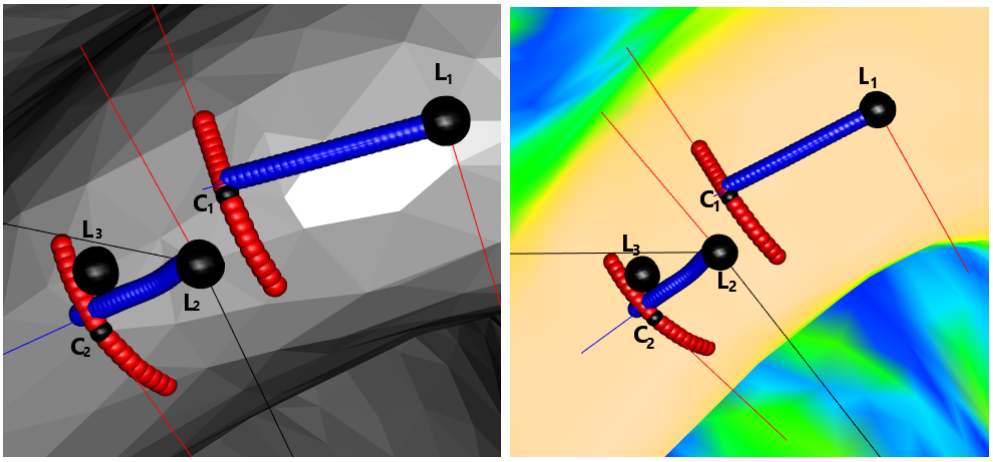



FIGURE 3.14: The procedure to identify an initial estimation of the curve on a ring-like object; Left: the surface is displayed with grey triangles; Right: the triangulated surface is coloured according to the *bending depth* $\kappa$, as a larger size is showed more brown and smaller size is showed more purple.

The algorithm is visualised in Figure 3.14. Denote the given landmark by $L_1$. A candidate point $C_1$ can be selected by moving a certain distance along $L_1$'s first principal direction. This movement can be achieved by creating a "plane cut" (consisting of blue dots) of arc length 10 units following the blue straight line pointing out of $L_1$. This procedure to generate the blue points is called a "movement".

Because the points to estimate along the ridge direction on the curve are required to have the maximum bending depth $\kappa$ in the direction across the ridge, another two-sided "plane cut" is created following the second principal direction of $C_1$ (consisting of red dots). This direction is shown by the red straight line pointing out of $C_1$. This usage of the red dots is called a "comparison". The one with the

maximum $\kappa$ on this "plane cut" will be found and its nearest vertex $L_2$ on the surface is the new identification. $L_2$ will be used as the starting point of the next iteration for $L_3$.

**Further details of the algorithm**

Because the algorithm aims to find a curve consisting of raw data points, the new identification may not be one of the red dots. As introduced in Section 2.1, only the vertices of the triangles are the observations captured by the camera system. Likely, a point on the red "plane cut" is not an observation but an estimate. The algorithm has already run by estimating the curvatures of the observed points. If it is replaced by estimating the curvatures of the estimated points, the deviation of the whole curve estimation will increase. To reduce the total deviation, instead of using the one with the maximum $\kappa$ on the red "plane cut", its closest raw data point is selected. There still might be deviations between the optimal red point and the selected vertex, but the image is dense enough to tolerate this deviation rather than the one between the actual observed and the estimated points. It can be seen on the left of Figure 3.14, that the points $(L_i, i = 1, 2, 3)$ identified on the curve are all the vertices of the original triangles.

Since this procedure is used on this tiny local surface, likely, there are not enough points within a user-defined distance on the "plane cut". So the "plane cuts" may be re-sampled to reach a higher density. Each dimension of coordinates is fitted by natural cubic spline with arc length. For a re-sampled high-density curve, it can be assumed that all the re-estimated equally-spaced points are on the surface. Besides, $\kappa$s of the re-sampled points are approximated by linear interpolations on these high-density "plane cuts" for the comparison procedure.

Furthermore, the identified initial curve may wiggle strongly if there is no direction-control. So in each iteration, a pair of planes are created as "boundaries", by rotating the direction around the normal vector at the starting point. Those "boundaries" aim to control the identification direction and are created by the vector between the last and current identification. For example, in the second iteration in Figure 3.14, after generating the "plane cut" (red dots) from $C_2$, two

vectors (black straight lines pointing out of $L_2$) are created by rotating $\overrightarrow{L_1L_2}$ around the normal vector of $L_2$. As the curve is identified anticlockwise along the ring, the two angles of the rotations are $\frac{\pi}{2}$ anticlockwise and $\frac{\pi}{4}$ clockwise in this example. The "boundaries" are then the two planes expanded by the two rotated vectors and the normal vector of $L_2$. These "boundaries" will ensure the next identification is on the ring direction by deleting points on red "plane cut" which are too far away from the direction of $\overrightarrow{L_1L_2}$.

Figure 3.15 (left) is an example of the identified curve consisting of discrete points in the anticlockwise order. The distance of the "movement" at each iteration is 10 units and the distance of the comparison at each step is 5 units on either side of the candidate $(C_i, i = 1, 2)$ point. This can be called an "initial" curve for the following smoothing procedure.

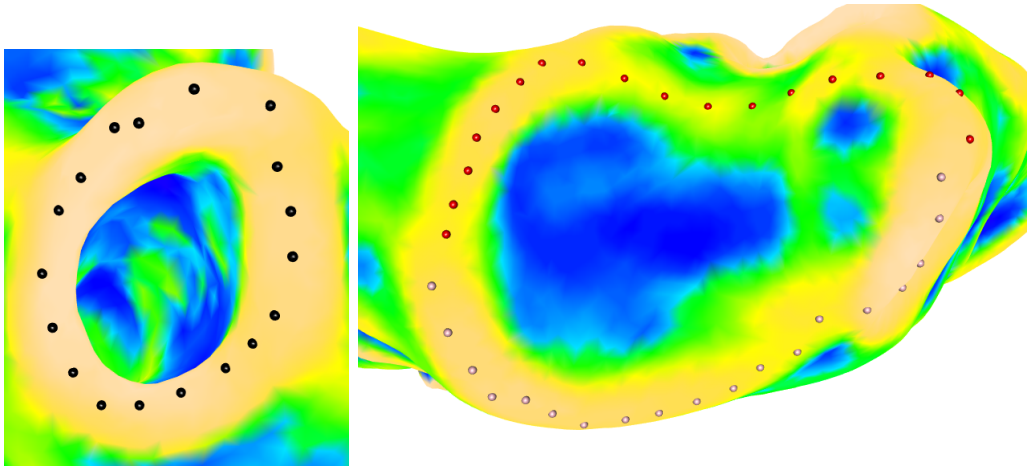


FIGURE 3.15: Left: an example of the identified ring-like curve; right: identification of the edge of a socket-like area on a fish jaw bone. Two halves of the edge are identified separately and shown in different colours (red and pink), as there are two landmarks given on this surface.

It should also be noticed that this method can be used to identify the ridges of different shapes. Figure 3.15 shows an example of identifying the edge of a socket-like area on the surface. Although the identified curve is still a circle, each half of the edge was identified independently, as there are two fixed anatomical

landmarks on both ends of the half-edge. Those two landmarks are required to be on the identified curves.

### Smoothing of the "initial" curve

The smoothing procedure aims to identify a new curve based on the initial curve, which has less variability. For each old point on the initial curve, this procedure creates a new point for its replacement on the curve.

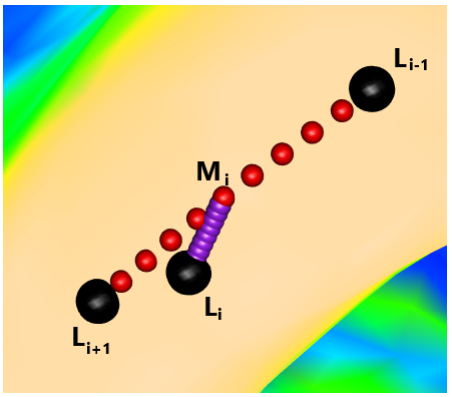



FIGURE 3.16: Smoothing procedure

Here is a procedure to adjust the position of $L_i$ according to $L_{i-1}$ and $L_{i+1}$ in Figure 3.16. A "plane cut" which has the shortest arc length between $L_{i-1}$ and $L_{i+1}$ is first created (consisting of red dots). One of the red points which has half arc length on this "plane cut" can be regarded as on the "average" position of $L_{i-1}$ and $L_{i+1}$. In order to take the position information of $L_i$ into account, the second "plane cut" (consisting of purple higher-density dots) which also has the shortest arc length is created from this middle red point $M_i$ to $L_i$. The one which has half arc length on the purple "plane cut" is used as the replacement of $L_i$.

Figure 3.17 compares the curve before smoothing and after twice smoothing. It can be noticed that the smoothed curve (red dots in the left and right figures) is much smoother than the initial curve (black dots in the left and middle figures) obtained in Section **??**. It seems that the final red curve have captured the shape of the ring-like ridge.

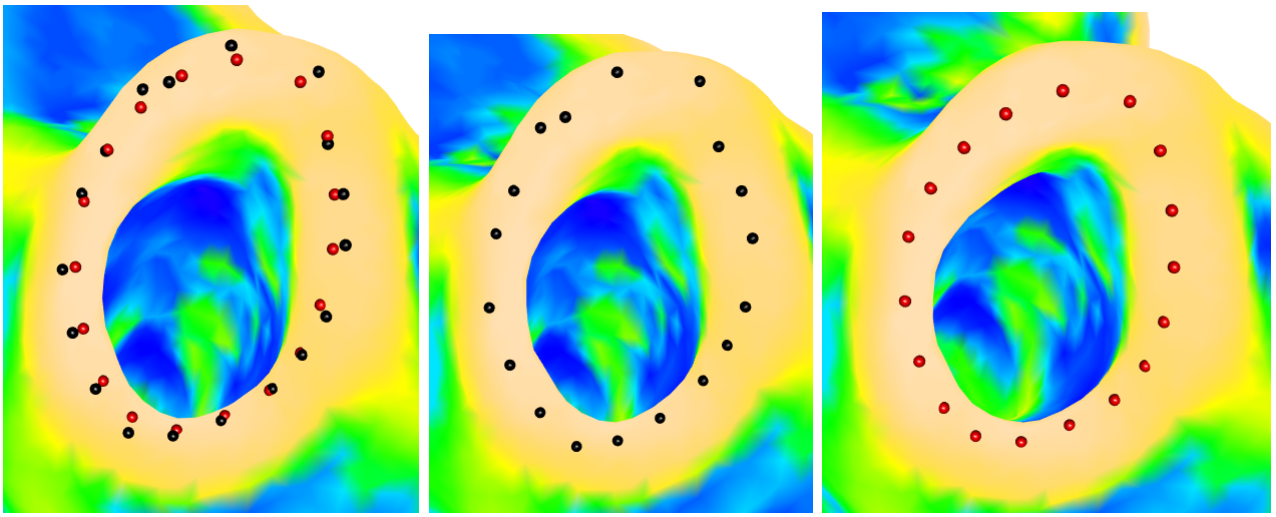


FIGURE 3.17: Left: a comparison of curves before and after the smoothing procedure; Middle: initial curve from Section 3.4.1; right: the result of smoothing the initial twice

Figure 3.18 shows an example of the estimated upper lip curve on a 3D image of a child who has a severe cleft lip. The curve is estimated based on two lip-corner landmarks which are manually placed and start from both two lip corners. The left figure is before smoothing and the right is after smoothing. It can be concluded that this curve estimation method has good performance on complex surfaces.

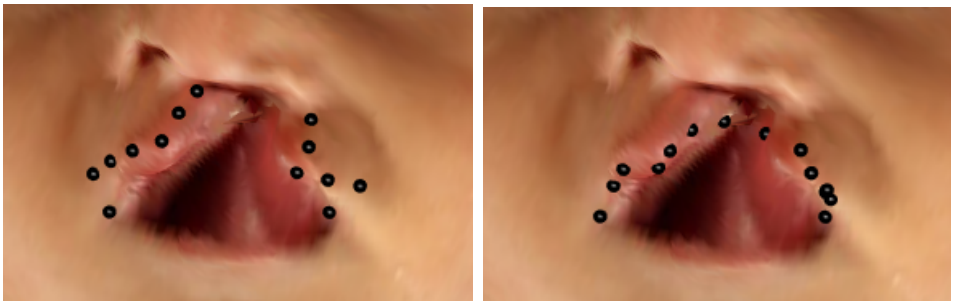


FIGURE 3.18: Before (left) and after (right) smoothing of the estimated upper lip curve on a 3D image of a child who has a severe cleft lip.

### 3.5.2 Simulation study

A simulation study is designed to evaluate the method in Section 3.4.1, which focuses on two example shapes, as shown in Figure 3.20: ring-like curve (left) and

cosine curve (right). After creating the shapes, the analytical true curve can be deduced so that the estimated curve can be compared. Different size of random noise is also added on the shapes to investigate the influence on the algorithm performance.

### Create the surface

The surface are created on a $100 \times 100$ grid with width 2 on 2D $x - y$ plane. The width unit is arbitrary. The third dimension of a ring-like ridge is created by

$$z = \left(2\pi\sigma^2\right)^{-\frac{1}{2}} \exp\left\{-\frac{(d_c - 1)^2}{2\sigma^2}\right\} \tag{3.12}$$

where $d_c$ denotes the Euclidean distance between any 2D point and the centre of the 2D $x - y$ grid. By this definition, the true ridge will revolve around the centre of the grid with radius 1, and have the height $z = (2\pi\sigma^2)^{-\frac{1}{2}}$. Figure 3.19 (left) shows this ring-like shape with the true curve (black dots).

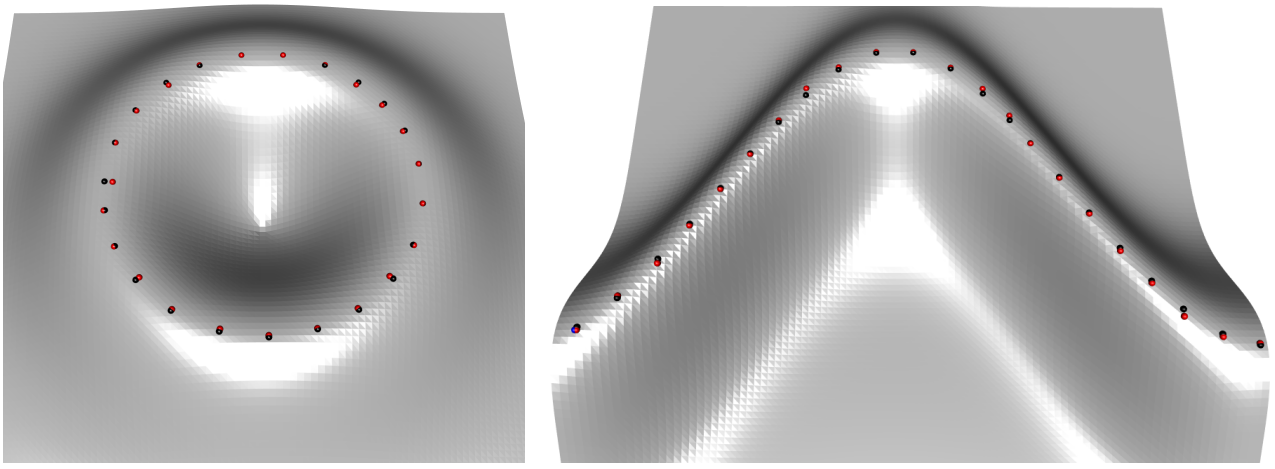


FIGURE 3.19: Left: A simulated ring-like ridge on a surface without noise; Right: A simulated cosine ridge on a surface without noise.

To simulate the surface with noise, a weighted Gaussian random field is added onto the surface. The weight function is $w = 1 - \exp\{-D\}$, where $D = (d_c - 1)^2$ for this ring-like ridge. This weight function ensures that the target ring-like ridge exists, no matter how much noise can be in the neighbourhood. Figure 3.20 (left) shows this ring-like shape with the true curve (black dots) and noise.

Similarly, the third dimension of a cosine ridge is created by

$$z = \left(2\pi\sigma^2\right)^{-\frac{1}{2}} \exp\left\{-\frac{\left(y - \cos\frac{3}{2}x\right)^2}{2\sigma^2}\right\} \tag{3.13}$$

By this definition, the true ridge will be in a cosine shape, and have the height $z = (2\pi\sigma^2)^{-\frac{1}{2}}$. Figure 3.19 (right) shows this cosine shape with the true curve (black dots). The weight function for noise is the $w = 1 - \exp\{-D\}$, where $D = \left(y - \cos\frac{3}{2}x\right)^2$. This weight function ensures that the cosine ridge exists, no matter how much noise can be in the neighbourhood. Figure 3.20 (right) shows this cosine shape with the true curve (black dots) and noise.

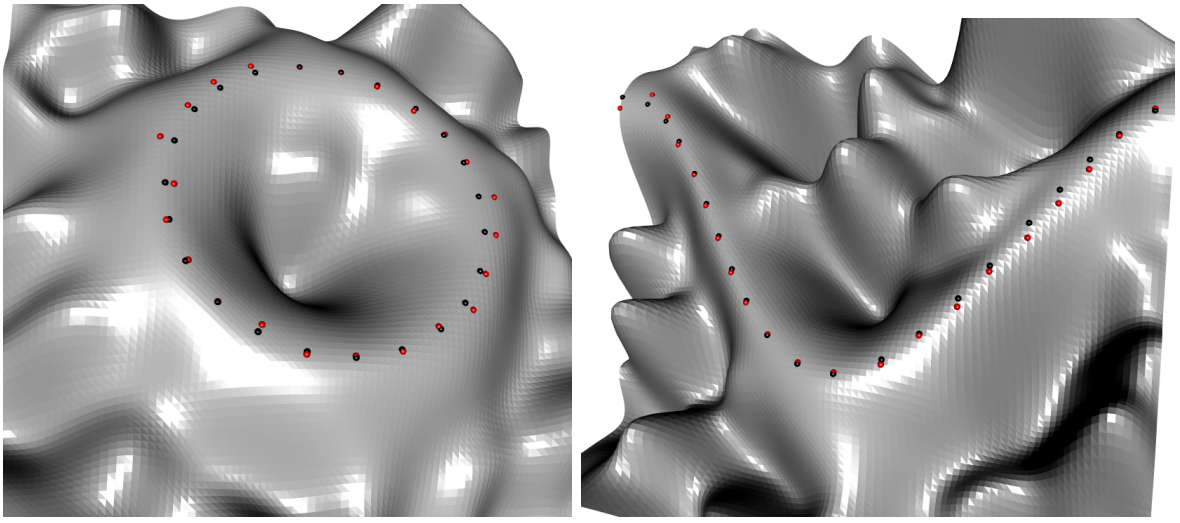


FIGURE 3.20: Left: A simulated ring-like ridge on a noisy surface; Right: A simulated cosine ridge on a noisy surface

### Estimate the ridge curves

The true curve of the ring-like ridge has the coordinates

$$\left(\cos\left(\arctan\frac{y}{x}\right), \sin\left(\arctan\frac{y}{x}\right), \left(2\pi\sigma^2\right)^{-\frac{1}{2}}\right) \tag{3.14}$$

and the true curve of the cosine ridge has the coordinates

$$\left(x, \cos\frac{3}{2}x, \left(2\pi\sigma^2\right)^{-\frac{1}{2}}\right) \tag{3.15}$$

Starting from a landmark which is manully placed on the ridge, the curve can be estimated by the algorithm in Section 3.4.1. It can be noticed from Step 3 in Section 3.4.1 that this estimation method depends heavily on the estimation of curvatures. As the curvatures are estimated by the local vertices, the size of the neighbourhood used can be an important parameter which may influence the result. Denote the radius of the neighbourhood by $d_r$. The optimal $d_r$ is investigated using the simulated shapes without noise, as shown in Figure 3.19. Red dots are the identified estimates of the ridge curves and black dots are obtained by the Formula 3.14 in Figure 3.19 (left) and the Formula 3.15 in Figure 3.19 (right).

Figure 3.21 shows the relationship between the average Euclidean distance of estimated points on the curve against the different radius of the neighbourhood $d_r$. It seems that in terms of these two shapes, the parameter does not have a large influence (0.3%) on the estimates of the curvatures when $0.1 < d_r < 0.2$. Thus, a radius of 0.15 is chosen to estimate the curvatures on these two shapes in the simulation.

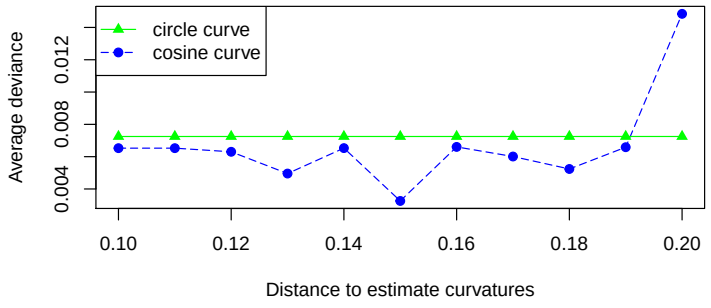



FIGURE 3.21: Influence of the neighbourhood on the estimation of curvatures

Figure 3.22 shows the influence of noise on the accuracy of the estimate of both curves. In Figure 3.20, the scale of noise added on the surface is 40%, which is a quite large scale on the surface. The noise has a larger or quicker influence on the estimate of the circular curve. However, even a 50% scale of noise has only 3.5% average deviance per point on the estimate on the curve.

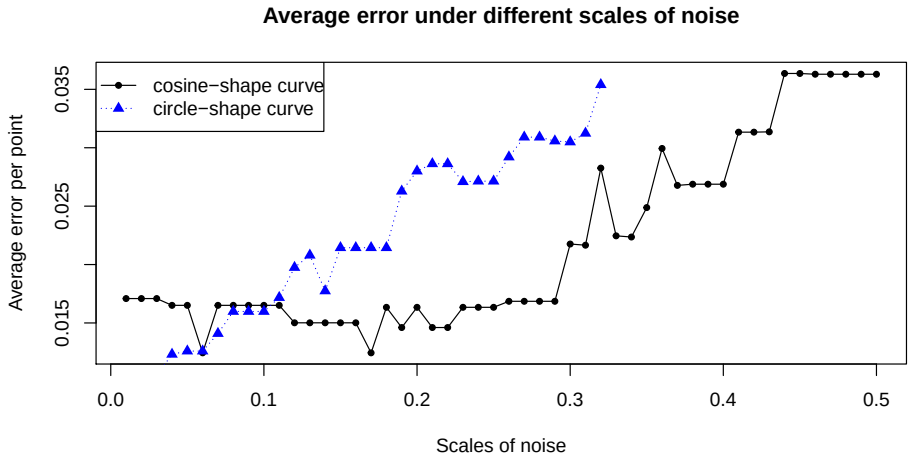
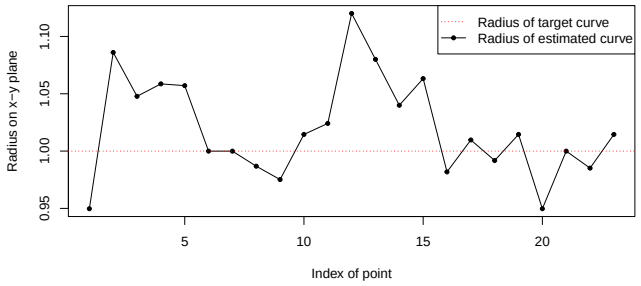


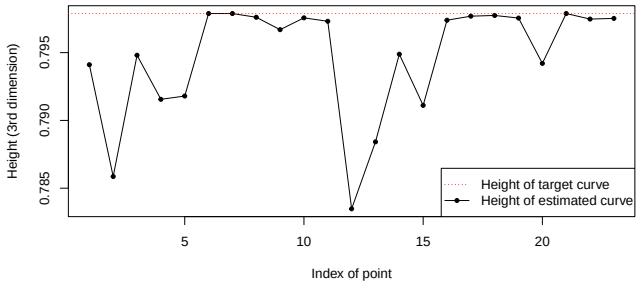

FIGURE 3.22: Influence of the noise scale on the estimation of curvatures

Figure 3.23 shows the decomposition of the deviance of the estimated curve (black solid line) from the truth (red dotted line) in Figure 3.20. For the circular curve, as the curve is a circle on x-y plane, the deviance is decomposed into the radius and height, as shown in Figure 3.23(a) and 3.23(b). For the cosine curve, as the estimated and target points have the same x-coordinates (in the first dimension), the deviance is decomposed into the second and third dimension, as shown in Figure 3.23(c) and 3.23(d). It seems that even with a large scale of noise, the deviance on the third dimension is at most 0.5% and other dimensions are about 2.5%, which are quite small.

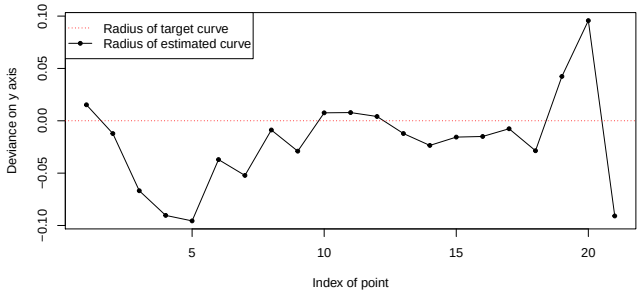
In conclusion, this method seems to do well in identifying curves with few landmarks and the error is within an acceptable scale of. In fact, it can be seen in Section 3.4.1, as long as there is no large noise close to the curve, the algorithm can obtain an appropriate estimation.



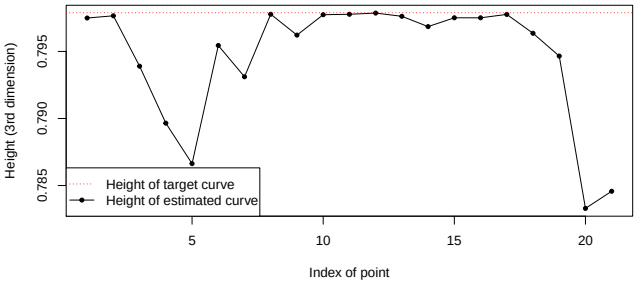

(a) Deviance of radius of the circular curve




(b) Deviance of height of the circular curve




(c) Deviance of 2nd dimension of the cosine curve




(d) Deviance of 3rd dimension of the cosine curve

FIGURE 3.23: Decomposition of the deviance of the estimated curve (black solid line) from the true curve (red dotted line) in Figure 3.20

## 3.6 Conclusion of 3D curve estimation

This chapter introduced two approaches to estimate 3D shape curves. The first contribution is the calculation of the geodesic distance which replaced the approximation with Euclidean distance. It enlarged the range of application of the existing method and made it capable to be applied on complex surfaces. The algorithm used the interpolation to reduce the computational cost and worked well on the fish mandibles. The second novelty made use of computational geometry to automatically estimate 3D shapes. The advantages include that it does not require the guidance of large number of landmarks and can deal with surfaces with gaps and holes. The simulation study indicated that the average deviance per point on the curve is 3.5% when there is 50% scale of random noise added on the surface. The performance of the algorithm is stable for 3D curve estimation.

# Chapter 4

# Analysis based on identified curves

There is a variety of local-feature-extracting applications of the estimated 3D curves in different disciplines. For example, existing technologies allow us to flexiblely cut a 2D photo along any curve we determine. This chapter will introduce a state-of-art method in Section 4.1 which dissects shapes along any estimated curves in 3D space. Another novelty in this chapter is the use of the estimated curves in morphology. Traditional methods in biology for shape analysis mainly focuses on 3D landmarks, while Section 4.2 introduces local feature extraction based on curves with an example of fish mandibles.

The estimated curves can also be used in existing popular shape analysis, such as principal component analysis for shape difference across individuals in Section 4.2.1, local feature alignment to remove phase variation Section 4.3 and local feature reconstruction in Section 4.4. A brief history of each topic will be included in each section.

## 4.1 Dissection of shapes

Recently, there are increasing demands in 3D shape analysis, and a particular need to study not only the whole surface but some specific parts has arisen. Meanwhile, thanks to the recent advancement of novel 3D scanning technology, the precision of a face model has become capable of showing details that were previously not available. Thereby, it comes to our interest that dissecting 3D objects can make a great contribution to surgery assessments by focusing on specific areas of interest.

The dissection of shapes is relevant to validation studies of 3D camera performance. It involves the comparison of the detailed outputs from the new 3D camera prototypes and established 3D capturing systems. The comparison is based on both the whole shape and on designated parts of interest. One example of the new prototype system is the assessment of cleft surgery. The comparison of pre- and post-surgical cleft lip areas can have more significant impacts on the result assessment than the comparison between whole faces.

In fact, mesh partitioning is not an extraordinary topic in shape analysis in the last two or three decades. Shamir [46], Theologou et al. [47] and Maglo et al. [48] provided comprehensive surveys on mesh segmentation techniques. As Shamir [46] mentioned, 3D mesh partition, or mesh segmentation, is useful to recognise and match shapes, map textures, edit meshes, make animations and so on. With the help of the next level of details, surgical assessment is an emerging application of the mesh partitioning.

There are many ways to partition the shape. For example, Theologou et al. [47] introduced a good example to partition an object, which is a ball attached to a cube. Part-type segmentation is to divide the object into a cube and a ball, while surface-type segmentation divides the object into different faces of the cube and a ball. Another way to categorise is the region-based segmentation, which takes the object apart into regions with similar properties (such as curvatures). While the boundary-based segmentation divides the shape into meaningful (such as anatomical) segments. Most research partitions the mesh by grouping the triangles

into parts, but does not pay attention to the smoothness of the boundaries, such as the methods introduced in Rodrigues et al. [49], Theologou et al. [50] and Jiao et al. [51]. Schoeler et al. [52] introduced a mesh partitioning method which only applied on point clouds and did not deal with the triangulation at the boundaries of the partitions.

Some research considers not only partitioning the mesh but also smoothing the boundaries or contours after segmentation. Minetto et al. [53] introduced a fast algorithm to slice a triangulated mesh by parallel planes which are perpendicular to a $z$-axis. By comparing the coordinates of the vertices on the $z$ axis, the algorithm groups the triangles and calculates the intersects (line segments) between the planes and the triangles. The contours of the sliced pieces are constructed by assembling the segments into a set of closed polygons, not necessarily triangles.

*FiberMesh* is another tool to deal with existing 3D meshes. Nealen et al. [54] described the mechanics which can deform a user-sketched 3D curve and erase the mesh along the curve. However, after the curve is determined, the intersected points are replaced by the middle of the adjacent vertices and the mesh is re-meshed by the method in Surazhsky and Gotsman [55].

To deal with the need for a more definitive comparison study, we have developed a method to dissect 3D shapes along a 3D curve, which will be later illustrated in Section 4.1.1. It not only divides the mesh into segments, but also produces two complete left and right pieces to the curve without throwing any information away. The mesh can be cut by flexible 3D curves, which may be created by the intersection of the surface and a series of non-parallel planes. The contours of the cuts are constructed by triangles and can be very smooth. Only the points along the 3D curve will be re-meshed and the original triangulation will remain as far as possible. Furthermore, the cut is clockwise and anti-clockwise orientation free. Note that the examples used in this section are 3D images captured by the Artec® camera system.

### 4.1.1 Identify a path and classify its surrounding points

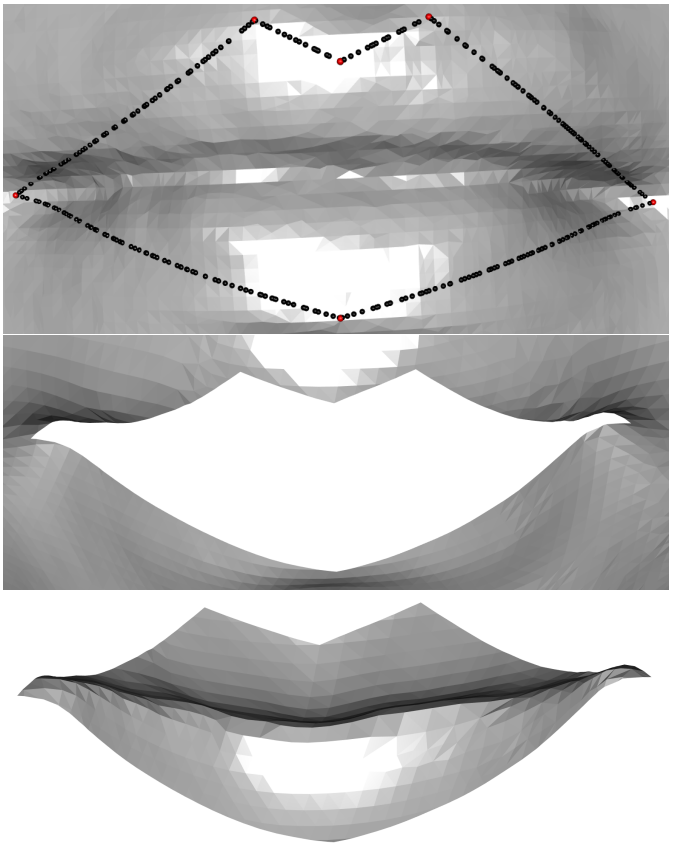


FIGURE 4.1: An example of the dissected two pieces of a 3D lip image.

Figure 4.1 shows an example of an identified path on a 3D lip image (top) and the resulting dissected shapes (middle, bottom). The path can be denoted as follows.

$$p = \{p_1, p_2, \ldots, \} \tag{4.1}$$

where $p_i = (x_i, y_i, z_i)$ denotes the discrete three dimensional points where the path crosses the edges of the triangulation, plus some additional landmark locations (red dots). The dissection is along the path (black dots) which is identified based on the manually placed landmarks. In other words, the whole surface will be cut along the path and the triangulated point cloud of the surface will be grouped into "inner" and "outer" areas, with respect to the path.

### 4.1.2 Assumptions of dissection algorithm

- The points on the path follow the same anti-clockwise order with the vertices of a triangle on the shape;
- Any adjacent pair of points on the path does not cross any other edges;
- There are no duplicated points in the path;
- The dissected shape is disconnected[1] but the path is not necessarily closed. In other words, the edge of the surface can be part of the path.

### 4.1.3 Principle of dissection

As the path crosses the edges of the triangles, most of $p_i$s lie on the edges of triangles, apart from a small number of landmarks. The two vertices at the ends of each crossed edge should belong to the left or right, with respect to the path. As the shape is composed of vertices which are connected by triangles, the principle of splitting the shape along the path begins by excluding the triangles whose edges are crossed by the path.

After removing the triangles, the two resulting shapes each have a new triangulation which includes the path points $p_i$s. In other words, both of them have the path as a new edge, creating two complete shapes with new triangulations along the path.

### 4.1.4 Algorithm of dissection

Step 1: Follow the order of the points on the path and find the edges of the triangles which are crossed;

Step 2: Following the order of the points on the path, identify the left and right groups of the vertices on the edges;

[1] A topological space is said to be disconnected if it is the union of two disjoint non-empty open sets.

Step 3: Find the triangles which have those crossed edges and exclude those triangles from the original triangulation. Identify the left and right pieces of the surface;

Step 4: Create a new triangulation for left and right pieces which have the dissection path as an edge. New triangles are built by adding new groups of three 3D coordinates.

### 4.1.5 Details of the algorithm

#### Details of Step 2

Denote the vertices by $v_{i1}$ and $v_{i2}$ on the edge which is crossed by the path at $p_i$. The procedure for identifying the left and right groups of $v_{i1}$ and $v_{i2}$ is as follows.

1. Approximate the normal vector $\vec{n}$ at $p_i$ by the average of the normal vectors of $v_{i1}$ and $v_{i2}$;
2. Calculate the direction vector $\vec{d} = p_{i+1} - p_i$ at $p_i$;
3. Create an axis $\vec{c}$ by the cross product of the normal vector $\vec{n}$ and the direction vector $\vec{d}$ at $p_i$;
4. Calculate the projections of $v_{i1}$ and $v_{i2}$ onto the axis $\vec{c}$. If there is only one crossing point on the edge, the vertex with smaller projection will be in the right group and the other is in the left group, with respect to the path;
5. If the edge is crossed by more than one point on the path, compare the projection of the vertices only with the points which are the closest;
6. If the previous and subsequent points cross the same edge and these three points are within the same triangle, the current point is not used to identify groups.

As shown in Figure 4.1, if the points' order on the path is anticlockwise, the shape is divided into the left (bottom) and right (middle) pieces to the path. We call one

the "inner" surface when focusing on a piece and the "outer" surface for the other. For instance, when we study and build a new triangulation for the left surface, "inner" refers to the left piece with respect to the path and "outer" refers to the right piece. In the figures, the "inner" surface is shown in blue shadow and the "outer" surface is shown blank.

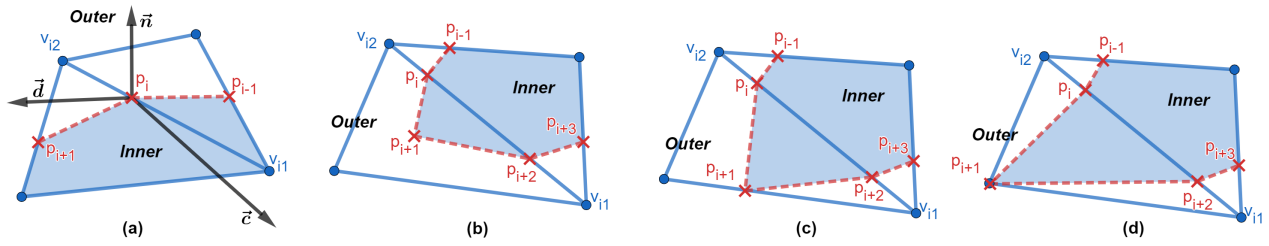



FIGURE 4.2: Identify the inner and outer groups of the vertices of the edges which are crossed by the path (red dashed lines).

As shown in Figure 4.2 (a), the surface is represented by the blue solid vertices and straight blue edges, and the red dashed lines are the identified path where the points $p_i$ lie. The figure focuses on the left piece with respect to the path, so above the path is the outer area and below is the inner area.

Figure 4.2 (a) describes Procedure 1 to 4 of this section, in the case that the edge $\overrightarrow{v_{i1}v_{i2}}$ is crossed by the path only once. The normal vector of $p_i$, denoted by $\vec{n}$, is approximated by the average of the normal vectors of the vertices $v_{i1}$ and $v_{i2}$. An axis, denoted by $\vec{n}$, can be created by the cross product of $\vec{n}$ and the direction vector $\vec{d}$. The projections of vertices $v_{i1}$, $v_{i2}$ and the point $p_i$ on to this axis can be used to specify the inner and outer groups. In this example, vertex $v_{i1}$ will be classified as an inner coordinate, as its projection onto the axis is larger than the projection of $p_i$. Similarly, vertex $v_{i2}$ will be classified as an outer coordinate, as its projection onto the axis is smaller than the projection of $p_i$.

There are two special cases to consider.

**Multi-crossings** When edge $\overrightarrow{v_{i1}v_{i2}}$ is crossed twice or more, as shown in Figure 4.2 (b,c,d), the projections of the vertices $v_{i1}$ and $v_{i2}$ will be compared only

with the projections of their closest points on the path. In this example, vertex $v_{i1}$ will be identified as outer by comparing with the projection of $p_i$, and vertex $v_{i2}$ will be identified as outer by comparing with the projection of $p_{i+2}$.

**Repeated identification** When there is a sharp change of the path on a point $p_{i+1}$, as shown in Figure 4.2 (c), if $p_{i+1}$ lies on an edge of a triangle, the vertex $v_{i1}$ will be identified again but as in the inner group when the algorithm comes to the crossing point $p_{i+1}$. So there is a condition that, no matter whether the point $p_j$ lies on an edge ($j = i + 1$ in Figure 4.2 b), inside a triangle (Figure 4.2 c) or on a vertex (Figure 4.2 d), as long as $p_{j+1}$ and $p_{j-1}$ are on the same edge and in the same triangle with $p_j$, none of the vertices are identified based on $p_j$.

### Details of Step 3

Figure 4.3 shows the surface with exclusion of the triangles whose edges are crossed by the path (red dots). The shape can be split by cutting along the path.

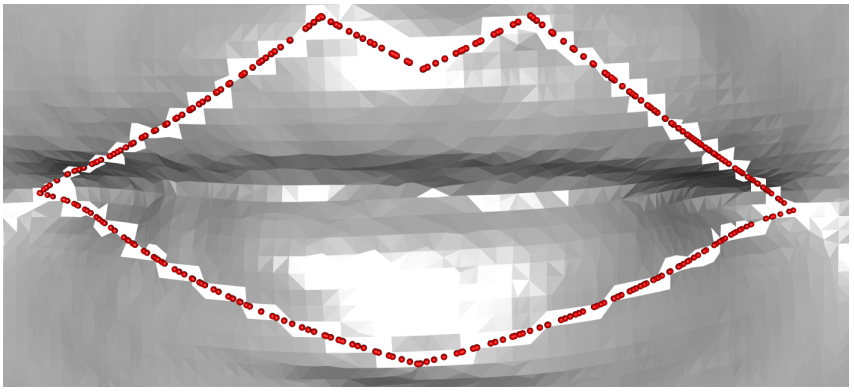


FIGURE 4.3: The triangles whose edges are crossed by the path (red dots) are deleted to separate the surface.

### Details of Step 4

After the exclusion of the triangles, the left and right area of the shape can be separated by the groups identified in Step 2. In order to complete the dissection,

the path (red dots in Figure 4.3) should be added into the triangulation of the inner and outer surface as an edge.

The idea is to make use of the triangulation which was present before the shape was split. Denote the left piece of shape by $S_L$ and the right piece of shape by $S_R$; the triangulation of the shape before split by $T$; the triangulation of the left piece after the split by $T_L$ and the triangulation of the right piece after the split by $T_R$. When the path is anti-clockwise to $S_L$, it is clockwise to $S_R$. Thus, if $\triangle p_i p_j p_k$ belongs to $T_L$, $\triangle p_k p_j p_i$ will belong to $T_R$.

In Section 4.1.5, for every point on an edge of $T$, the indices of the left and right vertices on the same edge can be recorded. Define a function $t : \mathrm{I}_p \rightarrow \mathrm{I}_v$, where $\mathrm{I}_p$ denotes the indices of points lying on the path, $\mathrm{I}_v$ denotes the indices of vertices which are the closest on the surface to the path; $t(i)$ is the index of the "inner" vertex which lies on the same edge as the $i$th point on the path. The function $t$ is neither injective nor surjective.

The algorithm connects the edge $p_i p_{i+1}$ to the "inner" surface. For the simplest case, there are three situations when $p_i$ is on an edge of $T$ alone or, in other words, $t(i) \in \mathrm{I}_v$.

- If $t(i+1) \notin I_v$, or if $t(i+1) \in I_v$ and $t(i+1) = t(i)$, add triangle $\triangle p_{i+1} v_{t(i)} p_i$ (Figure 4.4 a and b);
- If $t(i+1) \in I_v$ but $t(i+1) \neq t(i)$, add triangles $\triangle p_i v_{t(i+1)} v_{t(i)}$ and $\triangle p_{i+1} v_{t(i+1)} p_i$ (Figure 4.4 c).

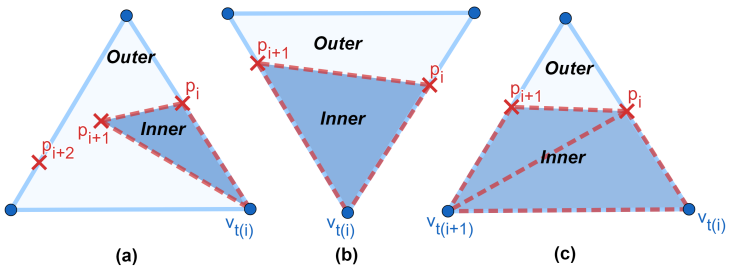



FIGURE 4.4: Situations when the $i$th point $p_i$ on the path is on an edge of $T$

when $p_i$ is not on the edge of $T$ alone or, in other words, $t(i) \notin \mathrm{I}_v$,

- If $t(i+1) \notin I_v$, use the last vertex in use $v_{t(i-m)}$ to connect and add triangle $\triangle p_{i+1} v_{t(i-m)} p_i$ (Figure 4.5 a);
- If $t(i+1) \in I_v$ but $t(i-1) \notin I_v$ and $p_i$ lies on an edge which is crossed by the path twice or more, or if $t(i+1), t(i-1) \in I_v$ and $t(i+1) = t(i-1)$, add triangle $\triangle p_{i+1} v_{t(i+1)} p_i$ (Figure 4.5 b and c);
- If $t(i+1), t(i-1) \in I_v$ and $t(i+1) \neq t(i-1)$, add triangles $\triangle p_i v_{t(i)} v_{t(i-1)}$ and $\triangle p_{i+1} v_{t(i)} p_i$ (Figure 4.5 d).

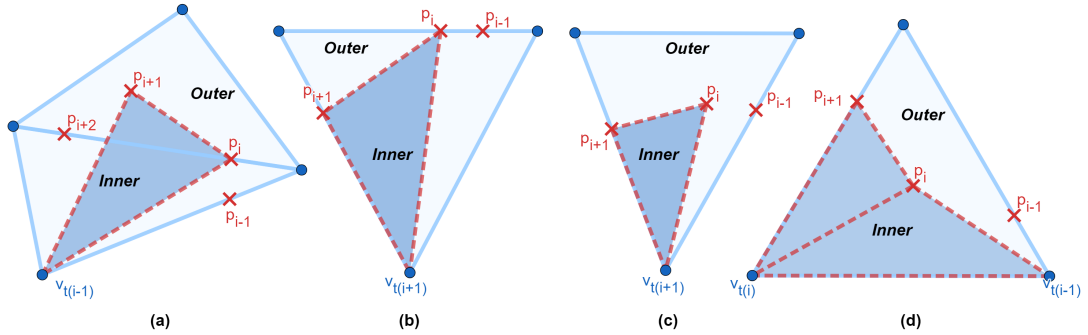



FIGURE 4.5: Situations when the i-th point $p_i$ on the path is not on an edge

There is a special set of rules for a concave[2] surface. If $p_i$ is on an edge of $T$ and $p_{i-1}$ and $p_{i+1}$ cross the same edge and are within the same triangle with $p_i$,

- If the path is clockwise to the "inner" piece (build right piece of the shape), add triangles $\triangle v_{t(i-1)} v_{t(i-1)+1} p_i$ and $\triangle p_i v_{t(i+1)} p_{i+1}$ (Figure 4.6 a);
- If the path is anti-clockwise to the "inner" piece (build left piece of the shape), add triangles $\triangle p_i v_{t(i+1)} v_{t(i-1)+1}$ and $\triangle p_{i+1} v_{t(i+1)} p_i$ (Figure 4.6 c);

If $p_i$ is inside a triangle of $T$ and $p_{i-1}$ and $p_{i+1}$ cross the same edge and are within the same triangle with $p_i$,

[2]A concave polygon will always have at least one reflex interior angle, that is, an angle with a measure that is between 180°and 360°exclusive.

- If the path is clockwise to the "inner" piece (build right piece of the shape), add triangles $\triangle v_{t(i-1)}v_{t(i-1)+1}p_i$, $\triangle v_{t(i-1)+1}v_{t(i+1)}p_i$ and $\triangle p_i v_{t(i+1)}p_{i+1}$ (Figure 4.6 b);

- If the path is anti-clockwise to the "inner" piece (build left piece of the shape), add triangles $\triangle p_i v_{t(i-1)+1}v_{t(i-1)}$, $\triangle p_i v_{t(i+1)}v_{t(i-1)+1}$ and $\triangle p_{i+1}v_{t(i+1)}p_i$ (Figure 4.6 d);

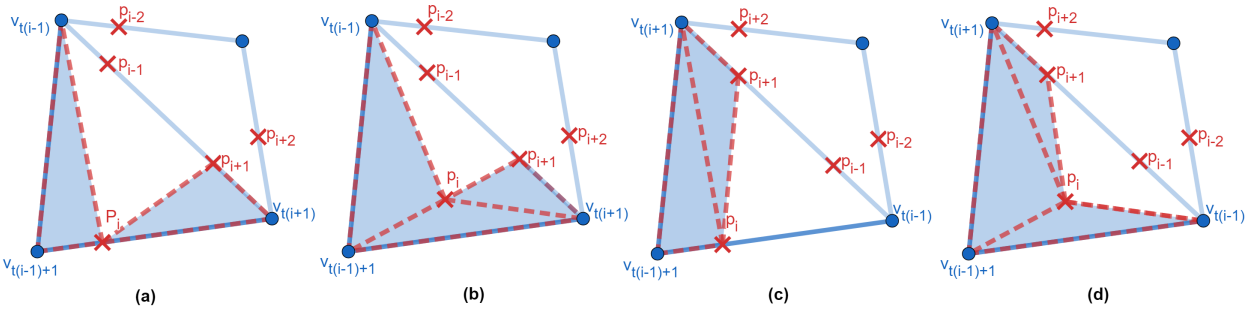



FIGURE 4.6: Situations when the i-th point $p_i$ on the path is at a concave "inner" surface

## 4.2 Further analysis on fish mandibles

The mandible is a core anatomical structure in feeding and so its shape variation during species evolution are ecologically important in biology. There are 119 fish mandibles (52 females and 67 males) disarticulated from the upper jaw and scanned using a Bruker Skyscanner machine (model 1172; Bruker, Billerica MA) which is located at the University of Strathclyde in Glasgow, as introduced by McWhinnie [56]. There are 34 landmarks on each shape which are manually placed using ©IDAV system *Landmark*. The data are provided by Dr McWhinnie.

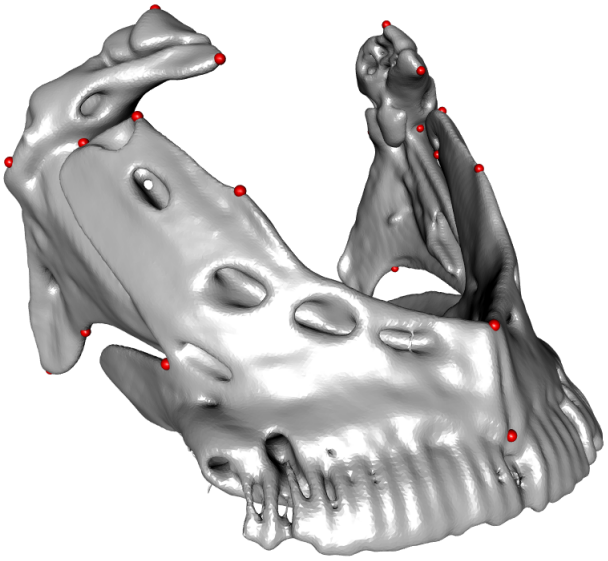


FIGURE 4.7: An example of a fish mandible with 34 manually placed landmarks (red dots).

The local feature of the fish mandibles will be modelled by five anatomical curves (configurations), which are estimated by the methods described in Chapter 3. The technical materials in this section are introduced in Section 2.2.2. The coordinate matrix of the configuration, such as points or curves, is Procrustes matched and vectorised. PCA is applied on the Procrustes residuals which are defined in Equation 2.13. Different principal components determine different directions of each point of the configuration. For example, the left and right lip corner on a lip curve will have the first principal component to the left and right correspondingly and a horizontally longer lip will have a larger score on this principal component.

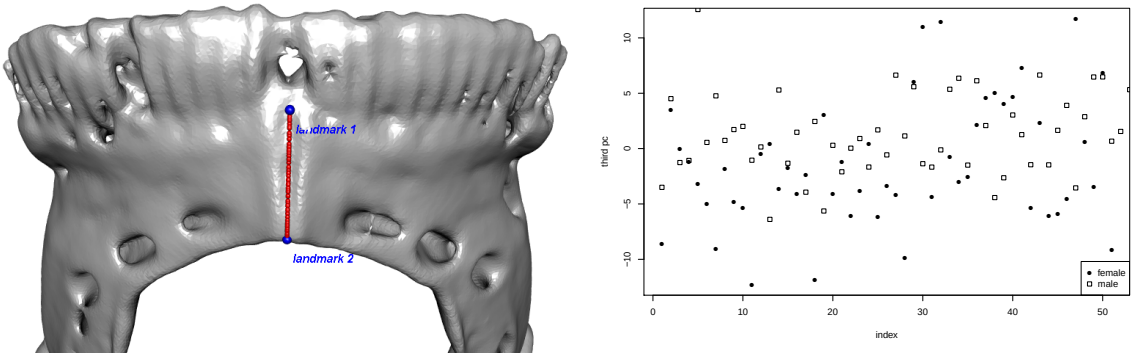
The principal component scores in Equation 2.15 are used in t-tests to investigate whether there is a significant average jaw-shape difference between female and male fish mandibles. As defined in Equation 2.15, the average of the whole sample score on a PC should be zero. So if the group score difference is significant on any PC, there will be positive and negative averages in two groups, which can also be noticed by the score plots.

Furthermore, as introduced in Equation 2.17, "typical" configurations in each group, also called extreme cases to lie two-standard-deviation from the mean shape

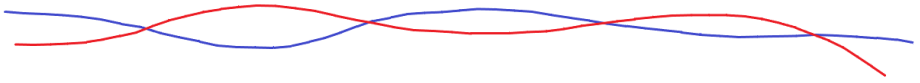
of the group so that they show clearer features of the shape group. Besides, Linear Discriminant Analysis (LDA) is applied to predict the gender from the shapes, with prediction accuracy provided. The shape variation of an individual curve and grouped curves will be analysed. The full analysis of those 3D models will be illustrated in this section.

### 4.2.1 Local and general feature extraction

The mid-line mandible curve, Curve 1, is estimated as shown in Figure 4.8 (a) left. After Procrustes matching of 119 estimated curves, the 3rd (p-value=0.005<0.05) principal component is significant at the 5% level. It explains 16% of the variance of the total variation of this curve. In Figure 4.8 (a) right, the scores of the significant pc are plotted and the extreme cases of all mandibles for Curve 1 are shown in (b). Note that the blue curve in an extreme-case plot always refers to the point which has a positive score on this pc in the score plot. It seems that males (the group that the blue curve in Figure 4.8 (b) belongs to) have a hill (pointing out of the surface) in the middle of the curve, while females (red curve in Figure 4.8 bottom) have a corresponding valley shape in the middle and two small hills close to two landmarks of the curve. The result of LDA shows that the correct classification rate of gender based on Curve 1 is 66.2%.



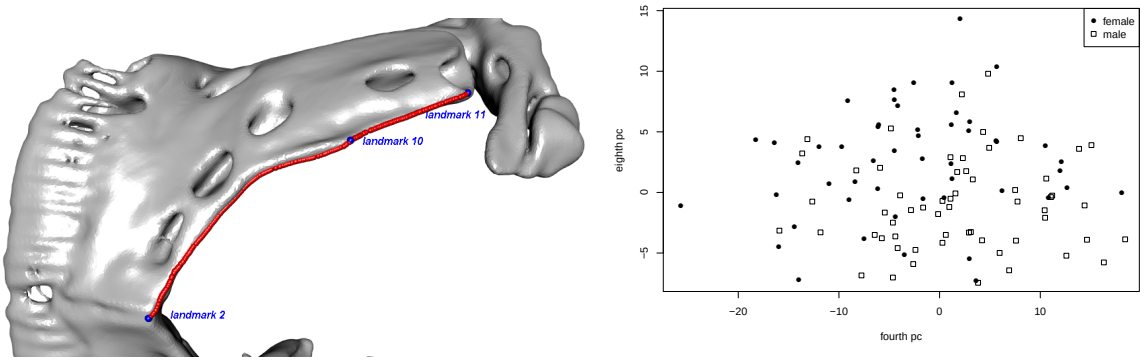

(a)



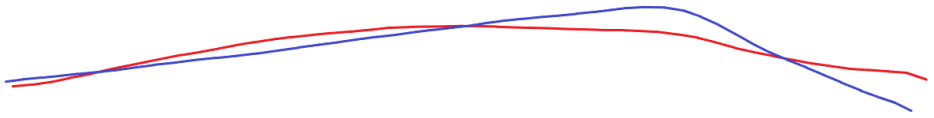
(b)

FIGURE 4.8: (a) left: an example of the estimated Curve 1 (red line) based on landmark 1 and 2 from top to bottom; (a) right: a score plot of the third principal component after Procrustes matching and PCA, specified in gender; (b): a screen shot of two 3D extreme cases on the 3rd pc of Curve 1 (landmark 1 on the left and blue for males).

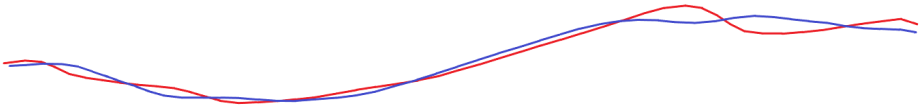
The upper-right mandible curve, Curve 2, is estimated as shown in Figure 4.9 (a) left. After Procrustes matching of 119 estimated curves, the 4th (p-value=0.005<0.05) and 8th (p-value=0.000<0.05) pcs are significant at the 5% level. They explain 12% of the variance of the total variation of this curve. In Figure 4.9 (a) right, the scores of the significant pcs are plotted and the extreme cases for Curve 2 are shown in Figure 4.9 (b) and (c). It seems that males have a sharper shape around landmark 10 and male shapes fluctuate more between landmarks 10 and 11. For linear discriminant analysis on curve 2, the correct classification rate is 72.0%.




(a)



(b)



(c)

FIGURE 4.9: (a) left: an example of the estimated Curve 2 (red line) based on landmark 2, 10 and 11 (blue dots); (a) right: a score plot of two significant principal components after Procrustes matching and PCA, specified in gender; (b): a screen shot of two 3D extreme cases on the fourth pc of Curve 2 (landmark 2 on the left and blue for males); (c): a screen shot of two 3D extreme cases on the eighth pc of Curve 2 (landmark 2 on the left and blue for females).

The upper-left mandible curve, Curve 3, is estimated as shown in Figure 4.10. After Procrustes matching 119 estimated curves, the 5th (p-value=0.006<0.05), 8th (p-value=0.005<0.05) and 9th (p-value=0.017<0.05) pcs is significant at the 5% level. They explain 13% of the variance of the total variation of this curve. In Figure 4.11 (a), the scores of the significant pc are plotted and the extreme cases for Curve 1 are shown in Figure 4.11 (b) and (c). Similar to curve 2, females tend to have a sharper shape around landmark 19, and they fluctuate more on this curve. For linear discriminant analysis on curve 3, the classification rate is 64.7%.

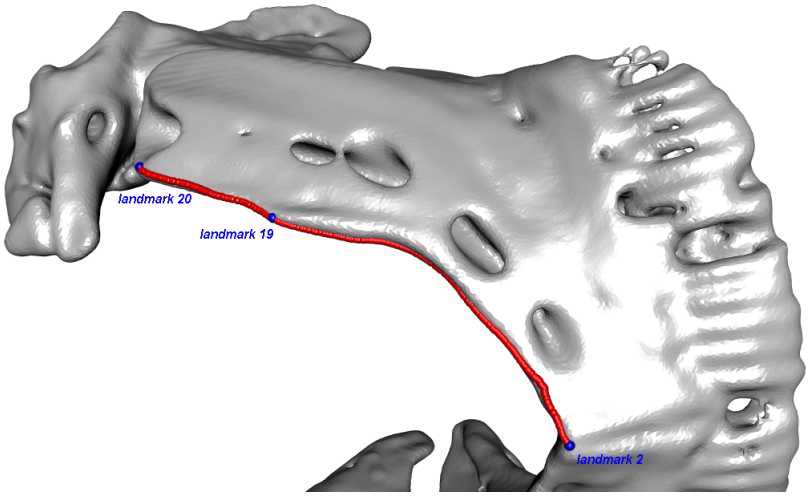
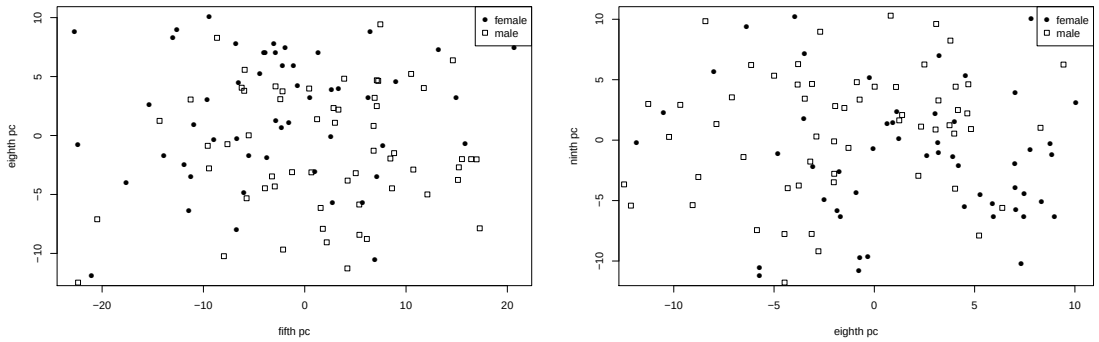
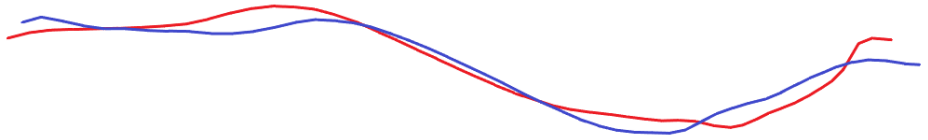



FIGURE 4.10: An example of the estimated Curve 3 (red line) based on landmark 2, 19 and 20 (blue dots).

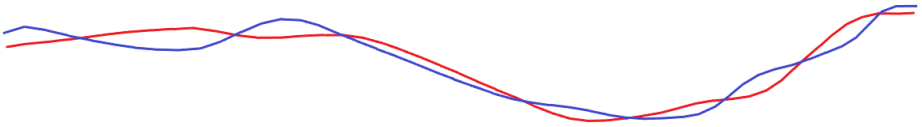
The backward-right curve, Curve 4, is estimated as shown in Figure 4.12 (a) left. After Procrustes matching 119 estimated curves, the 7th (p-value=0.012<0.05) and 8th (p-value=0.006<0.05) pcs are significant at the 5% level. They explain 6% of the variance of curve 4. In Figure 4.12, the scores of the significant pc are plotted in (a) right and the extreme cases for the 7th and 8th pcs of Curve 4 are shown in (b) and (c) respectively. It seems that males (blue line in (b)) tend to have a sharper and higher shape at landmark 29, and a longer valley between landmarks 6 and 29, while female (blue line in (c)) shapes tend to have a narrow valley. Additionally, males (red line in (b)) have greater fluctuation in shape between landmark 5 and 29 with a hill close to landmark 5. For linear discriminant analysis on curve 4, and the classification rate is 60.1%.



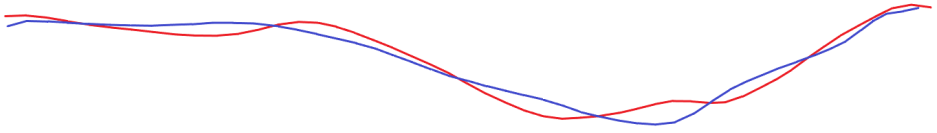

(a)



(b)

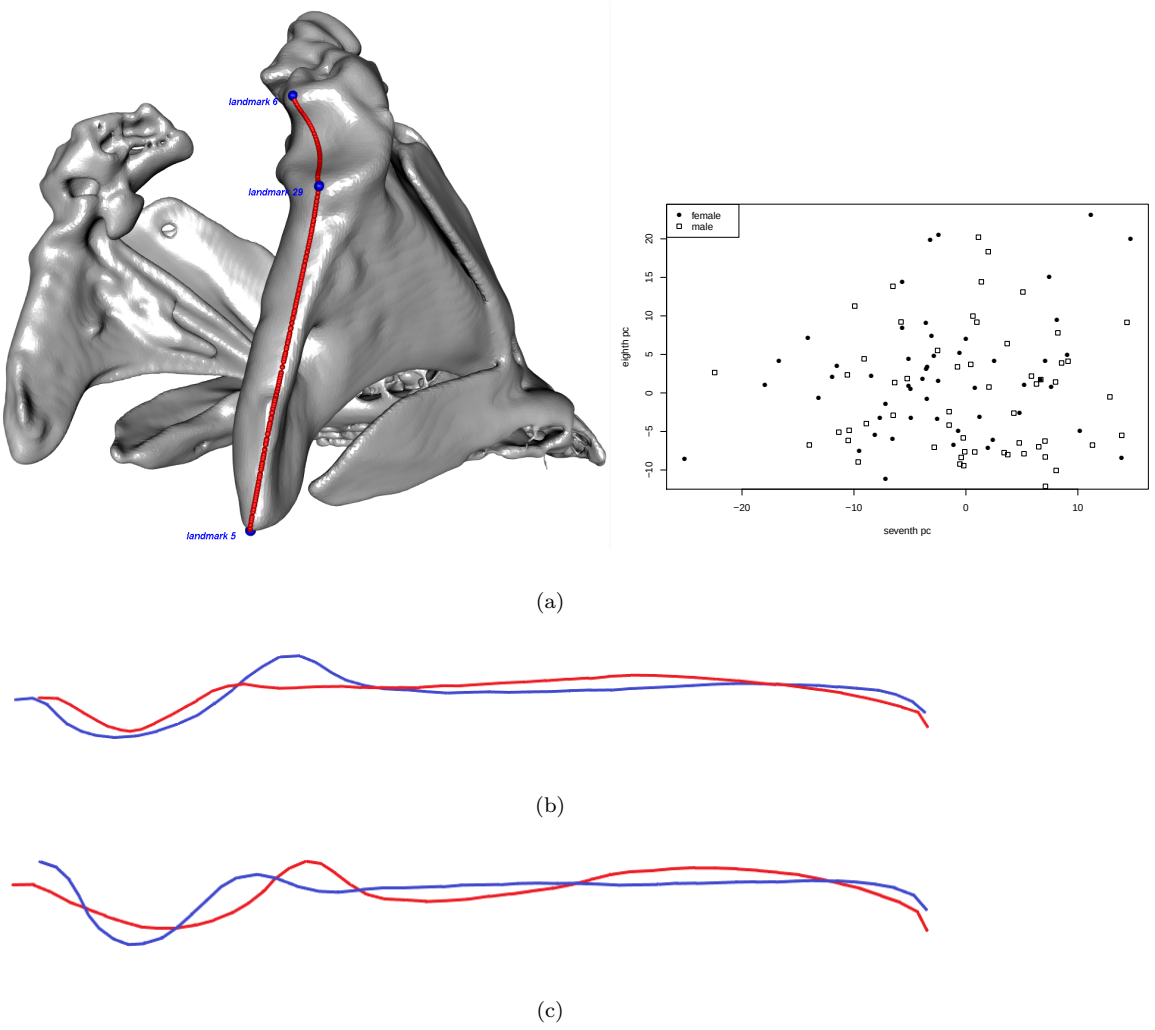


(c)



(d)

FIGURE 4.11: (a): score plots of the significant principal components after Procrustes matching and PCA, specified in gender; (b), (c) and (d): screen shots of 3D extreme cases on the fifth, eighth and ninth pc for Curve 3 (landmark 2 on the right and blue for males in (b) and (d); blue for females in (c)).




(a)

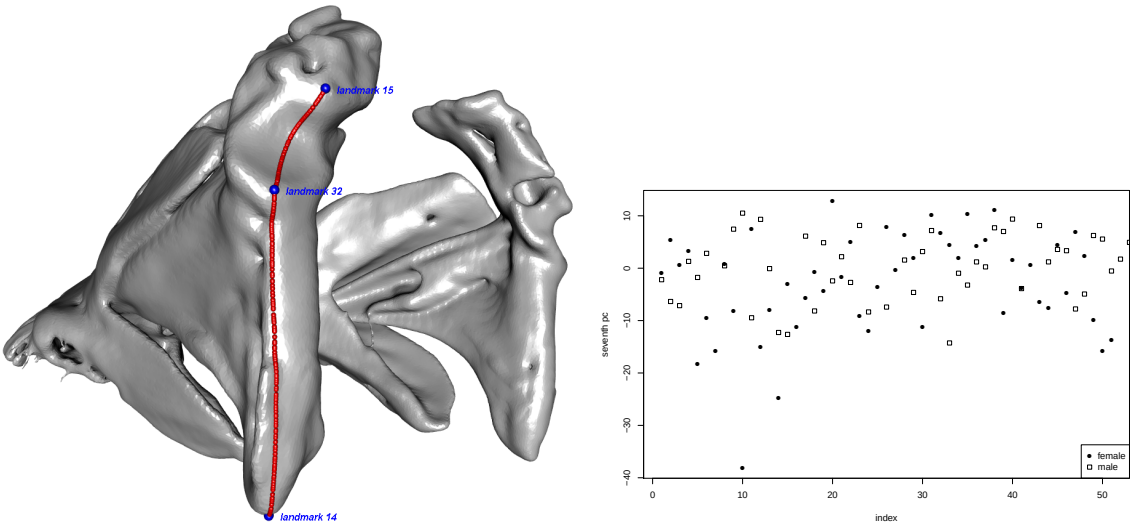
(b)

(c)

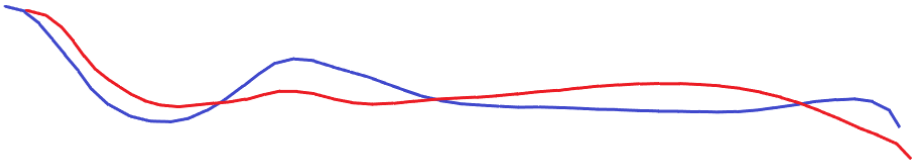
FIGURE 4.12: (a) left: an example of the estimated Curve 4 (red line) based on landmark 5, 29 and 6 (blue dots); (a) right: a score plot of two significant principal components after Procrustes matching and PCA, specified in gender; (b): a screen shot of two 3D extreme cases on the seventh pc of Curve 2 (landmark 2 on the left and blue for males); (c): a screen shot of two 3D extreme cases on the eighth pc of Curve 2 (landmark 5 on the right and blue for males in (b); blue for females in (c)).

The backward-left curve, Curve 5, is estimated as shown in Figure 4.13 (a) left. After Procrustes matching 119 estimated curves, the 7th (p-value=0.006<0.05) pc is significant at the 5% level. It explains 4% of the variance of curve 5. In Figure 4.13, the scores of the significant pcs are plotted in (a) right and the extreme cases for Curve 5 are shown in (b). It seems that males (blue line in (b)) tend to have

a sharper shape at landmark 32 and a deeper valley between landmark 32 and 15, while females tend to have a hill between landmarks 14 and 32. For linear discriminant analysis on curve 5, the classification rate is 57.0%.




(a)



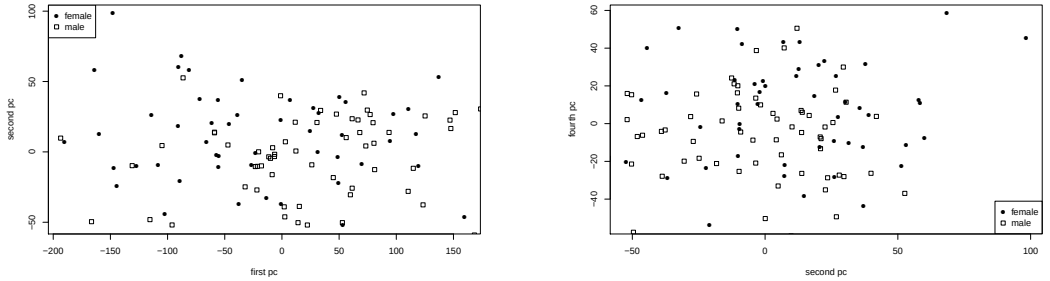
(b)

FIGURE 4.13: (a) left: an example of the estimated Curve 5 (red line) based on landmark 14, 32 and 15 (blue dots); (a) right: a score plot of the seventh principal component after Procrustes matching and PCA, specified in gender; (b): a screenshot of two 3D extreme cases on the seventh pc of Curve 5 (landmark 14 on the right and blue for males).

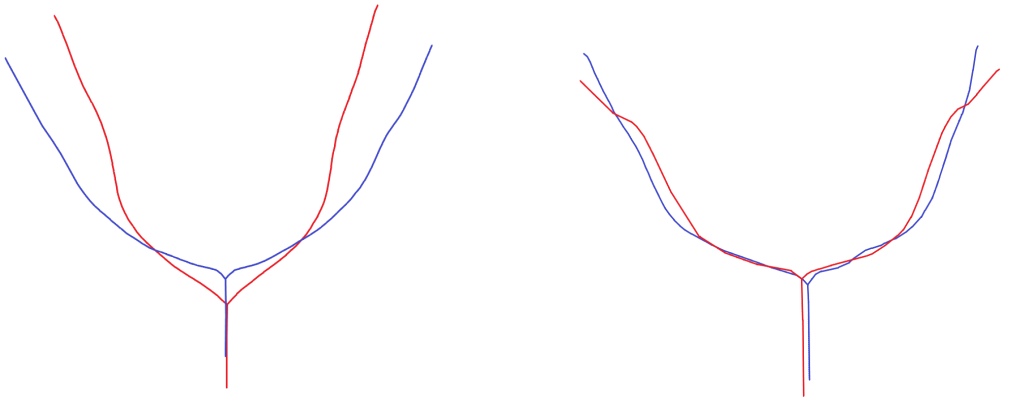
The upper mandible consists of Curve 1, Curve 2 and Curve 3. After Procrustes matching 119 estimated curve sets, the 1st (p-value=0.017<0.05), 2nd (p-value=0.002<0.05) and 4th (p-value=0.009<0.05) pcs are significant at the 5% level. They explain 70% (57% on the 1st pc) of the variance of the upper mandible. The scores of the significant pcs are plotted in Figure 4.14 (a). The extreme cases for each principal component are shown in (b) left for the first; (b) right and (c) for

the second (top and side view) and (d) for the fourth. It seems that females (red line at the top left) tend to have sharper and narrower shapes, while males tend to have shorter forward and backward, and wider left and right shapes, which are more like semicircles (in (b) left). Besides, male shapes (red line in (b) right) tend to have sharper changes at landmarks 10 and 19, and wider ends at landmarks 11 and 20. Moreover, females tend to have a sharper hill at landmark 2 but a flatter shape elsewhere (blue lines in (c) and (d)). For linear discriminant analysis, the classification rate is 69.2%.

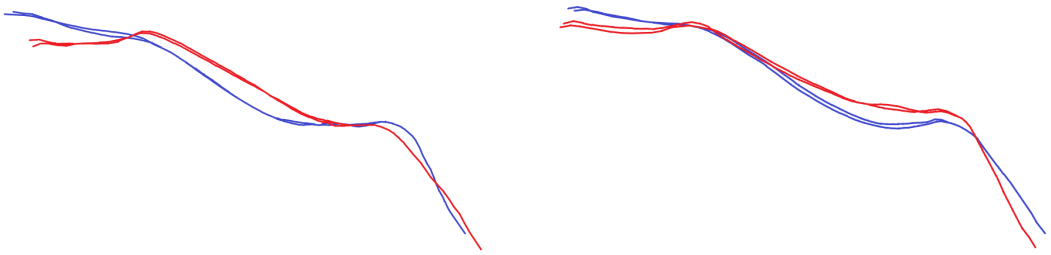
The five curves together outline the whole mandible. After Procrustes matching 119 estimated curve sets, the 2nd (p-value=0.000<0.05), 3rd (p-value=0.000<0.05) and 4th (p-value=0.007<0.05) and 10th (p-value=0.008<0.05) pcs are significant at the 5% level. They explain 26% of the variance of this model. The scores of the significant pcs are plotted in Figure 4.15 (a) and the extreme cases for all five curves are shown in (b) and (c). In Figure 4.15, it seems that, compared with males, females tend to have narrow shapes with tightened bottom ends on curve 4 and 5 (blue lines in (b)); males tend to have more skewed shapes to their right and sharper hills at intermediate landmarks (blue lines in (c)). For LDA on all three curves together, the classification rate is 76.8%.




(a)

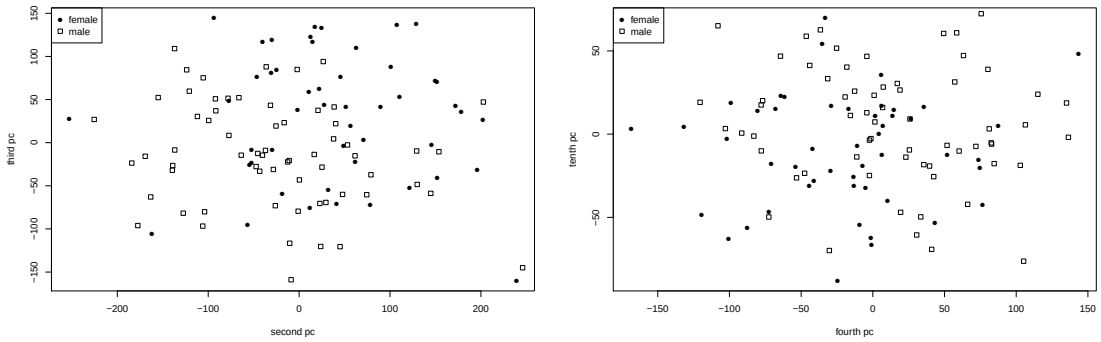


(b)

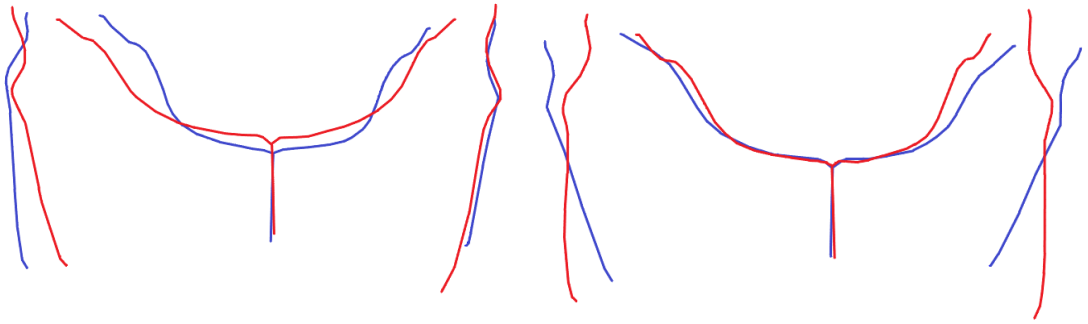


(c)

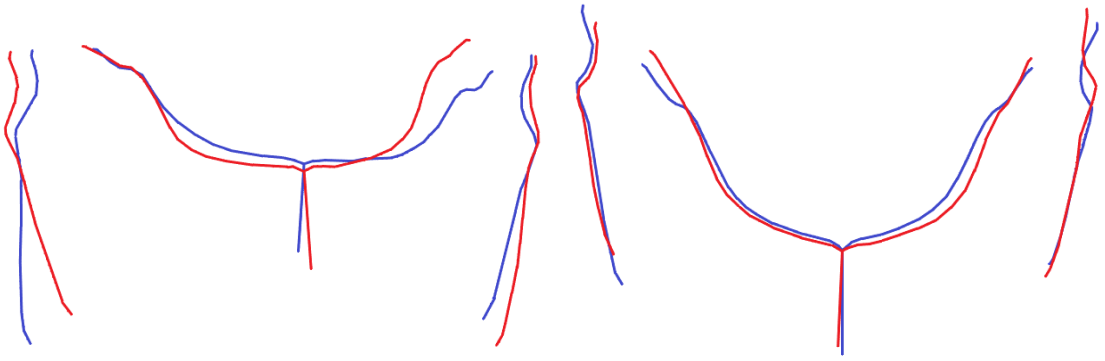
FIGURE 4.14: (a): score plots of the significant principal components after Procrustes matching and PCA, specified in gender; (b) left: a screenshot of two 3D extreme cases on the first pc of the upper mandible (landmark 2 at the bottom and blue for males); (b) right: a screenshot of two 3D extreme cases on the second pc (landmark 2 at the bottom and blue for females); (c): screenshots of two 3D extreme cases on the second and fourth principal component (landmark 2 on the right and blue for females).




(a)



(b)



(c)

FIGURE 4.15: (a): score plots of the significant principal components after Procrustes matching and PCA, specified in gender; (b) screenshots of two 3D extreme cases on the second and third pc of the whole mandible (landmark 2 at the back and blue for females); (c) screenshots of two 3D extreme cases on the fourth and tenth pc of the whole mandible (landmark 2 at the back and blue for males).

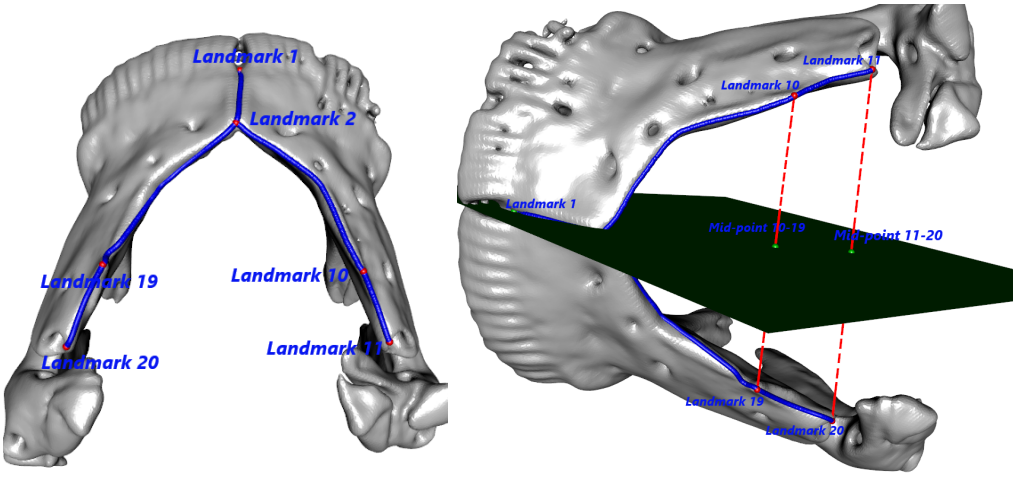
### 4.2.2 Asymmetry and allometry

After identifying the five pairs of anatomical curves, a common step in biology is to remove the asymmetry of each mandible. As suggested by Klingenberg [57], correcting for asymmetry by making symmetrical models is better than just ignoring the concept of symmetry/asymmetry altogether as some studies do, unless asymmetry is the sole purpose of the study. The programme *MorphoJ*, commonly used by evolutionary biologists, will symmetrise data and perform Procrustes matching. Instead of analysing landmarks, this thesis studies the estimated curves. To test the viability of this asymmetry-free procedure for curves, the asymmetry level has been calculated by the method introduced in Bock and Bowman [58]. It is to reflect the shape in any plane, Procrustes match the shape with its reflection and average their point-by-point Euclidean distances by the total number of points on the shape. On average, the asymmetry level is less than 5% of the total size of the jaw, which is small enough to be removed.

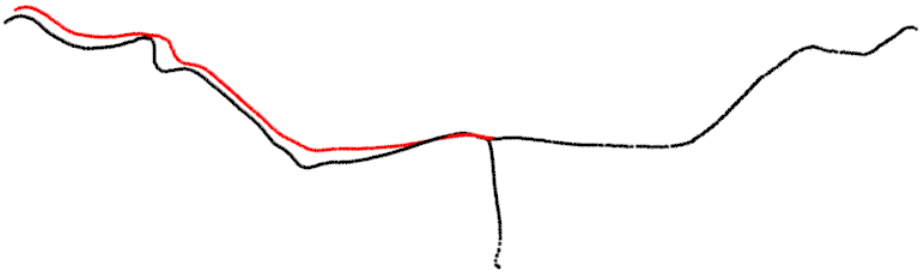
A standard approach for bilaterally symmetric forms in biology is to reflect one side of the asymmetrical object across the mid-line and average the coordinates [59]. The novel approach is to deal with 3D shapes which are shown in Figure 4.16. The blue line in (a) or the black line in (b) is a 3D model of the front mandible shape, with the same set of landmarks in Section 4.2.1. The procedure is as follows.

- Instead of a mid-line, a mid-plane is expanded by landmark 1 and the two mid-points of the pair of landmark 10 and landmark 19 and the pair of landmark 11 and landmark 20, which are shown by three green dots;
- This mid-plane is used to create the mirror reflection of the right bilaterally symmetric shape. The reflection is connected with the original right part to be a new model;
- The new model is Procrusts matched with the original model. The Procrustes mean shape is used as the model with the asymmetry removed,

which is shown by the red line in (b). In this example, the asymmetry level is roughly 1% per point of the total size of the jaw.




(a)



(b)

FIGURE 4.16: (a) left: an example of upper mandible curves as introduced in Figure 4.14; right: A mid-plane of the curves; (b): the comparison between the mirror reflection of the right curves (red) and the true left curves (black).

Normally, it works with the full complement of landmarks or curves. However, it is very common that samples have pieces broken off when collected [59]. For the data in this project, there are 123 complete jaws and 54 broken ones. So before removing the asymmetry, according to Zelditch et al. [60] and Klingenberg

et al. [61], the broken parts have been filled in by the mirror reflection of the corresponding unbroken parts.

Another aspect which should be removed is allometry. It can be considered as a part of the shape information which is related to growth. In other words, the shape difference between young and old fish is not of interest in this study. It can be removed by the regression of Procrustes matched data on the centroid sizes of the unmatched data [62]. Then the residuals which are asymmetry and allometry free would be used for further analysis with genetic information in biology.

## 4.3 Alignment of the curves

The shape of human faces has been studied for hundreds of years. Nowadays, taking advantage of the development of modern imaging equipment, statistical data analysis has been widely available based on three-dimensional data of the surface constructed by a stereo camera system. Vittert et al. [7] developed a methodology using the principle features (3D curves) to model facial data, with details in Vittert [8]. Figure 4.17 left shows an example model of the anatomical facial curves using 23 anatomical landmarks which are defined by Katina et al. [26].

The model consisting of identified 3D facial curves can be used in a variety of ways. For example, particular anatomical curves (such as a lip boundary) may be compared and involved in the evaluation of pre- and post-surgical results. The comparison of the curves can be achieved by Procrustes matching and alignment. This section mainly discusses the alignment methods, with the applications on mid-line nasal curves starting from the midpoint between the eyebrows and ending at the top of the upper lip on a human face, as shown in Figure 4.17 right. The data to align consists of 3D images of 45 people from control subjects.

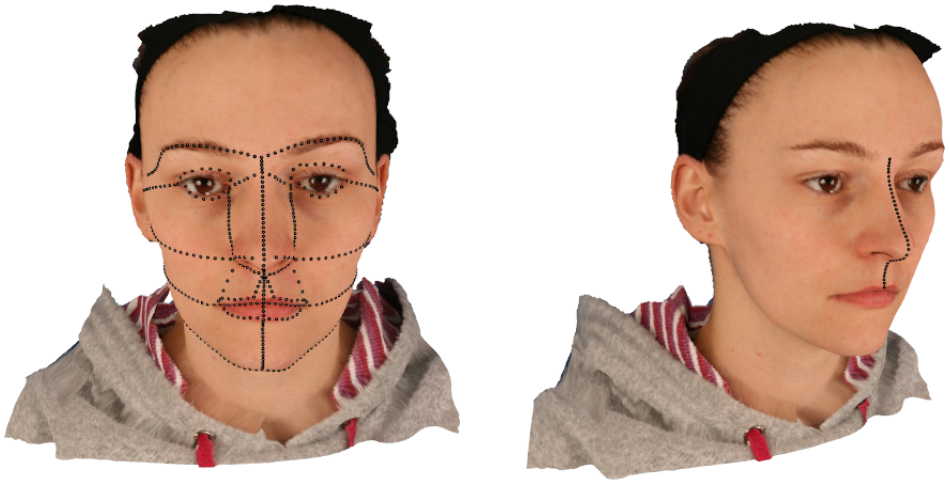


FIGURE 4.17: Left: a 3D image with anatomical curves captured by DI4D® camera system; right: a 3D image with a mid-line nasal curve.

### 4.3.1 Perpendicular plane method

Miller [63] evaluated several methods to analyse 3D curves. The most preferable one is called the Perpendicular plane method, which decomposes the curve by projecting on to 2D planes which are expanded by two pairs of axes. However, instead of using the arbitrary axes generated by the camera systems, three axes which are created by principal component analysis (PCA) can form a more meaningful coordinate system. Note that the use of PCA here is different from the one described in Section 2.2.2 for shape analysis. The aim here is only to transform the axes so that they are on the directions where the data (configuration) varies the most. In other words, in this section, it is applied on three "dimensions" (columns of the coordinate matrix) of one configuration, such as a 3D curve or a set of landmarks on an individual face; while in Section 2.2.2 it is applied on the vectorised (by Equation 2.11) sets of configurations, such as the sets of curves or landmarks on a group of faces.

Thus, the transformed coordinates of the data points on the new axes are the corresponding principal component scores, and form the "score curves" of interest in the following sections. Figure 4.18 shows an example of the transformed data

projected to two 2D planes which are expanded by transformed axes. The first and second principal components, denoted by $y$ and $z$, construct the $y$-$z$ plane which is shown in Figure 4.18 (left). The $y$-$z$ plane reflects the bending of the curves into and out of the face. The first and the third principal components, denoted by $y$ and $x$, expand the x-y plane which is shown in Figure 4.18 (right). The $x$-$y$ plane reflects the traversing over the face. The second and the third components expand the $x$-$z$ plane. The $x$-$z$ plane reflects cuts through the face horizontally which is not of interest, as the curvature information is held mostly in the $y$-$z$ and $x$-$y$ plane. Although the plots in Figure 4.18 look like two continuous lines, they are formed by discrete points, as shown in Figure 4.17 right, with a high density.

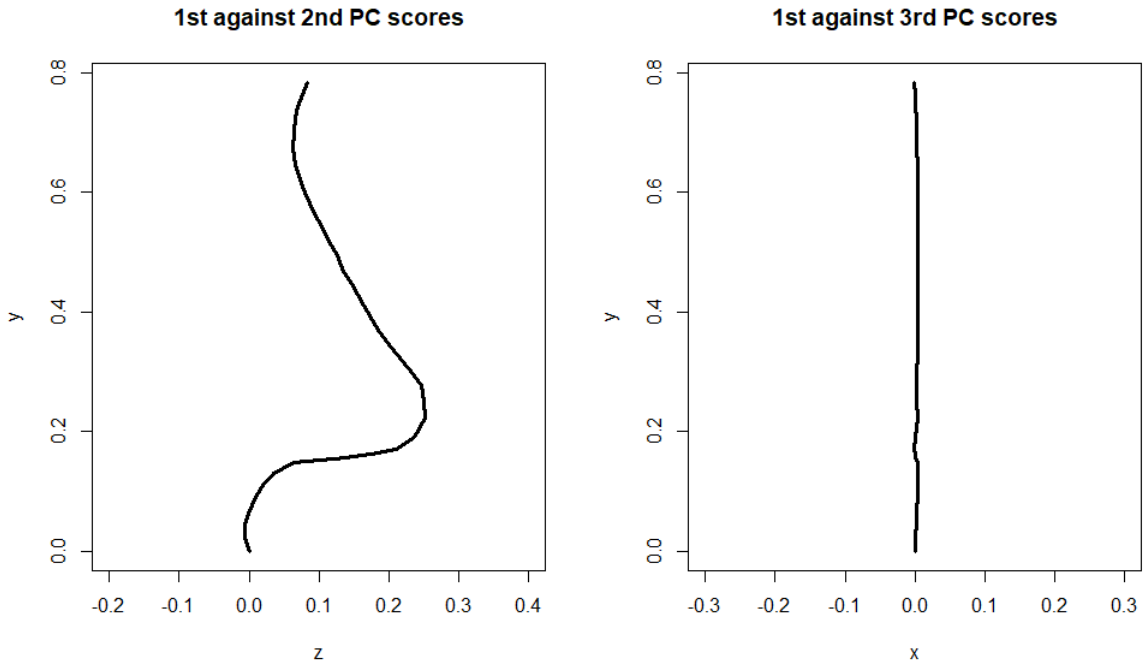



FIGURE 4.18: A example of the plots of the PCA-transfered coordinates (principal component scores) on the $y$-$z$ plane and $x$-$y$ plane

### 4.3.2 Calculating curvatures

The transformed mid-line nasal curves, or score curves, introduced in Section 4.3.1 consist of 45 matrices. Each matrix has three columns for three principal

components which are used as three axes and denoted by $x$, $y$ and $z$. Miller [63] mentioned a method to calculate the curvatures, denoted by $\kappa$, of a planar curve which is shown in Equation 4.2. Because the curvatures form another curve of interest, we refer to the "curvature curves" which need to be distinguished from the score curves. The method is based on the first and second derivatives of the planar curve along its two dimensions as follows.

$$\kappa = \frac{x'y'' - x''y'}{(x'^2 + y'^2)^{\frac{3}{2}}} \tag{4.2}$$

To obtain the derivatives, a score curve which consists of discrete points need to be smoothed by arc length $t$ as mentioned in Section 3.1. Both cubic-spline and local linear smoothing are used to fit the curve. The former has a faster speed and the latter has a more precise (better) fitting. As introduced in Section 2.2.3, the curvature curve will be flatter or smoother if the degree of freedom is smaller and more fluctuate if the degree of freedom is larger.

Figure 4.19 plots the fitted curvatures of the planar curves on the $y$-$z$ plane (blue) and $x$-$y$ plane (red). On the $y$-$z$ plane, the curvature of bending into the face is positive at the point underneath the nose and the curvature of bending out of the face is negative at the tip of the nose. On the $x$-$y$ plane, the curvature of bending to the right-hand side of the face is positive and the curvature of bending to the left-hand side of the face is negative. The curvature curve on $x$-$y$ plane has sharp changes at its minimum and maximum, which results from the sharp bending of the nose curve on $x$-$y$ plane, as shown in Figure 4.18.

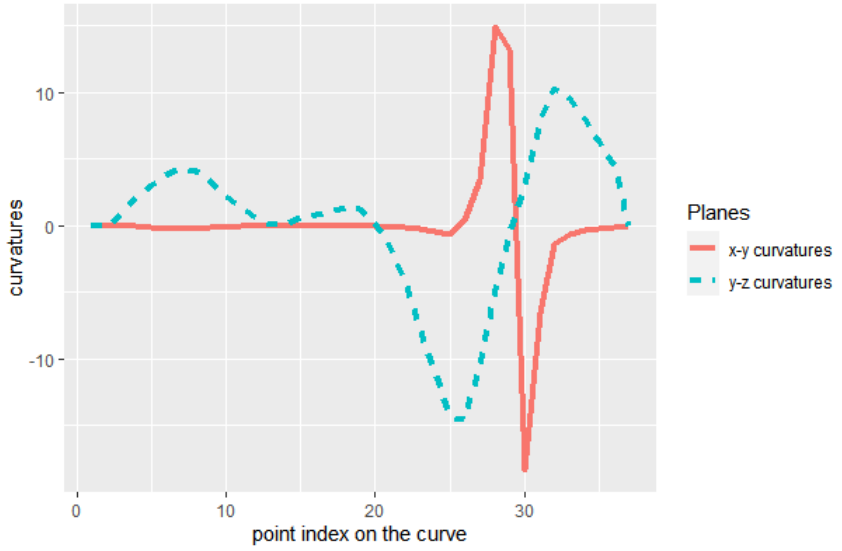



FIGURE 4.19: A plot of the curvature along the projections of a mid-line nasal curve on $y$-$z$ plane (black) and $x$-$y$ plane (red) against its corresponding arc length

### 4.3.3 Warping the curvature curves

In order to compare the curvatures at the same positions on each curve, the curvature curves need to be aligned. In this section, all "curves" will refer to the curvature curves.

A set of curvature curves on $y$-$z$ plane is shown in the first figure in Figure 4.20. Each curvature curve is a function of arc length, so is the variation of the curves. The variation of the curve set is composed of phase and amplitude, which result from different arc length of a point with the same index and the curvature at the point correspondingly. In order to leave alone the amplitude variation of interest, the phase variation needs to be removed from the total variation by alignment, or warping, which are commonly called registration. One of the alignment methods is to align the projected 2D curves.

There are two main methods used to align the curves: landmark and template registration, followed by an application on 45 mid-line nasal curvature curves on $y$-$z$ plane. Another Bayesian method is also briefly introduced.

### Landmark registration

The first method pulls corresponding landmarks on each curve into fixed positions (measured by arc length $t$) along the curve where different amplitudes of the landmarks can be compared. For example, Local maxima and minima can be characteristic points, as suggested in Miller [63]. A monotonous and non-decreasing warping function $g$ can transform these points onto certain positions.

For the mid-line nasal curvature curves, suppose the landmarks are chosen to be the starting point ($t = 0$), the $25^{th}$ (pronasale), the $33^{rd}$ (subnasale) and the ending point ($t = 1$), which will be used to produce warping functions. For the $i$th curve $i$, denote the arc lengths of its point 25 and 33 by $t_1$ and $t_2$ and denote the positions where all the curves will be projected by $\tau_1$ and $\tau_2$. The relationships between the unaligned and aligned positions can be expressed in Equation 4.3.

$$g_i(0) = 0, \quad g_i(t_1) = \tau_1, \quad g_i(t_2) = \tau_2, \quad g_i(1) = 1 \tag{4.3}$$

### Template registration

Landmark registration is for specific points aligned. If the whole curve needs to be aligned, another warping method, template registration is illustrated by Kneip and Ramsay [64]. Template registration uses monotone functions to align the curves by transforming their domain such that every registered curve is equal to the same "template" multiplied by a constant. After the transform, the phase variation, which results from the different locations of the same features, can be removed from the total shape variation.

Each curvature curve, denoted by $x_i$ can be parameterised by the arc length $t$ as a curve function defined on a closed interval. Each curve function is aligned by a warping function denoted by $h_i$ and the aligned curve function is denoted by $y_i$. Denote the mean of unaligned functions by $\mu$ and the mean of aligned functions by $\nu$. Template registration is to register each unaligned function to be a "simple change" of a template, so that the registered function $y = x(h(t)) = \theta(t)\xi(t)$, where $\xi(t)$ is the template function. To avoid the complexity of the "simple change", the

procedure uses a group of template functions $\xi_j, j = 1, \ldots, p$ which are $p$ basic components of variation of the space. Then the registered space can be expressed as

$$\chi_{\xi_1 \ldots \xi_p} = \left\{ x | (x \circ h) = \sum_j^p \theta_j \xi_j, \theta_j \in \mathbb{R} \right\} \tag{4.4}$$

After the phase variation removed by the registration procedure, functional principal component analysis will be applied on the registered functions $y_i$ to study the amplitude variation. Suppose the eigenvalues and corresponding eigenvectors are $\lambda_1 \geqslant \lambda_2 \geqslant \cdots$ and $\zeta_1, \zeta_2, \ldots$Then the principal component score is $c_{ij} = \int [y_i(t) - \nu(t)]\zeta_j(t)dt$. So the functional PCA approximation of the registered curves will be.

$$\nu_i^p(t) = \nu(t) + \sum_j^p c_{ij}\zeta_j(t) \tag{4.5}$$

where $E[\nu_i^p(t)] = \nu(t)$. So the space is also spanned by the first $p$ principal components $\zeta_1, \ldots, \zeta_p$. The registered curves can be expressed as

$$\begin{aligned} y_i(t) = x_i[h_i(t)] &= \nu(t) + \sum_j^p c_{ij}\zeta_j(t) \\ &= E[y_i(t)] + \sum_j^p c_{ij}\zeta_j(t) \\ &= \sum_j^p E(\theta_{ij})\xi_j(t) + \sum_j^p c_{ij}\zeta_j(t) \end{aligned} \tag{4.6}$$

A measure of the independence of the warping function $h$ and the registered curve $y$ is defined in Equation 4.7.

$$C = \frac{E\int x^2(t)dt}{E\int y^2(t)dt} = \frac{E\int Dh(t)y^2(t)dt}{E\int y^2(t)dt} \tag{4.7}$$

The total variance can be decomposed as in Equation 4.8.

$$\begin{aligned} MS_{total} &= E\int [x(t)-\mu(t)]^2 dt \\ &= CE\int y^2(t)dt - \int \mu^2(t)dt \\ &= CE\int [y(t)-\nu(t)]^2 dt + C\int \nu^2(t)dt - \int \mu^2(t)dt \end{aligned} \tag{4.8}$$

Based on Equation 4.5, we have $E\int[\nu_i^p(t)-\nu(t)]^2dt = \sum_j^p \lambda_j$ and $E\int[y_i(t)-\nu_i^p(t)]^2dt = \sum_{j>p}\lambda_j$. Then Equation 4.8 can be expressed as Equation 4.9.

$$\begin{aligned} MS_{total} &= E\int [x(t)-\mu(t)]^2 dt \\ &= CE\int [\nu_i^p(t)-\nu(t)]^2 dt + CE\int [y_i(t)-\nu_i^p(t)]^2 dt + C\int \nu^2(t)dt - \int \mu^2(t)dt \end{aligned} \tag{4.9}$$

So the total variance can be decomposed as in Equation 4.10 to 4.12:

$$MS_{amplitude} = CE\int [\nu_i^p(t)-\nu(t)]^2 dt \tag{4.10}$$

$$MS_{phase} = C\int \nu^2(t)dt - \int \mu^2(t)dt \tag{4.11}$$

$$MS_{residual} = CE\int [y_i(t)-\nu_i^p(t)]^2 dt \tag{4.12}$$

### Application of landmark and template registration

Figure 4.20 shows an example of applying landmark and template registration on the 45 mid-line nasal curvature curves projected on $y$-$z$ plane (shortened by curvature curves). Those unaligned curvature curves are shown on the top left. The landmark warping of the $i$-th curve uses a function $g_i(t)$ of arc length $t$ shown on the top right, and the landmarks are chosen at the maxima and minima of the curvatures (the peak and bottom of the curvature curve). The landmark-registered curves are shown on the bottom left, with landmarks at roughly $t = 0.7$ and $t = 0.9$. The template-registrated curves are shown on the bottom right in Figure 4.20.

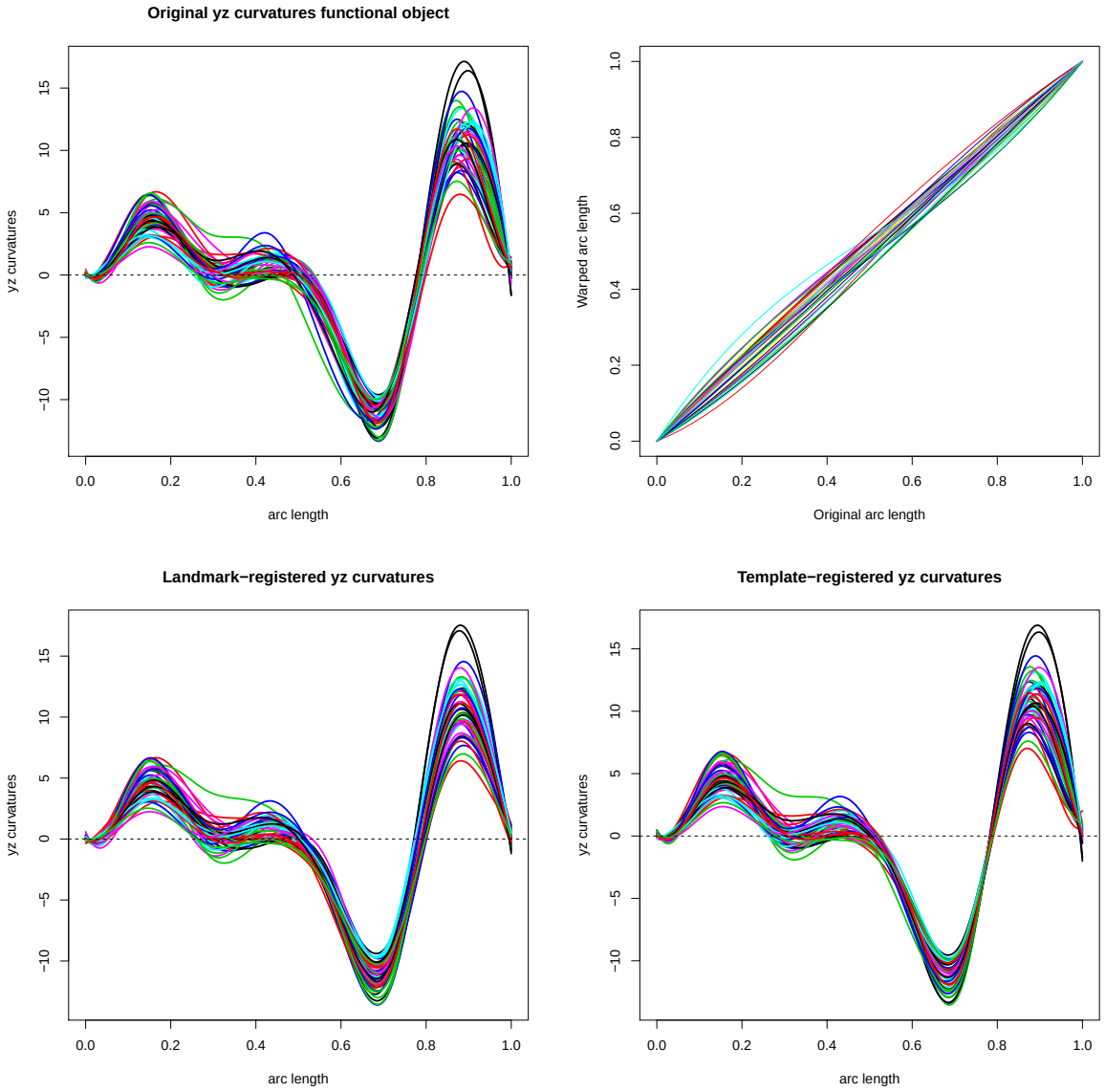



FIGURE 4.20: A curve registration's application on mid-line nasal curvature curves projected on $y$-$z$ plane. Top left: unaligned curvature curves; top right: landmark-registered warping functions $g_i(t)$ against arc length $t$, which register the curvature curves at the minima and maxima of curvatures; bottom left: landmark-registered curvature curves; bottom right: template-registered curvature curves.

It can be seen that the landmark-registered plot has a better warping at two landmarks, while the template-registered plot shows a more concise warping on the whole curves. Furthermore, the phase variation of the unaligned functions can be calculated by the command *AmpPhaseDecomp* in the package "fda" in R or

by the formula below (Kneip and Ramsay [64])

$$\text{Phase variation} = C\int \nu(t)^2 dt - \int \mu(t)^2 dt \tag{4.13}$$

where $\nu(t)$ is the average of the registered curves and $\mu(t)$ is the average of the unregistered curves. $C$ is the coefficient of the relationship between the first derivative of the warping function and the square of the registered functions, which equals to 1 when their relationship is independent (Kneip and Ramsay [64]). However, it seems that this relationship should not be independent, which may be investigated in future works.

For the mid-line nasal example, the phase variation calculated after template registration is 20%. However, the phase variation after landmark registration is negative, which indicates that this registration has increased the total variation. It is not difficult to understand that the reason is aligning several points sacrificed the positions of the rest.

**Bayesian Registration**

Another popular method in functional data analysis to register the curves is building Bayesian models. For example, Telesca and Inoue [65] built a flexible Bayesian hierarchical model to register the curves. It has an advantage that it does not need to smooth the data preliminarily. With the nature of a statistical model, it can provide the assessment of the uncertainty of the shape variation after registration. Cheng et al. [66] introduced a method to register 1D and 2D curves, but can also be extended to 3D. This section provides a brief introduction to this method. The investigation of this method may be in future works.

Denote the unregistered curve by $f$ and its warping function by $\gamma$, which are functions of the arc length $t$. Instead of registering $f$s, this method transfers them by a function and builds Bayesian models to register the transfered curves. This function is called the Square Root Velocity Function (SRVF) which remains the

variations unchanged. It is defined as a function $q$ of $f$

$$q(t) = \frac{\dot{f}(t)}{\sqrt{\|\dot{f}(t)\|}}$$

The SRVF of the registered function $f \circ \gamma$ is $q^*(t) = \sqrt{\dot{\gamma}(t)}q(\gamma(t))$. For the sample of two unregistered curves $f_1$ and $f_2$, their SRVF $q_1$ and $q_2$ are used for Bayesian registration. It can be used to calculate the elastic metric $d(q_1, q_2) = d_{Elastic}(f_1, f_2)$, which obeys the property $d_{Elastic}(f_1 \circ \gamma, f_2 \circ \gamma) = d_{Elastic}(f_1, f_2)$. This property ensures that the distance between the two target functions is unchanged if both transformed by a common warping function. Further details can be found in Section 4.3 in Cheng et al. [66].

## 4.4 Curve reconstruction

Shape reconstruction is another interesting topic of the identified 3D curves, which is to translate the curvature analysis back to the 3D space. Following the widely-used off-line 3D scanning technology, online 3D reconstruction is still under development, as mentioned in Keller et al. [67]. Online 3D reconstruction is capable of producing real-time 3D imaging by scanning using 3D cameras, such as the immediate visual rendering from the scanner. The most well-known application is augmented reality (AR) which fuses the real-world shapes from 3D images and displays to the user. The reconstruction procedure consists of: data association, which is to find the corresponding point of the real-world object on the reconstructed shape; minimization of alignment error, which is to reduce the deviance between the paired points. The challenge is that the camera may imprecisely estimate sufficiently salient local features during the reconstruction. Some researches attempted to tackle this problem. For example, Lefloch et al. [68] used the maximum and minimum principal curvature (introduced in Section 2.2) to fit a quadratic model of the shape to enhance the reconstruction performance.

Using curvature information is not an uncommon way to reconstruct the shape. Wook et al. [69] used the second derivatives which are derived from Zernike polynomials. Here we adopt the method introduced in Miller [63] to implement curve reconstruction based on the principal curvatures. The combination of both curves and their curvature information would benefit the reconstruction procedure of data association and error minimisation.

This section discusses the procedure which is mentioned in Miller [63] to reconstruct shapes based on their curvature, including both planar curves and 3D space curves. A real dataset of mid-line nasal curves in Section 4.3 will also be used here as an application. In the end, a simulation study is performed to investigate the effect of noise in the data on the reconstruction procedure.

### 4.4.1 Reconstruction of 2D curves

As mentioned in Miller [63], the curvature can be regarded as the changing rate of turning angles at any point of the curve. Thus, the integration of the curvatures gives the angles of all the points on the curve. Furthermore, in terms of data association, the coordinates can be gained from the cosine and sine transformation of the angles. This reconstruction procedure of planar score curves can be expressed by Equation 4.14 and 4.15, as illustrated in Miller [63].

$$r(t) = \left( \int_0^t \cos(\phi(u))du + a, \int_0^t \sin(\phi(u))du + b \right) \tag{4.14}$$

$$\phi(t) = \int_0^t \kappa(u)du + \phi(0) \tag{4.15}$$

Here $r(t)$ is the coordinates of any point on the curve, as a function of the arc length $t$; $\phi$ is the angle of the point; $\kappa$ is the curvatures as a function of the arc length; $(a, b)$ and $\phi(0)$ are the initial coordinates and angle of the reconstruction. In fact, when the reconstructed curve and the original curve are matched by Procrustes methods,

of which moving, scaling and rotating are used for transformation (Dryden and Mardia [29]), the coordinate of the starting point, $(a, b)$ will be of little use.

However, the roughness or alternatively, smoothness of the curvatures will influence the reconstruction with the change of the degree of freedom when fitting the score curve. A smoother curvature curve will lack detailed information to reconstruct the scores. Figure 4.21 left shows an example of the influence of degree of freedom on the reconstruction deviance on the $y$-$z$ plane (black; example in Figure 4.18 left) and $x$-$y$ plane (red; example in Figure 4.18 right). During the procedure of selecting the best degree of freedom, the cubic-spline smoothing is chosen because of its fast speed and little fitting deviance from local linear smoothing. The best degree of freedom can be 10 for the score curve on the $y$-$z$ plane, but it is more complicated for the score curve on the $x$-$y$ plane.

With a total of 37 observations, the degree of freedom about 17 gives the smallest reconstruction deviation with the danger of over-fitting. The reason can be found if we plot the reconstruction of the score curve on $x$-$y$ plane, as shown in Figure 4.21 right. The original score curve on the $x$-$y$ plane is plotted by a black line, where there is a small but sharp change of $x$ within the range of $-0.25 < y < -0.1$. This sharp change results in a sharp change of the curvature curve within that range, with an example of the red line in Figure 4.19. On the right of Figure 4.21, a smaller degree of freedom (green and blue lines) tends to produce a reconstruction with little influence of the details ($-0.25 < y < -0.1$), while a larger degree of freedom (red line) tends to produce a reconstruction from every detail of the curvature curve so that increases the deviance of the total reconstruction in the range of $-0.25 < y < -0.1$. This large degree of freedom issue will be investigated further in a simulation study in Section 4.4.2.

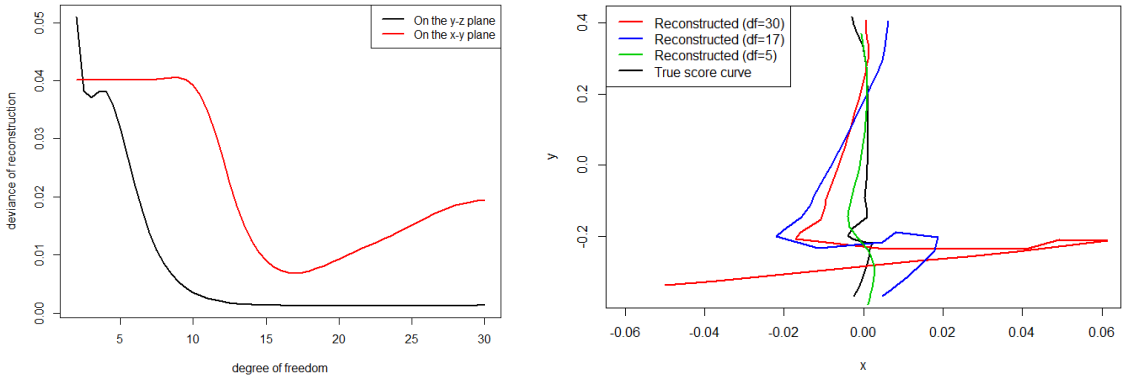



FIGURE 4.21: Left: the degree of freedom of penalty-spline smoothing when fitting the score curves on two planes and calculating the curvature curve, against Procruste matched deviance of the reconstruction; Right: the effect of degree of freedom when reconstructing the score curve on the $x$-$y$ plane.

In this case, without too large deviance, we can choose 5 degrees of freedom for the score curve reconstruction on the $x$-$y$ plane. As shown in Figure 4.22, the reconstructed score curves (red) and the original curve (black) are matched using Ordinary Procrustes Analysis (OPA). The deviances of the reconstruction per point are 0.86% and 3% on $y$-$z$ plane and $x$-$y$ plane correspondingly.

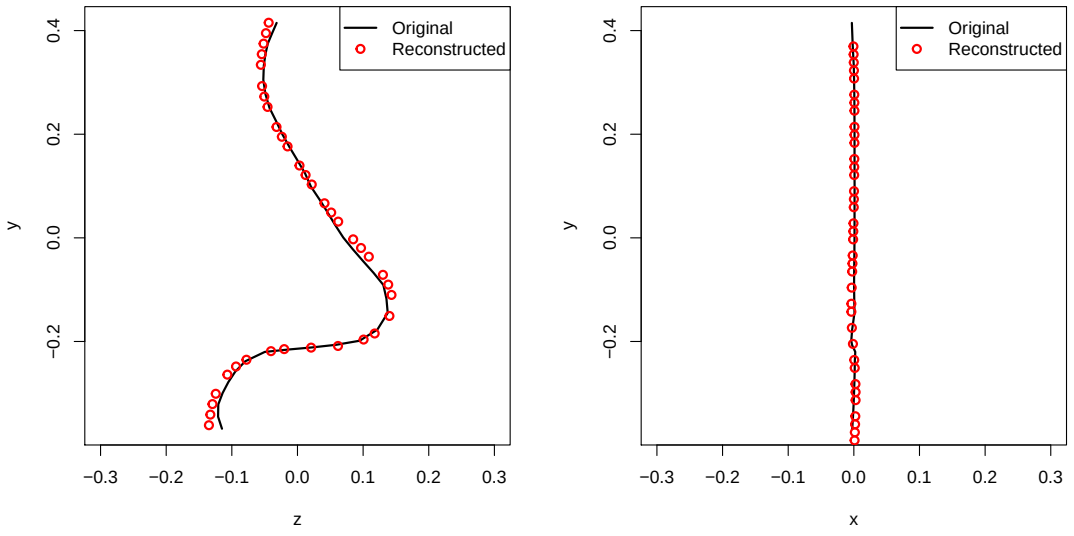



FIGURE 4.22: Procruste matched the original score curves (black lines) and the reconstructed score curves (red dots) with degree of freedom chosen to be 10 on the $y$-$z$ plane and 5 on the $x$-$y$ plane.

### 4.4.2 Reconstruction of 3D curves and a simulation study

After reconstructing the curves on the y-z and x-y planes, the space curve can be reconstructed back to the 3D space, as illustrated by Miller [63]. However, to find the optimal initial angles $\phi_0$ (in Equation 4.15) of the two planes, instead of minimizing the differences between y coordinates produced by both planes, the general approach is to minimise the Euclidean distance between the reconstructed and original 3-dimensional curves when the latter is known. Using the same degree of freedoms in Figure 4.22 and the optimal initial angles 18° ($y$-$z$) and 90° ($x$-$y$), the reconstruction of the 3D score curve is shown in Figure 4.23.

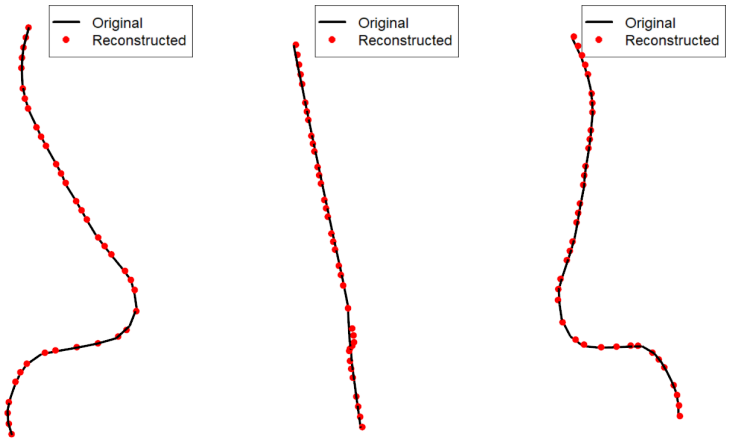



FIGURE 4.23: Screen shots of Procrustes matched original 3D score curve and the reconstructed 3D score curve in the right-side (left), left-side (right) and front (middle) views.

When calculating derivatives in Section 4.3.2 and 4.4.1, we assume the two dimensions of a planar curve share the same degree of freedom. However, it can be reasonable that the three axes have different degrees of freedom in smoothing. When the original 3D score curve is known, the optimal degree of freedoms of three dimensions can be found directly by minimising the distance between the reconstructed and the original curves. In this way, we found most of the optimal degrees of freedom are quite large. For example, the optimal degrees of freedom of a mid-line nose data are 23, 19, 13 for $y, z, x$ axes correspondingly.

This requirement of high degrees of freedom may result from the noise in real data so that the fitting paid too much attention to the noise. For example, the noise may come from the camera system when collecting the data, or the identification procedure of the curves. We investigate in a simulation study in the following steps to test the effect of noise on this space-curve reconstruction procedure.

Step 1: Create a 3D score curve;

Step 2: Add three sets of random noise to three dimensions on the score curve to simulate a noisy score curve;

Step 3: Calculate the curvatures of the simulated noisy curve and reconstruct based on those curvatures;

Step 4: Calculate the deviance between the reconstructed score curve and the score curve without noise;

Step 5: Repeat from Step 2 for required times;

Step 6: Take the average of the deviances per point between the reconstructed score curve and the score curve without noise. Calculate the percentage of the deviance mean with respect to the total arc length.

The data created for this simulation study were designed to be similar to the midline nasal score curves. Each dimension is a second-order differentiable function of the arc length $t$ as follows so that we can easily obtain their first and second derivatives to calculate curvatures. This procedure is very time-consuming with integration used in Equation 4.14 and 4.15. One possibility is to approximate the integration by Riemann Integration. The speed has been improved from hours to 20 seconds for a single simulation.

$$x = 0.004 \sin(10t) \tag{4.16}$$

$$y = 0.8(1 - t) \tag{4.17}$$

$$z = \frac{1}{4} \exp\left\{ -\frac{(t - 0.5)^2}{2 \times \left(\frac{1}{7}\right)^2} \right\} \tag{4.18}$$

The noise added on each dimension is from a standard normal distribution with zero means. Figure 4.24 shows an example of the added noise in three different views. The black line is the original simulated 3D score curve, the red line added noise with standard deviation sd $= 0.01$ on $x$, $y$ and $z$ and the blue line is the reconstructed score curve from the red line.

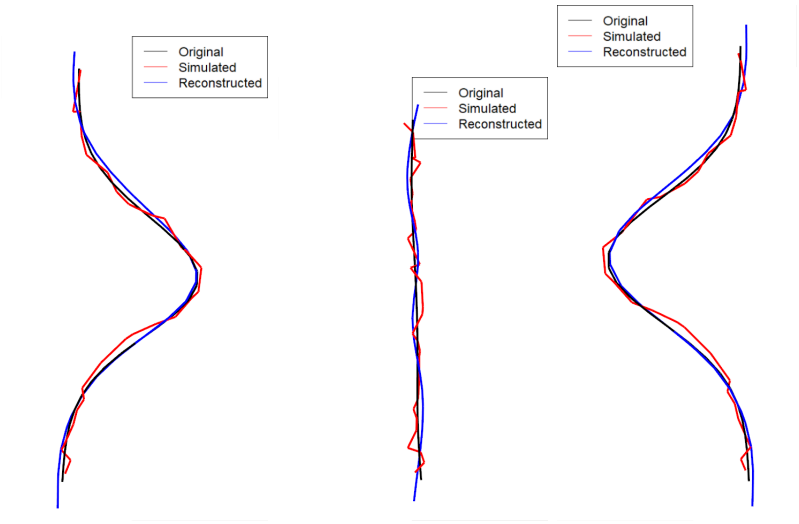



FIGURE 4.24: Screen shots of Procrustes matched original 3D score curve (black) and the reconstructed 3D score curve (blue) in the right-side (left), left-side (right) and front (middle) views. The red line shows the original 3D score curve with simulated noise.

As discussed in Section 4.3.2 and 4.4.1, since the dataset consists of groups of discrete points, the analysis should be based on models, where issues of choosing smoothing methods and degrees of freedom are involved. Cubic-spline smoothing with penalty is much quicker which will be used in this simulation study.

Figure 4.25 on the left shows the true score curve (black line) and the noisy score curve (red line) with the simulated noise (a standard normal distribution with sd = 0.01). In fact, the fluctuation of the noisy curve is rather small compared to the total arc length 1 of the 3D true score curve. The right figure shows the true curvature curve (black line) and the fitted curvature curves (other colours) with different degrees of freedom. Because of the noise, a larger degree of freedom makes the fit unrealistically fluctuate and a smaller one seems to fit much closer to the true curvature curve (black line on the right). Figure 4.26

This can explain the issue raised in Section 4.4.1 or the one shown in Figure 4.21, where higher degree of freedom is required for the best reconstruction. The noise increased the curvatures which misled the reconstruction. In this circumstance, the choice of degree of freedom should be based on the expectation of the shape.

For example, Figure 4.26 on the left shows the curvatures on $y$-$z$ plane. The best fit is still under a large degree of freedom (9) even though there is no noise added on this plane. This is because the shape of this curvature curve has a large "valley" in the middle and it requires a large degree of freedom to fit to the small values. Thus, we can conclude as follows.

- If there are few sharp small changes of the curvatures but no large one, choose a small degree of freedom (less than 5);
- If there exsits a large change of the curvatures, choose a slightly larger degree of freedom (about 10).

Figure 4.26 on the right shows the true 3D score curve (black lines) and the reconstructed curve (red dots) which are based on the noisy score curve (red line) with the simulated noise (standard normal distribution with sd = 0.01). The reconstruction deviation per point is about 1.5% of the total arc length, which seems to be consistent with the standard deviation of added noise.

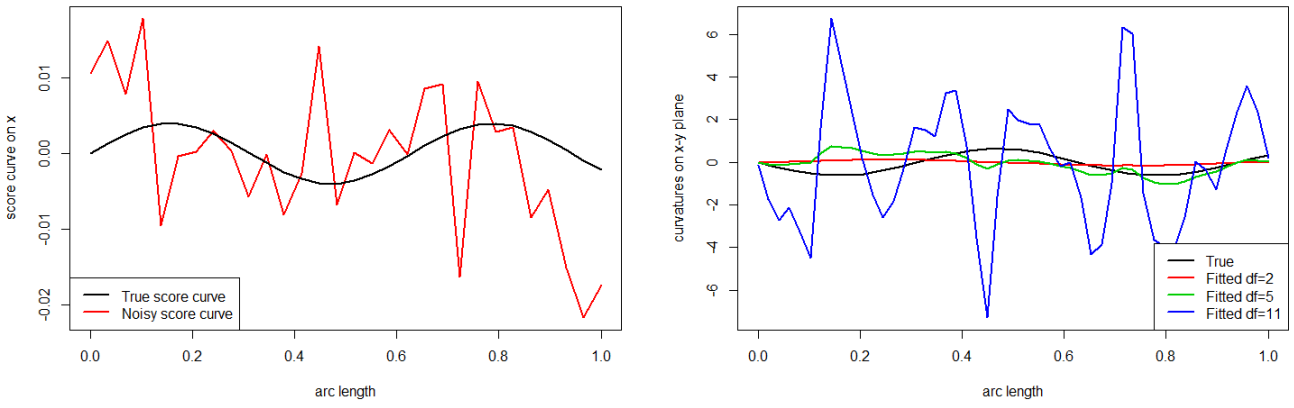



FIGURE 4.25: Left: noisy score curve on $x$ (red) which is the true score curve (black) on $x$ with added noise (sd = 0.01); Right: the effect of degree of freedom on the fitted curvature curves on $x$.

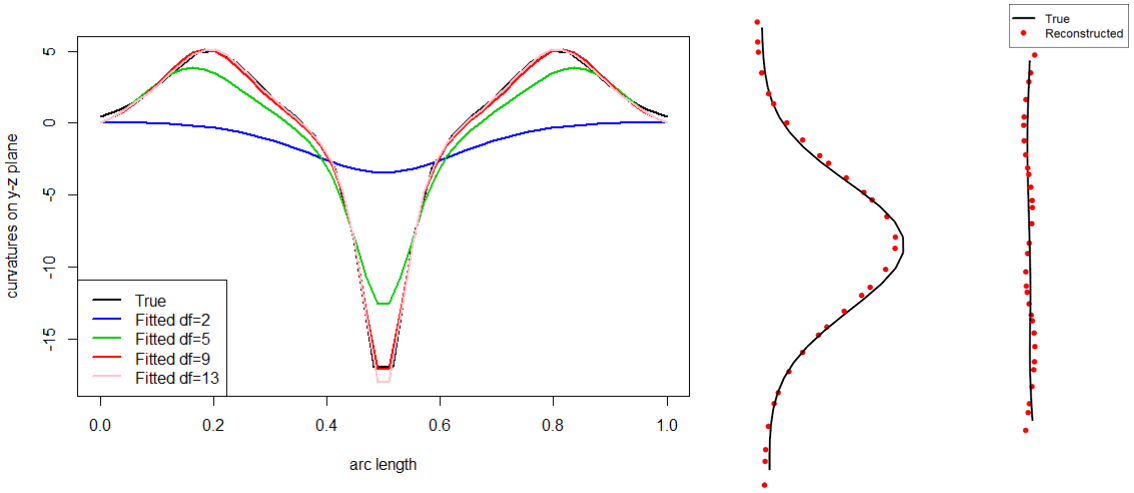



FIGURE 4.26: Left: true curvature curve (black) on the $y$-$z$ plane and fitted curvature curves (other colours) with different degrees of freedom; Right: true 3D score curve (black lines) and the reconstructed (red dots) from curvatures which are fitted with degrees of freedom 1,6,5 on $y, z, x$ correspondingly

### 4.4.3 Summary of curve reconstruction

This section focuses on the reconstruction of both 2D and 3D curves based on their curvatures. The method is following Equation 4.14 and 4.15, where the integrals are Riemann-approximated. Because the curves consist of discrete points, the calculation involves natural spline smoothing with a penalty. The smoothness of the fitting is key to the reconstruction, which is investigated in a simulation study. Data with different size of noise are used to test the power of the reconstruction method under the pressure of noise which may be created during the data collection.

The study indicates that the choice of fitting smoothness depends on the shape of the curve to reconstruct. A curve with smaller fluctuation requires larger smoothness, even if there are sharp changes. One with strong bendings in the middle requires larger roughness, but the smoothness with the degree of freedom 9 for a total of 37 points is adequate for curves like the mid-line nasal profile.

# Chapter 5

# Smile Train Project

An estimated 170,000 children are born in the developing world each year with cleft lip and palate, a surgically treatable facial deformity that impairs children's health, well-being and survival. Some initiatives invite experienced doctors to go to the developing world to undertake surgery. In contrast, Smile Train seeks to train local doctors in professional facial disease surgeries, such as cleft lip and palate. This training project needs to be assessed and feedback given on the practice of surgeons. The intention is to improve the quality of data available by employing 3D rather than 2D imaging, and hence to contribute to a step-change in the support for surgical staff and Smile Train, a charity which supports children suffering from facial defects, with a consequent significant benefit for patients.

To collect the data for this project, various 3D scanners have been considered. However, due to some disadvantages of certain types, such as high cost, poor mobility and strong flashing light that is uncomfortable for some patients, a new camera system is under development in cooperation with NCTech® Ltd, Edinburgh. Validation studies have been conducted to compare the accuracy of estimating shape position with other scanner systems. So far, the new camera has been taken to Brazil data collection stations, though the camera system is still under development.

This camera system under development composed of two Intel™RealSense D415 sensors attached to a Windows tablet with the calibration algorithm written by NCTech® Ltd. Those sensors are non-contact active structured-light 3D scanners. As shown in Figure 5.1, they are stored in a black frame and connected to the tablet by a cable. Another two non-contact scanners involved are quite expensive but with high accuracy: a passive stereo camera systems produced by DI4D® and hand-held active structured-light scanners, named "Eva" and produced by Artec®. In Figure 5.2, the DI4D® camera system is shown on the left with two studio lights by its sides and "Eva" is shown on the right. Both of them are mature products and have been already used in industries and researches. Thus, we regard them as the "ground truth" to test the performance of the new camera system at each developing stages.

FIGURE 5.1: The new camera system is composed of two Intel™RealSense D415 sensors attached to a Windows tablet.

FIGURE 5.2: Left: DI4D® camera system with two studio lights; Right: The scanner, "Eva" produced by Artec®.

They also have some disadvantages as the former is not convenient to transport and the latter has flashing lights which may be uncomfortable for young children. Katina et al. [26] described a validation study for a new definition of a set of facial landmarks. It tested the variation of landmark coordinates considering the effects of the observer, subjects, days and image capture repeats.

In this chapter, the validation studies, which are quantitative assessments of each stage of the development of the new camera system, will not only investigate the landmarks, but also curves on the performance of machines. Note that some long tables in this chapter are in Appendix A. The project is still at its early stage of data collection. Further work will focus on statistical modelling for the assessment of the surgeries.

## 5.1 Validation study 1: an early stage

### 5.1.1 Introduction

This validation study benchmarks the performance of NCTech® and Eva (produced by Artec®), based on both landmarks (manually placed) and curves (estimated by methods in Chapter 3). There are 23 3D images in total which are captured from four participants in this study. Figure 5.3 shows an example of two images from Eva (1st and 3rd) and NCTech® (2nd and 4th) in the front and side view. At this early stage, the new camera system cannot build a triangulation for the captured point cloud, so the triangulation is created using MATLAB™. It can be noticed that, at this stage, compared with left figures captured by Eva, the surface is not smooth enough and has some "orange peel" effects.

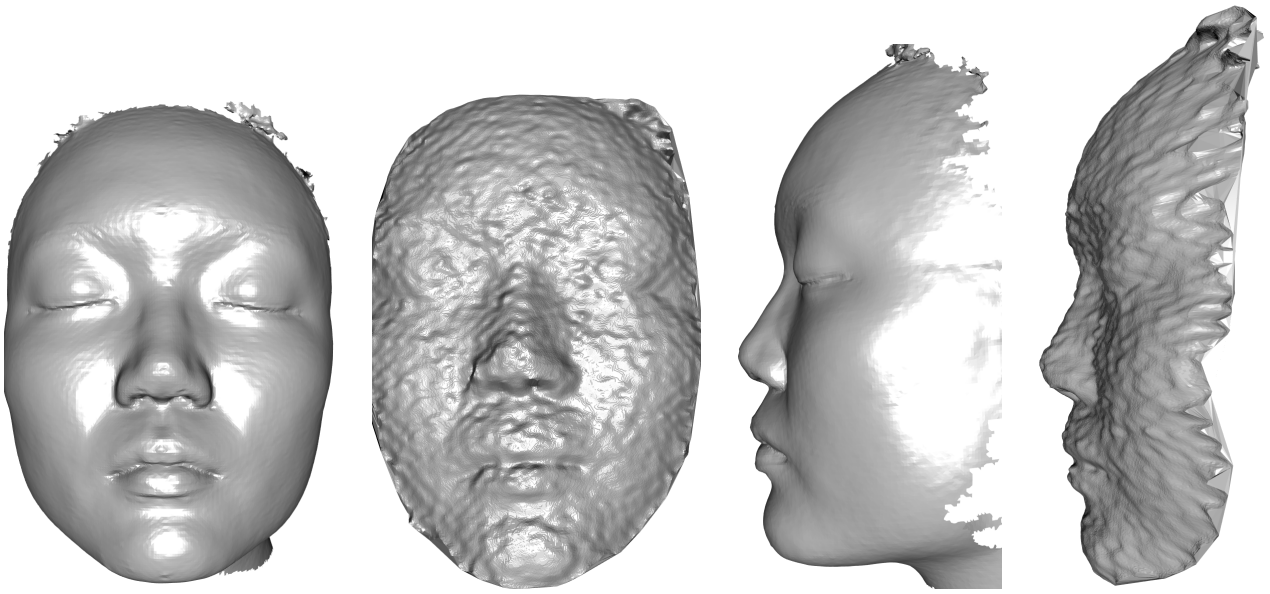


FIGURE 5.3: Screen shots of the whole surface; the first two are in the front view and the last two are in the side view.

The configurations (a set of landmarks or curves) on the images from the same participant which are captured by the same machine are Procrustes matched[1] so that there are eight average configurations from four participants.

### 5.1.2 Comparison on landmarks

There are 13 landmarks manually placed on the 3D images by using the *Landmark* software and the *CloudCompare* software for Eva and NCTech® images correspondingly. Figure 5.4 shows an example image from Eva with landmarks (red points) on the left and landmarks labelled by their orders on the right.

[1]Procrustes registration is applied to matching the set of landmarks of each individual. For example, the difference between the two sets of landmarks consists of position and rotation when they are in the same units. In order to compare the estimates of the same landmark from different cameras, Procrustes superimposition can be a popular tool to remove the location and rotational effects, which are not of interest, of the coordinate estimates. Further details can be found in Section 2.2.2

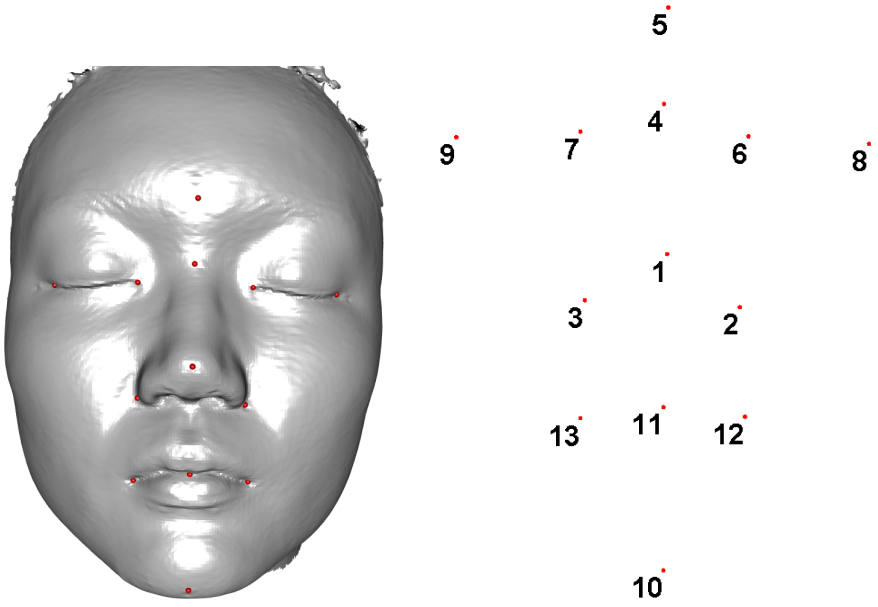



FIGURE 5.4: An example of the landmarks (red dots) manually placed on the image. The landmarks are ordered from 1 to 13.

Table 5.1.2 summarises the comparison of Procrustes-matched landmark positions from two machines for each participant. For each participant, OSS $= \sum_{i=1}^{n} d_i^2$ is the ordinary Procrustes sum of squares of two average configurations (two sets of $n$ landmarks) from two machines, where $d_i$ is the Euclidean distance between any pair of the same landmark captured by two machines; "rmsd" is the square root of the mean OSS with averaging on landmarks. Both OSS and rmsd are statistics to measure the deviation, where the former measures the total squared and latter measures the average deviation per landmark of the images produced by two camera systems.

$$\text{rmsd} = \sqrt{\frac{\text{OSS}}{n}} = \sqrt{\frac{\sum_{i=1}^{n} d_i^2}{n}} \tag{5.1}$$

The table shows that in most cases the average deviations per landmark are quite small (rmsd<3mm); Some of them are larger than 5 millimetres. The performance of the new camera system seems good at this early stage, because large deviations may come from the operator error to place the landmarks, especially at some ambiguous facial area. For example, it may be difficult to manually place a consistent position across the faces at the inner corner of the eyes (the 6th and 7th landmark)

or the corner of the lip (the 13th landmark). In addition, there may also exist a participant error, as the OSS of the second and fourth participant are quite small compared with the other two.

| **participant** | **1** | **2** | **3** | **4** | **sd** |
|---|---|---|---|---|---|
| **OSS** | 191.33 | 86.23 | 284.48 | 49.35 | |
| **rmsd** | 3.84 | 2.58 | 4.68 | 1.95 | 1.23 |
| lmk 1 | 2.05 | 2.09 | 2.89 | 0.72 | 0.90 |
| lmk 2 | 4.21 | 2.33 | 3.22 | 1.78 | 1.06 |
| lmk 3 | 3.48 | 2.20 | 4.36 | 2.25 | 1.04 |
| lmk 4 | 1.65 | 2.79 | 1.42 | 0.39 | 0.98 |
| lmk 5 | 1.57 | 2.52 | 2.14 | 4.21 | 1.14 |
| lmk 6 | 7.66 | 3.99 | 9.18 | 1.35 | 3.55 |
| lmk 7 | 6.38 | 3.49 | 5.75 | 1.73 | 2.14 |
| lmk 8 | 3.05 | 1.91 | 1.12 | 1.15 | 0.91 |
| lmk 9 | 3.90 | 2.30 | 1.32 | 2.45 | 1.06 |
| lmk 10 | 2.94 | 3.12 | 5.08 | 2.11 | 1.26 |
| lmk 11 | 2.21 | 0.86 | 3.27 | 1.74 | 1.00 |
| lmk 12 | 0.67 | 2.60 | 6.88 | 0.53 | 2.96 |
| lmk 13 | 3.77 | 1.79 | 6.00 | 1.68 | 2.03 |

TABLE 5.1: Differences of the landmarks from two machines (unit: millimetre)

### 5.1.3 Comparison on middle lip curves

A valley curve was created in the middle of the lip on each 3D image. The curve consists of two "plane cuts" (in Section 3.2.1) between the 11th and 12th landmark; the 11th and 13th landmark. Figure 5.5 and 5.6 show the lip curves (red) on the surfaces, as well as the 11th, 12th and 13th landmark (black points), in the front and side views. Each of the identified curve pair was resampled to have 100 points in order to be comparable to each other. The resampling method is spline smoothing of the coordinates. Although it cannot ensure every resampled

point is actually on the surface, the high density of the curve provides a good approximation.

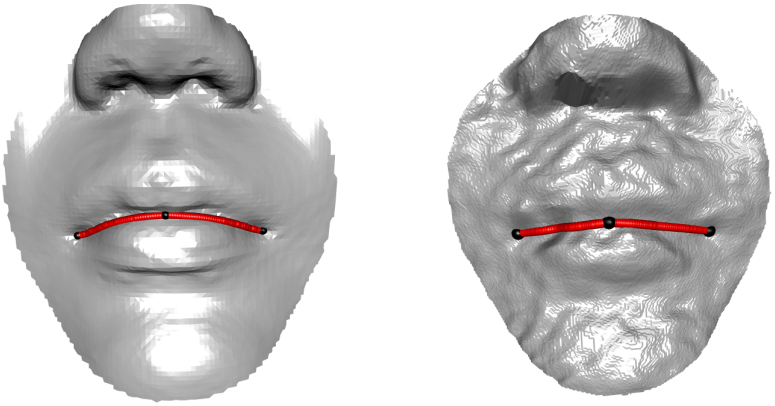


FIGURE 5.5: Lip curves (red points) on the images produced by Eva on the left and the new system on the right with landmarks (black points) in the front view.

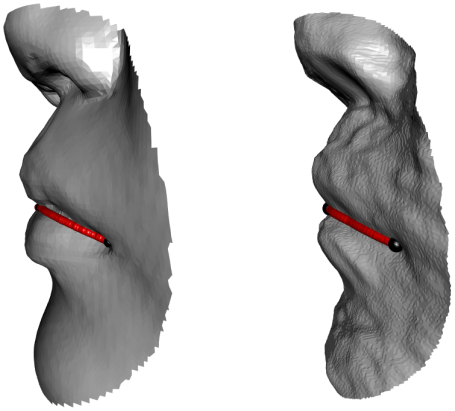


FIGURE 5.6: Lip curves (red points) on the images produced by Eva on the left and the new system on the right with landmarks (black points) in the side view.

Figure 5.7 displays two lip curves without the surface of the participant's images from two machines (Eva: black; NCTech®: red). The OSS and rmsd of each participant are summarised in Table 5.1.3 where rmsd is on average of points on the curves. Besides, the time (in seconds) to create a lip curve on the same participant's lip is shown in Table 5.1.3. Attention should be focused on "User"

[2]. The slow speed to identify the curve for the new system may result from the algorithm to build a triangulation for its output. Eva's quicker performance seems to benefit from its built-in triangulation which uses the smallest number of triangles to connect the captured data points.

At this early stage, the performance of the new camera system performs reasonally good in identifying curves. Comparing Table 5.1.2 and 5.1.3, the OSS does not change dramatically when the number of points increase from 13 to almost a hundred, and the average deviation per point on the curve seems small.

FIGURE 5.7: Comparison of two lip curves for the same surface. Artec® (black) and NCTech® (red)

| **participant** | **1** | **2** | **3** | **4** |
|---|---|---|---|---|
| **OSS** | 148.64 | 261.72 | 1776.95 | 66.68 |
| **rmsd** | 1.22 | 1.62 | 4.22 | 0.82 |

TABLE 5.2: Differences of the lip curves from two machines (unit: millimetre)

| | **User** | **System** | **Elapsed** |
|---|---|---|---|
| **Artec®** | 7.36 | 0.00 | 7.39 |
| **NCTech®** | 85.51 | 0.00 | 85.58 |

TABLE 5.3: Time to create the lip curves from two machines (in seconds)

[2]The first two entries ("User" and "System") are the total user and system CPU times of the current R process and any child processes on which it has waited, and the third entry ("Elapsed") is the "real" elapsed time since the process was started. The "user time" is the CPU time charged for the execution of user instructions of the calling process. The "system time" is the CPU time charged for execution by the system on behalf of the calling process.

## 5.2 Validation study 2: an intermediate stage

### 5.2.1 Introduction

This validation study benchmarks the performance of NCTech®, Artec® and DI4D® machines together, based on a set of landmarks only. In this study, the landmark set on images is also manually placed in the software. The variation of the landmark set on the images produced by different machines can be heavily influenced by the manual placement which is not of interest in comparing the performance of the machines. The advantage of this study is that the landmarks are not only manually placed on the 3D images, but also manually drawn as black dots on participants' faces (with their permission). There are two advantages of this study compared with the one in Section 5.1 as follow.

- The digital position of each landmark is placed according to the colour (black dots) on the image so that the digital landmarks of different images have the same "reference" position. Thus, the variation resulting from placing the digital landmarks can be reduced and the difference of the digital landmarks can represent more efficiently the variation of the performance of different machines;
- Because the digital landmarks are placed with the "same"[3] reference, the digital distances between any pair of digital landmarks can be also compared to reflect the variation of the performance of different machines.

As five of the participants were scanned twice by the three machines and the sixth participant missed Artec® scans for one time, there are 35 outputs (images) in total. At this stage, the system can produce a triangulation by itself for the captured points.

[3]The reason of this quote is that the black dots are still what the machine estimated, but the placement rule is the same: refer to the colour.

### 5.2.2 Comparison on landmarks

There are 19 landmarks manually drawn on each face and 12 of them are clear on all of the images to compare. Because *Landmark* cannot display the texture of the imported images, only *CloudCompare* is used to manually place digital landmarks, according to the positions of the black points which are manually drawn on the faces. Figure 5.8 shows an image from DI4D® displayed in *CloudCompare*, with 19 manually-drawn landmarks (black points, mostly covered by pink points) and 12 digital landmarks (pink points), all labelled by their orders.

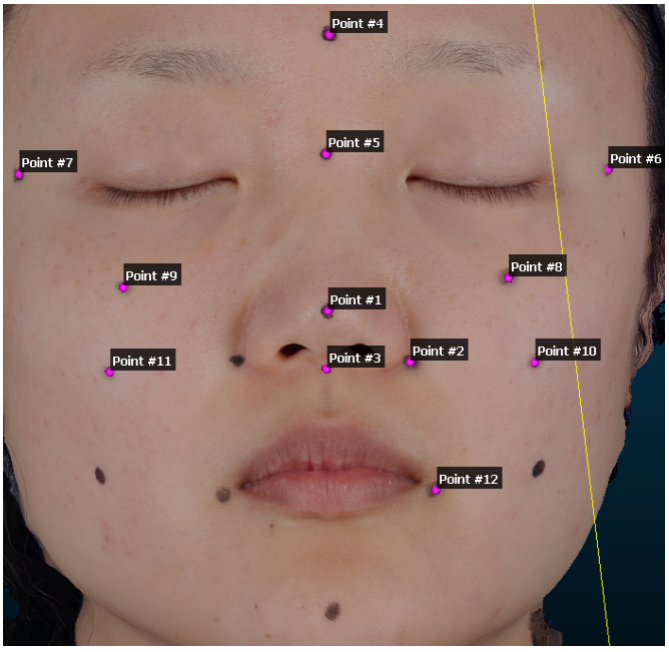



FIGURE 5.8: 19 manually-drawn landmarks and 12 digital landmarks used to compare the three machines

For each participant, each landmark has been estimated several times by three machines. Table A.1 summarises the standard deviation (sd) of each landmark (row) of each participant (a subject on the column) among all three machines, so that the variation of the performance of all machines together can be quantified. Because each landmark has three coordinates (three dimensions), the standard deviation of each landmark is represented by the average standard deviation of the three coordinates. Table A.2 summarises the performance between DI4D®

and NCTech® machines; while Table A.3 summarises the performance between DI4D® and Artec® machines.

By comparing the three tables below, it can be noticed that when the new system is involved in the performance comparison (Table A.1 and A.2), the standard deviation tends to be larger, but not dramatically so. As there are errors from operator to place different landmarks and the participants' different characteristics, the performance of the new system seems stable at this stage.

### 5.2.3 Comparison on Euclidean distances between landmarks

All 12 landmarks have been grouped to 6 pairs (green and grey lines in Figure 5.9) and the Euclidean distance of each pair has been calculated. The groups are points 1 and 3; points 2 and 12; points 4 and 5; points 6 and 7; points 8 and 9; points 10 and 11 whose orders are shown in Figure 5.8.

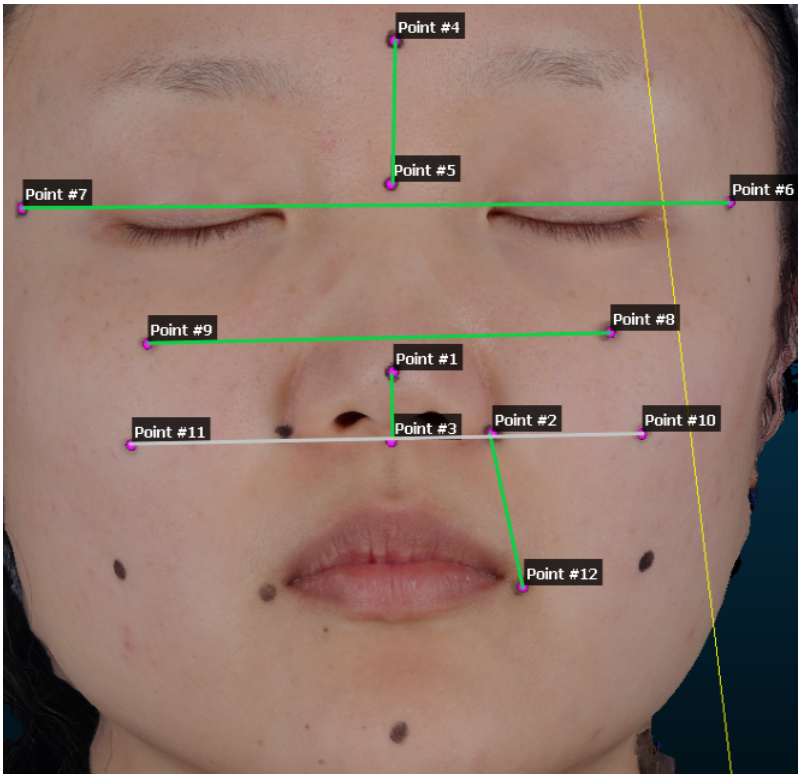



FIGURE 5.9: 6 pair of lines used to compare the three machines

Table A.4 summarises standard deviation of the distance of each pair (line on the row) and each participant (subject on the column) among all three machines. Table A.5 summarises the performance between DI4D® and NCTech® machines; while Table A.6 summarises the performance between DI4D® and Artec® machines. Consistent with the results in Section 5.2.2, the performance of the new camera system increases the standard deviation of most distances bewteen landmarks. However, the increase is not large and there even some decreases, such as line 3 of subject 4 from Table A.5 to Table A.6.

### 5.2.4 Deviations of each camera system

Previous studies focused on the standard deviation of the landmarks or distances of paired landmarks across machines. In this section, the standard deviation, or the variation of the estimate will be calculated by the same machine.

In this validation study, for each of 12 digital landmarks, NCTech® machine and DI4D® machine each estimated 6 participants× 2 repeats which is 12 times, while the Artec® machine estimated this 11 times. The deviation of each landmark estimation by each camera system can be calculated by subtracting the mean position from the estimates. In Figure 5.10, the deviations of each landmark of each participant from the same machine are in the same colour. The advantage of this comparison is that behaviours from different camera systems can be compared regardless of landmarks and subjects. Note that in this section, the deviation of each landmark is no longer the average on the three dimensions, but displayed directly in three-dimensional space. The dimensions are also shown in Figure 5.11 in three different colours.

In Figure 5.10, although some of the blue points are large, most of them are clustered in the centre. As the red points from DI4D® machine have a degree of dispersion, ignoring the far blue points (regarded as outliers), most of the blue have a reasonable dispersion. In Figure 5.11, the box sizes of NCTech® seem similar to the other two machines, although Artec® has a better behaviour. Thus, we

can conclude the performance of the new camera system is comparable with those mature products.

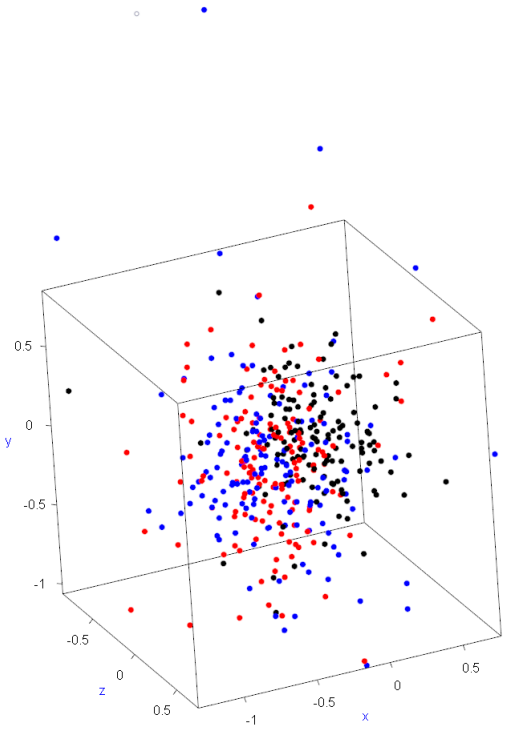



FIGURE 5.10: Deviations (represented by dots) of landmark estimates from Artec® (black), DI4D® (red) and NCTech® (blue) camera systems.

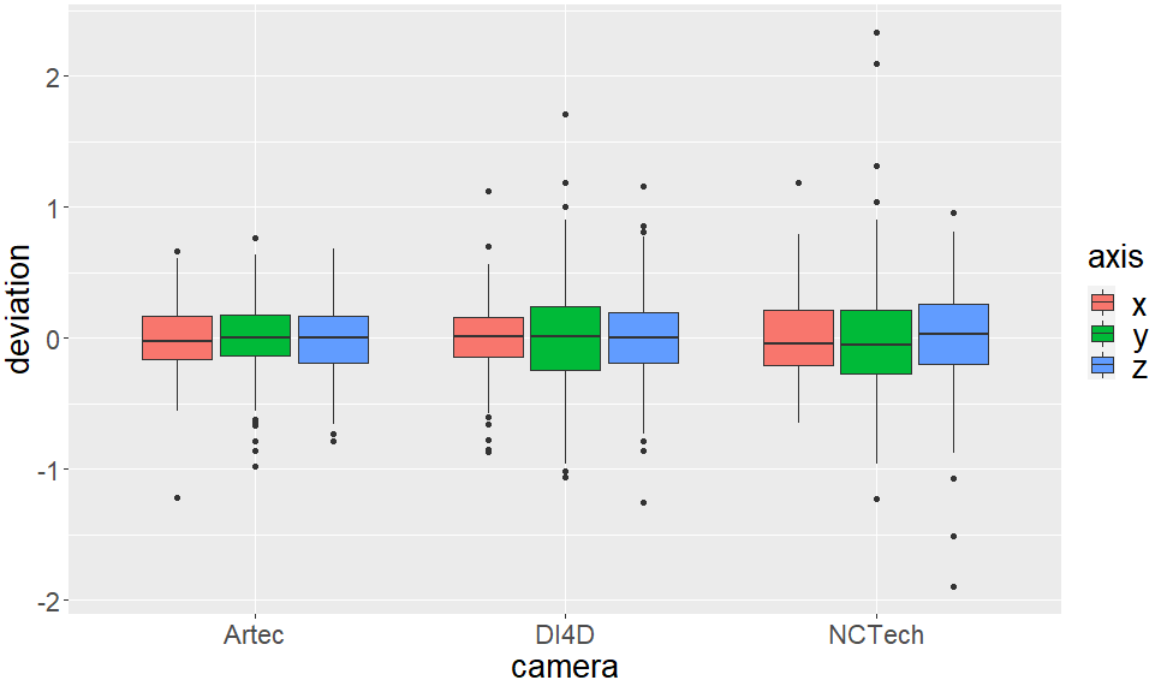



FIGURE 5.11: Boxplots of deviations of landmark estimates from all three cameras.

The Euclidean distances can be analysed in the same way. The deviations of the Euclidean distances between pairs of landmarks are shown in Figure 5.12 and Figure 5.13. Different from the landmark estimates, the Euclidean distance between a pair of landmarks only has one dimension, so is the deviation. The new camera system still has slightly larger spread of the deviations. Another feature in Figure 5.13 is that the Artec® machine behaves very stable, as the distribution of the deviation is symmetric and has the smallest variation.

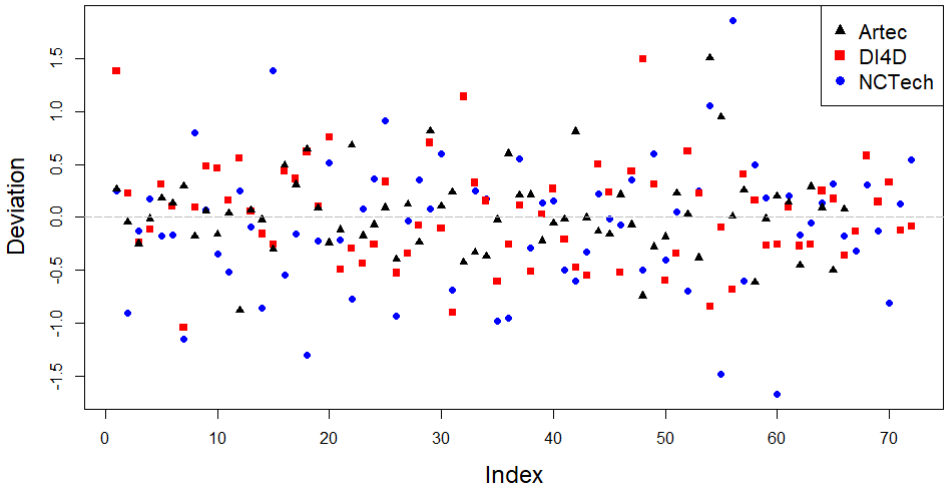



FIGURE 5.12: Deviations of distance estimates from all three cameras.

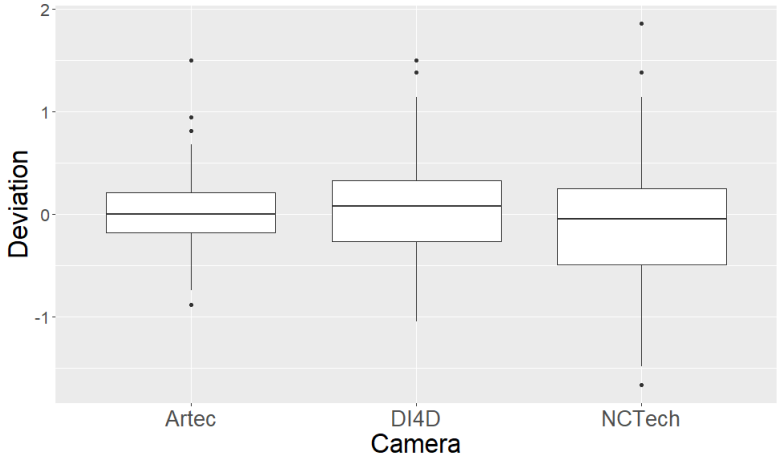
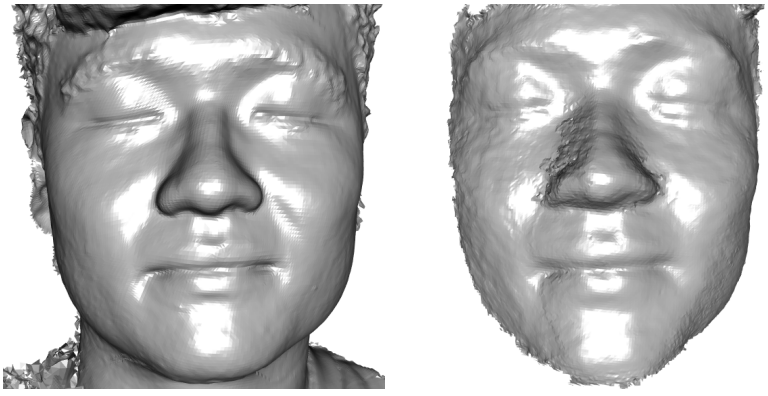



FIGURE 5.13: Boxplots of deviations of distance estimates from all three cameras.

## 5.3 Validation study 3: a later stage of development

### 5.3.1 Introduction

This validation study, at a later stage of the development, benchmarks the performance of NCTech® and DI4D® machines, based also on landmarks and curves. There are five participants. Each of them has been scanned twice by NCTech® and DI4D® machines. As the deviance within one machine has been investigated in previous validation studies, this one compares one image from the NCTech® machine with the same participant's image from the DI4D® machine. Figure 5.14(a) and 5.14(b) show an example of a pair of images from these two machines without texture. This allows detailed comparison between the performance of the two machines. The parameter of resolution has been reduced to 0.001, in order to compare with the outputs from DI4D® (in millimetre scale). With the improvement of the camera system, the quality of the image has been very much improved.



(a) DI4D® (b) NCTech®

FIGURE 5.14: Images coloured by grey from DI4D® and NCTech® camera systems

### 5.3.2 Comparison of landmarks

There are 21 landmarks manually placed on each image using *CloudCompare*. Figure 5.15 shows an example of a Procrustes-matched pair set of landmarks from DI4D® (black dots) and NCTech® (red dots) images of one participant. The landmarks are named based on the definitions in Katina et al. [26]. The method in this section is similar with the study in Section 5.1 where for each participant, we focus only on one image from one machine and compare the images.

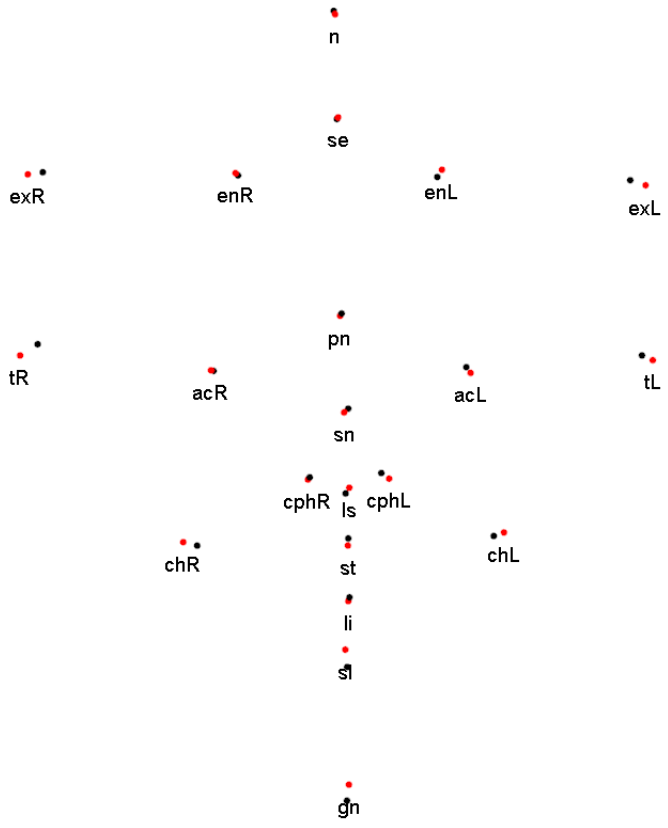



FIGURE 5.15: Procrustes matched landmarks from DI4D® (black) and NCTech® (red) images of one participant

For each participant on the column, the comparison between two images from two machines is summarised in Table A.7. The statistic "rmsd" for each participant

in this validation study is defined as follows.

$$\text{rmsd} = \sqrt{\frac{\sum_{i=1}^{21} {d_i}^2}{21}} \tag{5.2}$$

where 21 is the number of landmarks. Denote the Euclidean distance of the $i$th pair of Procrustes-matched landmarks by $d_i$, which is on the row for each landmark. Comparing with Table 5.1.2, the deviance of the same landmark position from two machines is much smaller, so is the standard deviation for each landmark across the participants.

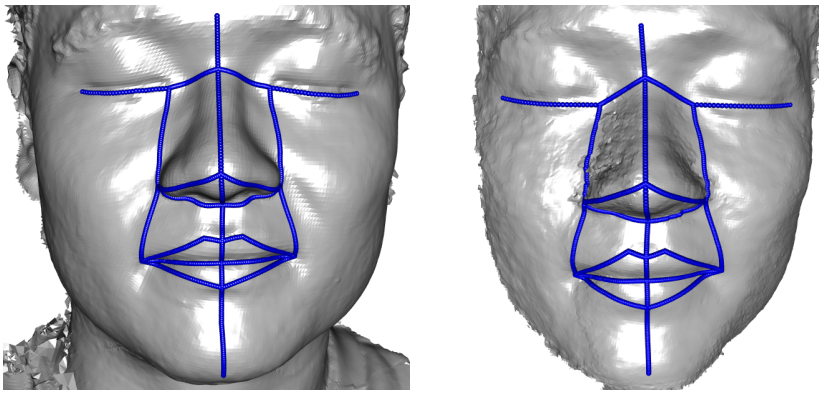
The effect on rmsd here consists of many more aspects other than the camera system variation of interest. The random effects include the variation of landmark placement by the observer, the variation of image capture by the same machine, the variation of different days to capture images, etc., which cannot be investigated in this study due to the small sample size. However, previous studies showed that it is likely that only observer variation has a large impact. In other words, it seems reasonable to conclude that the variation due to the observer dominates the variation between the pair set of landmarks, and the variation due to different machines can be small.

### 5.3.3 Comparison on curves

Figure 5.16(a) and 5.16(b) show a pair of images captured from the same participant by two machines, where the curves are identified by the methods in Section 3.2.1 and 3.2.2. Due to the difficulty of creating the curves, "upper lip right", "upper lip left", "lower lip right", "lower lip left", "nasal bridge right" and "nasal bridge left" were identified by optimising on the shortest arc length. Figure 5.17(a) and 5.17(b) show an example of Procrustes-matched pair set of curves from DI4D® (black dots) and NCTech® (red dots) images of the same participant. Note that the identified curve consists of a series of discrete 3D points.

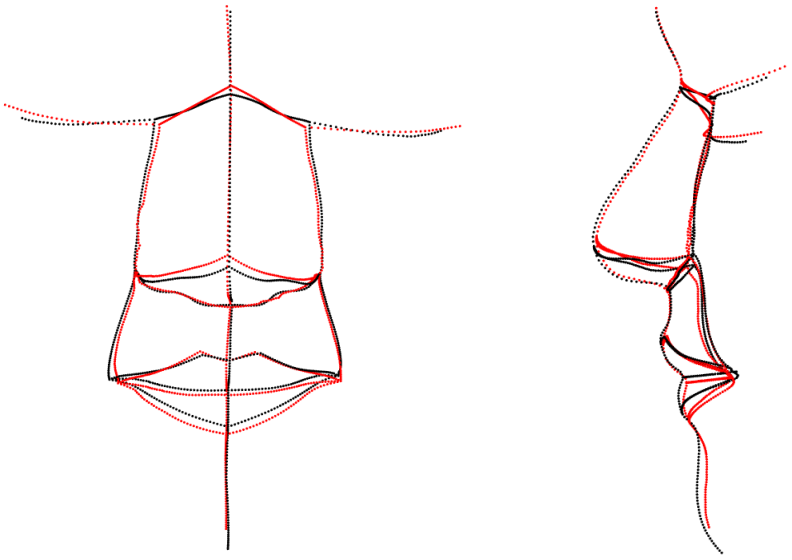
The comparison with curves also indicates that the variation between pair set of curves may be dominated by the variation of the landmark placement. For example, the surface in the pronasale area is a cap-like shape and bending the most on a human face. it can be difficult to identify the pronasale point which has the largest curvature by eyes. The same situation happens at the corners of the lip. Some people have thicker cheek whose corner of the lip can be more difficult to identify than the one whose bottom cheek area is flat. Furthermore, it can be problematic at the corners of the eyes, not only to identify the largest curvature point, but also the effect of lashes.

Large deviations of the curves mostly appear at the landmarks. It can be reasonable that they result from the manual placement of the landmarks. Apart from the error of landmark placemant, the curves seem consistent with each other. Although the overall quality of the image fro the new camera system is not as good as the mature prototype, the landmarks and curves within each anatomical area seem adequate for this facial-surgery project at this latter stage.



(a) DI4D® (b) NCTech®

FIGURE 5.16: Images from DI4D® and NCTech® camera systems with curves



(a) Front view (b) Lateral view

FIGURE 5.17: Procrustes matched curves from DI4D® (black) and NCTech® (red) images of one participant. The red curves are shown blue in Figure 5.16(a) and 5.16(b)

# Chapter 6

# Bayesian localisation of 3D facial landmarks

The variability of manually localising the landmarks is a difficulty in curve identification and shape analysis. For example, in Section 5.3.3, the manual localisation of the landmarks increases the variability of two sets of curves, which can mislead the comparison of the shapes. Another issue that may occur in the middle procedure of the analysis when some key anatomical landmarks are needed, but adding extra landmarks is very time-and-labour consuming. Thus, there is a demand to develop an automatic process which can remove the need for manual operations.

Since 1990, human facial landmarks have always played an important role in facial analysis applications, such as face recognition and expression analysis. In recent years, researchers (e.g. Cootes et al. [31], Sun et al. [70] and Deng et al. [71]) have focused more on automatic facial landmark localisation, known as face alignment or facial landmarks detection as, generally speaking, traditional manual landmarking is not accurate enough and is very time-and-labour consuming.

The methods of facial feature point detection (FFPD) have been categorised by Wang et al. [72], into the parametric and the non-parametric shape model-based. For parametric methods, Cootes et al. [31] introduced the point distribution model

(PDM) which Procrustes matches the training shapes and performs principal component analysis (PCA) to build a shape model. Wang et al. [72] noted that the most popular one is called Active appearance model (AAM), which combines PDM and texture models by PCA, as proposed by Cootes et al. [73].

Non-parametric methods have made considerable progress in the last few years, especially in computer science, such as the deep convolution neural network (DCNN) method demonstrated by Sun et al. [70]. Although Hu et al. [74] improved by only using 1000 data, computing methods including DCNN still involve big data sets to train the model. Zhang et al. [75] have also introduced a novel data and model-driven (parametric and non-parametric) approach, called Estimation-Correction-Tuning (ECT), which combines the advantages of the global robustness of data-driven methods (fully convolutional networks), outlier correction capability of the model-driven method (PDM) and non-parametric optimisation of Regularised Landmark Mean-Shift.

In this chapter, a novel and robust approach using Bayesian model will be introduced, which aims to use few data for training, but of high accuracy in a statistical sense. This method is a further development of Vittert [8]. It has been only developed to deal with four key landmarks, but more will be investigated in future research. Further work of its comparison with existing methods is also required in future investigation.

## 6.1 Automatically estimate landmarks by Bayesian models

As mentioned in Vittert [8], Bayesian approach is motivated by our common sense of prior knowledge of human facial landmarks which have similar positions on different human faces. For example, a set of landmarks, as defined by Katina et al. [26], has the strongest curvatures on anatomical areas, such as the tip of the nose and inner corners of the eyes. Aided by the indication of landmark positions

estimated manually on a prior set, the landmarks on a new face can be localised according to local curvatures or the expected positions of different landmarks.

Although this procedure expects to estimate a full set of three-dimensional landmarks simultaneously, it can be difficult to deal with a high-dimensional multivariate distribution, especially when maximising the posterior distribution. Thus, this thesis will only include the estimation of four anatomical landmarks which are defined by Katina et al. [26] as follow.

- *pronasale*: the crossing of mid-line nasal profile[1] and alar curves[2]
- *sellion*: the point of maximum curvature of the mid-line nasal curve at its nasal root
- *alare crest*: the left and right points of maximum curvatures on the alar curve where this meets the paranasal area.

## 6.2 Prior distribution of landmarks

The automatic Bayesian method can be developed with an existing data set of facial images. Landmarks are manually placed on the faces to provide prior knowledge on how the landmark positions are distributed on a human face. Methods for modelling a three-dimensional data set are illustrated in Section 2.2. The prior landmark sets on facial images are first matched through Generalised Procrustes Analysis (GPA) and their Procrustes tangent coordinates (Procrustes residuals) are calculated through Equation 2.13. The Procrustes residuals are vectorised as Equation 2.11 and each vector has length $3k = 12$ where $k$ is the number of landmarks in the set to estimate and $k = 4$ in this chapter.

The most common distribution of such high-dimensional (12-dimensional) modeling, the multivariate normal distribution, is used for the prior distribution of the

[1] Mid-line nasal profile contains ridge points from the nasal root along the dorsum of the nose and the columella

[2] Alar curve contains ridge points on the lateral extension of the nasal cartilage

positions of the landmark set. The set of landmarks to choose is expected to maximise the posterior distribution. Procrustes registration of the set of landmarks provides the prior distribution, denoted by $N_{12}(\bar{r}, s_r)$ where $\bar{r}$ (vector of length 12) and $s_r$ (square matrix with 12 rows/columns) are the mean and the covariance matrix of the Procrustes residuals as defined in Equation 2.13 and 2.14. The prior distribution can be expressed proportionally as

$$\rho(\theta) \propto \exp\left\{-\frac{1}{2}R^{\intercal}{s_r}^{-1}R\right\} \tag{6.1}$$

where $R$ denotes the matrix of Procrustes residuals. Note that the inverse matrix of $s_r$ may not exist, so the generalised inverse matrix is used. This satisties $s_r{s_r}^{-1}s_r = s_r$.

## 6.3 Likelihood distribution of landmarks

### 6.3.1 Initial estimates of landmarks

In Bayesian statistics, the likelihood is constructed from the conditional probability of observations given the landmark positions. Thus, the set of landmarks to choose is that which maximises the likelihood. A 3D facial image contains more than one hundred thousand points which are estimated by the camera system. An efficient way to find the optimal landmark set can be to narrow the optimisation area to a small neighbourhood of each landmark.

Firstly, the gross local features can be examined by selecting a subset of the point cloud with the Euclidean distance in the neighbourhood to be about 50 millimetres. Because a meaningful landmark is either at the peak of a cap or the bottom of a cup area, shape index can be used to search for the spherical cap or cup, as indicated by the colours in Figure 2.4. With the characteristic of a human face, the optimal local area of the tip of the nose can be found around the point in the subset which has a positive shape index and the strongest Gaussian curvature.

This point has a negative $\kappa_{min}$ on a ridge, as described in Section 2.2, and is used as an initial estimate of *pronasale*.

The nose-tip area has a positive shape index, while the inner corners of the eyes have negative shape indices. A curve between the two points which have the largest Gaussian curvatures in the corners will lie across the nose. The point on this curve with the strongest (smallest negative $\kappa_{min}$) will be the initial estimate of *sellion*.

The initial positions of the left and right *alare crest* can be estimated by matching the initial positions of *pronasale* and *sellion* to the prior landmark sets. After Procrustes matching, the average left and right *alare crest* in the prior landmark sets will intersect the face by rotating the line which goes through the initial positions of *pronasale* and *sellion*. The two intersecting points can be the initial estimates of the left and right *alare crest*.

### 6.3.2 Likelihood distribution

After finding the initial locations of the four landmarks, the optimisation area can be narrowed from the whole face to their small neighbourhood. Denote the coordinates of the $i$th landmark by $\theta_i$, $i = 1, \ldots, k$, the coordinates of its $n_i$ neighbouring points of by $\{\tilde{m}_{i1}, \tilde{m}_{i2}, \ldots, \tilde{m}_{in_i}\}$ and their curvatures by $y_i$=$\{y_{i1}, y_{i2}, \ldots, y_{in_i}\}$. On the small patch of each landmark, fit a quadratic model to their curvatures as follows.

$$\mathrm{E}(y_{ij}) = \beta_{i1} + \beta_{i2}\|\theta_i - \tilde{m}_{ij}\|^2, \quad j = 1, \ldots, n_i \tag{6.2}$$

or alternatively,

$$Y_i = X_i\beta_i, \quad \text{where} Y_i = \begin{bmatrix} y_{i1} \\ y_{i2} \\ \vdots \\ y_{in_i} \end{bmatrix}, X_i = \begin{bmatrix} 1 & \|\theta_i - \tilde{m}_{i1}\|^2 \\ 1 & \|\theta_i - \tilde{m}_{i2}\|^2 \\ \vdots & \vdots \\ 1 & \|\theta_i - \tilde{m}_{in_i}\|^2 \end{bmatrix}, \beta_i = \begin{bmatrix} \beta_{i1} \\ \beta_{i2} \end{bmatrix} \tag{6.3}$$

The likelihood distribution of the curvatures around the $i$th landmark $\theta_i$ is then

$$\rho(Y_i|X_i,\beta_i,{\sigma_i}^2) \propto ({\sigma_i}^2)^{-\frac{n_i}{2}} \exp\left\{-\frac{1}{2{\sigma_i}^2}(Y_i - X_i\beta_i)^\intercal(Y_i - X_i\beta_i)\right\} \tag{6.4}$$

In this $n_i$-dimensional multivariate normal distribution, we use hyper-parameters for the distributrions of $\beta_i, {\sigma_i}^2$

$$\beta_i \sim N_2(\mu_{0_i}, {\sigma_i}^2{\Lambda_{0_i}}^{-1}) \quad \text{or} \quad \rho(\beta_i|{\sigma_i}^2) \propto ({\sigma_i}^2)^{-1} \exp\left\{-\frac{1}{2{\sigma_i}^2}(\beta_i - \mu_{0_i})^\intercal\Lambda_{0_i}(\beta_i - \mu_{0_i})\right\}$$
$${\sigma_i}^2 \sim \text{Inverse Gamma}(a_{0_i}, b_{0_i}) \quad \text{or} \quad \rho({\sigma_i}^2) = ({\sigma_i}^2)^{-a_{0_i}-1} \exp\left\{-\frac{b_{0_i}}{{\sigma_i}^2}\right\}$$

Following the notation of Molinari [76], the joint conjugate prior distribution of $\beta_i$ and ${\sigma_i}^2$ is a Normal-Inverse Gamma and denoted by $NIG_2(\mu_{0_i}, \Lambda_{0_i}, a_{0_i}, b_{0_i})$.

$$\begin{aligned}\rho(\beta_i, {\sigma_i}^2) &= \rho(\beta_i|{\sigma_i}^2)\rho({\sigma_i}^2) \\ &\propto ({\sigma_i}^2)^{-1} \exp\left\{-\frac{1}{2{\sigma_i}^2}(\beta_i - \mu_{0_i})^\intercal\Lambda_{0_i}(\beta_i - \mu_{0_i})\right\} ({\sigma_i}^2)^{-a_{0_i}-1} \exp\left\{-\frac{b_{0_i}}{{\sigma_i}^2}\right\}\end{aligned} \tag{6.5}$$

## 6.4 Posterior distribution

According to Bayes theorem, the posterior distribution of the $i$th landmark $\theta_i$ can be expressed as follows.

$$\begin{aligned}
\rho(\beta_i, {\sigma_i}^2|Y_i, X_i) &\propto \rho(Y_i|X_i, \beta_i, {\sigma_i}^2)\rho(\beta_i, {\sigma_i}^2) \\
&\propto ({\sigma_i}^2)^{-\frac{n_i}{2}} \exp\left\{-\frac{1}{2{\sigma_i}^2}(Y_i - X_i\beta_i)^\intercal(Y_i - X_i\beta_i)\right\} \\
&\quad ({\sigma_i}^2)^{-1} \exp\left\{-\frac{1}{2{\sigma_i}^2}(\beta_i - \mu_{0_i})^\intercal\Lambda_{0_i}(\beta_i - \mu_{0_i})\right\} \\
&\quad ({\sigma_i}^2)^{-a_{0_i}-1} \exp\left\{-\frac{b_{0_i}}{{\sigma_i}^2}\right\} \\
&= ({\sigma_i}^2)^{-1} \exp\left\{-\frac{1}{2{\sigma_i}^2}(\beta_i - \mu_{n_i})^\intercal\Lambda_{n_i}(\beta_i - \mu_{n_i})\right\} ({\sigma_i}^2)^{-a_{n_i}-1} \exp\left\{-\frac{b_{n_i}}{{\sigma_i}^2}\right\}
\end{aligned} \tag{6.6}$$

where
$$\begin{aligned}
\mu_{n_i} &= ({X_i}^\intercal X_i + \Lambda_{0_i})^{-1}(\Lambda_{0_i}\mu_{0_i} + {X_i}^\intercal Y_i) \\
\Lambda_{n_i} &= {X_i}^\intercal X_i + \Lambda_{0_i} \\
a_{n_i} &= a_{0_i} + \frac{n_i}{2} \\
b_{n_i} &= b_{0_i} + \frac{1}{2}\left({Y_i}^\intercal Y_i + {\mu_{0_i}}^\intercal\Lambda_{0_i}\mu_{0_i} - {\mu_{n_i}}^\intercal\Lambda_{0_i}\mu_{n_i}\right)
\end{aligned}$$

Compare the formula of Equation 6.6 with 6.5, the likelihood distribution is a Normal-Inverse Gamma $NIG_2(\mu_{n_i}, \Lambda_{n_i}, a_{n_i}, b_{n_i})$. With the property of any probability distribution that

$$\iint \frac{{b_{n_i}}^{a_{n_i}}}{(2\pi)|\Lambda_{n_i}|^{\frac{1}{2}}\Gamma(a_{n_i})}\rho(\beta_i, {\sigma_i}^2|Y_i, X_i)\, d\beta_i\, d{\sigma_i}^2 = 1 \tag{6.7}$$

the marginal likelihood distribution of $\theta_i$ is

$$\begin{aligned}
\rho(Y_i|\theta_i) &= \iint \rho(Y_i|X_i, \beta_i, {\sigma_i}^2)\rho(\beta_i, {\sigma_i}^2|\theta_i)\, d\beta_i\, d{\sigma_i}^2 \\
&= \iint \rho(\beta_i, {\sigma_i}^2|Y_i, X_i)\, d\beta_i\, d{\sigma_i}^2 \\
&= \frac{(2\pi)|\Lambda_{n_i}|^{\frac{1}{2}}\Gamma(a_{n_i})}{b_{n_i}^{a_{n_i}}} \iint \frac{b_{n_i}^{a_{n_i}}}{(2\pi)|\Lambda_{n_i}|^{\frac{1}{2}}\Gamma(a_{n_i})}\rho(\beta_i, {\sigma_i}^2|Y_i, X_i)\, d\beta_i\, d{\sigma_i}^2 \\
&\propto \frac{|\Lambda_{n_i}|^{\frac{1}{2}}\Gamma(a_{n_i})}{b_{n_i}^{a_{n_i}}}
\end{aligned} \tag{6.8}$$

For simplification, suppose the mean Procruste residual $\mu_{0_i} = 0$:

$$\begin{aligned}
\Lambda_{n_i} &= {X_i}^\intercal X_i + \Lambda_{0_i} \\
\mu_{n_i} &= (\Lambda_{n_i})^{-1}({X_i}^\intercal Y_i) \\
a_{n_i} &= a_{0_i} + \frac{n_i}{2} \\
b_{n_i} &= b_{0_i} + \frac{1}{2}\left({Y_i}^\intercal Y_i - {\mu_{n_i}}^\intercal \Lambda_{0_i}\mu_{n_i}\right)
\end{aligned}$$

Thus, the chioce of the set of four landmarks is which maximises

$$\rho^\star \propto \rho(\theta)\prod_{i=1}^{4}\rho(Y_i|\theta_i) \tag{6.9}$$

For convenience, take logarithm of $\rho^\star$ and substitiute Equation 6.1 and 6.8, we have

$$\begin{aligned}
\rho_l^\star &= \log\{\rho^\star\} \\
&= \log\{\rho(\theta)\} + \sum_{i=1}^{4}\log\{Y_i|\theta_i\} \\
&= -\frac{1}{2}R^\intercal {s_r}^{-1}R + \sum_{i=1}^{4}\left(\frac{1}{2}|\Lambda_{n_i}| + \log\{\Gamma(a_{n_i})\} - a_{n_i}\log(b_{n_i})\right)
\end{aligned} \tag{6.10}$$

## 6.5 Algorithm

As mentioned in the beginning of this section, it can be computationally expensive to find the optimal set of four landmarks which maximising the posterior simutaneously. An efficient solution is to find them one-by-one while holding the others fixed. With the input of an individual face where landmarks are to be estimated and a prior set of faces with their landmarks manually placed, the algorithm is as follows.

Step 1: Find the initial positions of the four landmarks and their neighbourhood with curvatures; denote this initial set of landmarks by $L_{old} = \left\{{\theta_1}^0, {\theta_2}^0, {\theta_3}^0, {\theta_4}^0\right\}$;

Step 2: Replace the $i$th landmark with one in its neighbourhood; Procrustes match this replaced landmark set with the prior sets of landmarks for the prior probability $\rho(\theta)$; calculate $\rho_l^\star$ by Equation 6.10 of this replaced landmark set;

Step 3: Run Step 2 for all the neighbouring points of ${\theta_i}^0$, find the one which maximises $\rho_l^\star$ and denote it by ${\theta_i}^1$. Repeat for $i = 1, 2, 3, 4$ so that to obtain $L_{new} = \left\{{\theta_1}^1, {\theta_2}^1, {\theta_3}^1, {\theta_4}^1\right\}$;

Step 4: Calculate the point-by-point Euclidean distance between $L_{old}$ and $L_{new}$. If the distance is within certain small amount, finish the algorithm. Otherwise, replace $L_{old}$ with $L_{new}$ and repeat from Step 2 until convergence.

## 6.6 Application of the algorithm

Figure 6.1 shows the comparison of different methods and indicates the efficiency of this algorithm. Green dots are manually placed on the face as an reference; black dots are the initial estimate before applying the Bayesion algorithm (Section 6.3.1) and red dots are the estimate of the Bayesian algorithm (Section 6.5). Both the

*pronasale* and *sellion* show that the algorithm adjusted (corrected) the (black) initial estimate by moving them closer to the (green) manual placements.

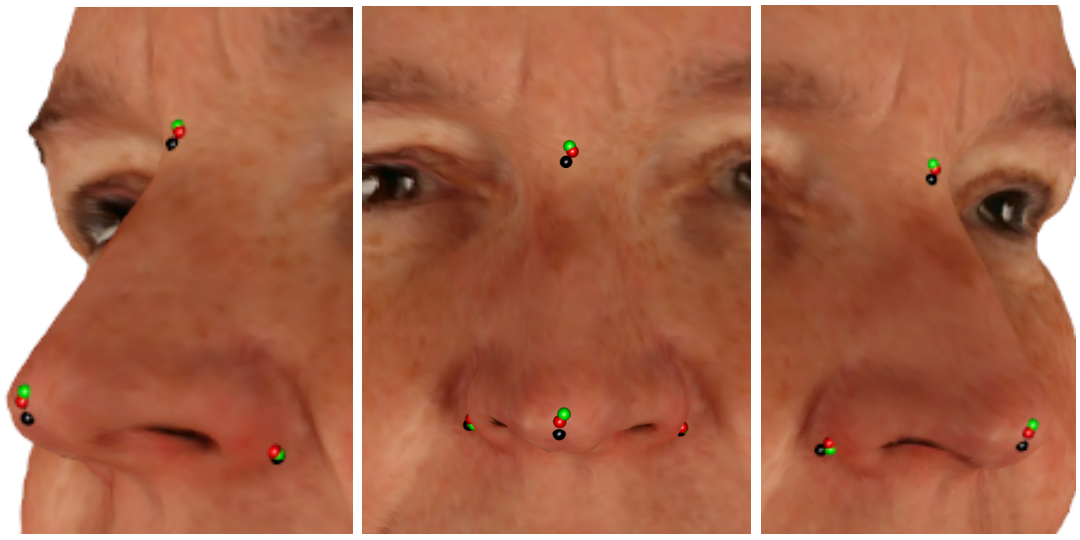


FIGURE 6.1: Comparison of landmarks estimated by different methods: manual placements (green dots), initial positions (black dots) and Bayesion approach (red dots)

One advantage of this method is that it reduces the variation from manual placements. This can be tested by comparing the Procrustes-matched variation of manually-placed landmark set and their Bayesian estimate set. There are 31 three-dimensional facial images captured from 31 female adults in the University of Glasgow by a DI3D machine. Both manual placement and Bayesian estimate of the four landmarks are obtained and matched by the Generalised Procrustes Analysis (without scaling). The Procrustes-matched variation of each set is summarised in Table 6.6.

| | **rmsd1** | **GSS** |
|---|---|---|
| **Manual** | 0.135 | 1584 |
| **Bayesian** | 0.069 | 485 |

TABLE 6.1: Comparison of the Procrustes-matched variation of manual placements and Bayesian estimate of the same group of participants.

where rmsd1 (unit: millimetre) is the square root of the average Procrustes distance[3] squared to their mean shape; GSS is the sum of squared norms of pairwise differences as defined in Dryden and Mardia [29] and shown in Equation 2.9. The table shows that the variation of the manual placements is more than twice of the Bayesian estimates. In other words, this Bayesian model efficiently reduced the error of manual placements which is not of interest.

| | ***pronasale*** | ***sellion*** | ***alare crest (L)*** | ***alare crest (R)*** |
|---|---|---|---|---|
| **mean** | 2.9 | 3.1 | 3.4 | 3.0 |
| **sd** | 1.5 | 8.4 | 2.1 | 2.2 |

TABLE 6.2: Average distances between the estimate and manual placement for each landmark and the standard deviations. (unit: millimetre)

Table 6.6 shows the performance of the algorithm specifically for each landmark. The Euclidean distance between the estimate and manual placement of each landmark on an individual face is calculated. The distance for each landmark indicates the performance of the algorithm is stable, with an average around 3mm. Among the four landmarks, *sellion* lies on a relatively flat surface, while others have a strong curvature in the neighbourhood. This may result in the large value of standard deviation so is a large variety of its position approximation. The overall performance of the algorithm is good that there is no large deviation from the manual reference but small adjustments. Those adjustments are more reasonable based on a Bayesian model than the placements only judged by eyes.

[3]The definitions of Procrustes distance and GSS are in Section 2.2.2

# Chapter 7

# Conclusion and Discussion

This thesis focuses on novel 3D shape modelling of local features and statistical analysis of the models. It is worth mentioning that conventional approaches are limited to 2D shape analysis, where the latest 3D models are still projected into a 2D plane. Therefore, this thesis conveys a revolutionary method for shape local features based on 3D raw data.

This chapter will provide conclusions and evaluations of the materials in this thesis, followed by some suggestions in future works in Section 7.2.

## 7.1 Conclusions and limitations

Traditional shape analysis focuses on anatomical points on the images which are far from each other known as "landmarks". Particularly, the analysis for local features requires more information and adds additional points between the landmarks, which are called "semi-landmarks". This thesis also investigates local features based on discrete points, but with richer geometric information but less complexity. Note that only the vertices of triangles are observations from the camera system and other points on the curve are estimated. Thus, the method to shape local features in this thesis is called "3D curve estimation".

Chapter 3 summarised two novel approaches for curve estimation. The first described in Section 3.3 is a development of an existing method introduced in Vittert et al. [7]. The existing method uses Euclidean distance for a fast speed and it works well on a relatively flat surface, such as the upper lip edge on a face. However, it could fail on a complex surface which bends heavily, such as a cichlid mandible bone. The improvement presented in this thesis has replaced the Euclidean distance by the distance along the surface, which the existing method was approximated by.

This improvement to calculate the distance along the surface also has fast performance and can deal with the surface which bends even in the opposite direction. However, because it estimates a curve with discrete points, a smoothing procedure is involved and this can be influenced by the smoothing parameter. The solution can be fixing a smoothing parameter at a conventional value with iteration and repeating estimation until convergence. This works well on surfaces where there is enough curvature information. However, it requires the guidance of curvatures for the whole curve reaching the local feature. If the local feature has weak curvatures in the middle, the iteration may be misled. For example, it may continue beyond the target feature to find stronger curvatures so that the convergence does not exist. Thus, the second novelty introduced in Section 3.4 in Chapter 3 aims to deal with surfaces which do not have an entire clear shape to identify, or where it is difficult to add landmarks to guide the estimating direction.

In such an ambiguous situation, curve estimation method needs to be very flexible. For example, when there is a ridge to identify and there is a small valley area in the way, the method may "jump" across the valley and continue the ridge path by determining a parameter ("step distance") at the beginning. Another example can be where there are several side paths in the middle, and the method can find the correct one by determining another pair of parameters ("direction boundaries"). It can be difficult to identify ridges and valleys on a child's face, as it is much flatter than an adult's face. This method has a good performance in estimating upper cleft lip edge of children where a break is in the middle.

Chapter 4 discussed four topics of shape analysis based on the estimated curves from Chapter 3. The first one in Section 4.1 is a novel method to dissect a local piece from a 3D shape. Popular software to edit images such as Photoshop by Adobe Inc. only provides functions for 2D images. Tools to edit 3D images are still under development. It acts like the scissor tool on 2D photos, but cuts along the 3D curves estimated in Chapter 3. The curve can be either closed or a path across the surface, which isolates some parts of the image or divides the image into parts. The algorithm works well on the dissection of single local shape, such as the lip or nose on a human facial image, but may fail if there are several connected local shapes isolated by a curve. In this case, each part of the local shapes should be computed separately.

The second topic discussed in Section 4.2 in Chapter 4 is to use principal component analysis (PCA) to compare the mean shapes of different groups, such as gender. In contrast to the conventional way to use PCA in statistics, shape analysis applied PCA on each coordinate of each point. This thesis used cichlid mandibles to explain how to extract local features from 3D curves using significant principal components' scores and specified the average shape difference of different gender groups. Although the percentage of variance that the significant principal components explain is normally not large, linear discriminant analysis (LDA) of the curves produced a classification rate up to 77%.

The last two topics included in Section 4.3 and 4.4 in Chapter 4 adapted the method introduced in Miller [63] to calculate curvatures of the curves and reconstruct the curve based on the curvatures. Other methods to align the curvature curves were discussed and an improved warping result was shown. During the reconstruction procedure, this thesis mainly investigated the effect of noise from the data on the optimal degree of freedom when fitting the discrete points. The reconstruction at the curve end tended to have larger deviance, which may result from the loss of curvature information. A higher density of points on the curve can address this issue, as it can provide more curvature information and make the curvature curve smoother.

After having Chapter 3 and 4 introduced, Chapter 5 went back to the data collection. using the technology introduced in Chapter 1, we have found that the existing camera systems are not portable and may disturb children. Hence, a new camera system is needed for our intended research topics, while its development for the project assessing the surgery of facial deformity in Brazil had commenced. At each stage of development, a validation study was designed to benchmark the performance of the new system and mature products. Both the landmarks and estimated curves were used to evaluate the differences in the images.

One issue is that the image difference of the same object may be enlarged by manual placements error of landmarks. The error which is not of interest may increase the total difference of images. One solution is to compare the manually-placed digital landmarks with the physical landmarks which were drawn on the participants' face. The colour of the physical landmark provides a reference to place the digital landmark rather than judging by eyes. Thus, the variation from placing the landmarks in the total difference can be reduced. It is also interesting to study how much variation has been reduced and how the total variation in a model can be decomposed.

Another solution to reduce the placement error was illustrated in Chapter 6, which aimed to automatically place the landmarks on three-dimensional images instead of manual placing. This novel Bayesian approach used a set of landmarks which were manually placed on the prior set of faces and the curvatures on the face to estimate its landmarks. The algorithm first used shape index and curvatures to estimate initial positions of the landmarks, and then found the optimal positions which maximised the object distribution one-by-one through iterations. By comparing the initial position, manual placement and the Bayesian estimation of the same landmark, the method corrected the landmark from its initial position. The more similar the prior set of faces to the observed face, the better result of the algorithm. The likelihood was calculated by a quadratic model of the points in its neighbourhood and it requires a high density of the point cloud, with the distance between adjacent points around one millimetre.

The work illustrated in this thesis has been planned to publish. Chapter 3 and 4 includes the methods in statistics and computational geometry, targeting journals such as Studies in Applied Mathematics and Applied Mathematics Letters. Chapter 6 introduced a novel Bayesian model for automatic facial landmark detection, targeting journals such as Journal of the Royal Statistical Society Series B-Statistical Methodology, Annals of Statistics and Statistical Science.

## 7.2 Further research interests

The Bayesian model to automatically detect landmarks in Chapter 6 is still at the early stage. A full set of 23 landmarks needs to be investigated, as well as a general Bayesian method to detect landmarks on any shapes. According to the principle of Bayesian models, as long as there is a prior set of similar shapes and a proper likelihood distribution, the position of the landmarks can be estimated. But there are plenty of details to investigate, including the influence of prior size and similarity between the prior set and the shape to estimate. A validation study to evaluate the Bayesian approach is also required.

A strong connection between the methods in this thesis is curve estimation and automatic landmark detection. Curve estimation introduced in Chapter 3 was based on the manual landmark placements, while the Bayesian automatic detection was studied afterwards. Thus, a procedure to automatically estimate curves without manual landmark placements can be investigated. This is not only useful in shape analysis, but also in the validation study assessing the behaviour of camera systems, as described in Chapter 5.

The methods of building statistical models and extracting local shape features have a wide range of applications. In the world of big data, machine learning is the most popular tool across disciplines, particularly in image processing. With an image as the input, convolutional neural networks (CNN) use filters in layers to analyse and recognise the objects. However, on one hand, as the method is mainly used to predict, one of the disadvantages of this method is that the parameters of

its model are difficult to interpret. A high prediction accuracy may be obtained by adjusting the parameters or functions which do not have the explainable meaning of the data. RichardWebster et al. [77] mentioned that the reason for a failed prediction can be hard to understand. On the other hand, because there is too much information which is sensible, a little improvement of the model may require a huge increase in the training data size. 3D curve estimation and landmark detection can help build shape models as inputs instead of the raw image. This procedure may reduce the required training data set and achieve higher accuracy.

# Appendix A

# Long Tables in Chapter 5

| **participant** | **1** | **2** | **3** | **4** | **5** | **6** |
|---|---|---|---|---|---|---|
| lmk 1 | 0.50 | 0.19 | 0.35 | 0.31 | 0.51 | 0.19 |
| lmk 2 | 0.56 | 0.49 | 0.43 | 0.24 | 0.81 | 0.29 |
| lmk 3 | 0.24 | 0.30 | 0.36 | 0.16 | 0.26 | 0.15 |
| lmk 4 | 0.21 | 0.59 | 0.27 | 0.39 | 0.29 | 0.15 |
| lmk 5 | 0.30 | 0.30 | 0.29 | 0.28 | 0.34 | 0.14 |
| lmk 6 | 0.32 | 0.29 | 0.41 | 0.34 | 0.38 | 0.21 |
| lmk 7 | 0.31 | 0.46 | 0.24 | 0.49 | 0.60 | 0.28 |
| lmk 8 | 0.29 | 0.15 | 0.43 | 0.21 | 0.30 | 0.18 |
| lmk 9 | 0.30 | 0.36 | 0.26 | 0.49 | 0.40 | 0.22 |
| lmk 10 | 0.27 | 0.32 | 0.46 | 0.33 | 0.56 | 0.17 |
| lmk 11 | 0.28 | 0.38 | 0.31 | 0.33 | 0.68 | 0.14 |
| lmk 12 | 0.61 | 0.31 | 0.32 | 0.41 | 0.28 | 0.14 |

TABLE A.1: Average standard deviations of landmark coordinates of all three machines (unit: millimetre)

| **participant** | **1** | **2** | **3** | **4** | **5** | **6** |
|---|---|---|---|---|---|---|
| lmk 1 | 0.51 | 0.19 | 0.43 | 0.37 | 0.62 | 0.17 |
| lmk 2 | 0.67 | 0.43 | 0.51 | 0.29 | 0.91 | 0.31 |
| lmk 3 | 0.16 | 0.19 | 0.44 | 0.17 | 0.24 | 0.16 |
| lmk 4 | 0.22 | 0.62 | 0.32 | 0.38 | 0.33 | 0.14 |
| lmk 5 | 0.29 | 0.25 | 0.33 | 0.30 | 0.41 | 0.15 |
| lmk 6 | 0.35 | 0.35 | 0.51 | 0.34 | 0.45 | 0.20 |
| lmk 7 | 0.35 | 0.38 | 0.29 | 0.59 | 0.64 | 0.29 |
| lmk 8 | 0.26 | 0.13 | 0.39 | 0.17 | 0.30 | 0.12 |
| lmk 9 | 0.26 | 0.23 | 0.29 | 0.55 | 0.45 | 0.18 |
| lmk 10 | 0.25 | 0.31 | 0.59 | 0.38 | 0.66 | 0.19 |
| lmk 11 | 0.26 | 0.26 | 0.39 | 0.35 | 0.59 | 0.14 |
| lmk 12 | 0.48 | 0.35 | 0.31 | 0.38 | 0.29 | 0.12 |

TABLE A.2: Standard deviation of landmark coordinates of DI4D® and NCTech® machines (unit: millimetre)

| **participant** | **1** | **2** | **3** | **4** | **5** | **6** |
|---|---|---|---|---|---|---|
| lmk 1 | 0.22 | 0.18 | 0.27 | 0.28 | 0.34 | 0.17 |
| lmk 2 | 0.21 | 0.30 | 0.23 | 0.13 | 0.16 | 0.31 |
| lmk 3 | 0.24 | 0.27 | 0.28 | 0.15 | 0.19 | 0.16 |
| lmk 4 | 0.17 | 0.73 | 0.16 | 0.27 | 0.17 | 0.14 |
| lmk 5 | 0.15 | 0.35 | 0.22 | 0.24 | 0.19 | 0.15 |
| lmk 6 | 0.19 | 0.19 | 0.36 | 0.25 | 0.20 | 0.20 |
| lmk 7 | 0.14 | 0.39 | 0.14 | 0.11 | 0.42 | 0.29 |
| lmk 8 | 0.21 | 0.19 | 0.37 | 0.18 | 0.16 | 0.12 |
| lmk 9 | 0.27 | 0.35 | 0.18 | 0.40 | 0.16 | 0.18 |
| lmk 10 | 0.21 | 0.27 | 0.39 | 0.20 | 0.38 | 0.19 |
| lmk 11 | 0.25 | 0.39 | 0.25 | 0.31 | 0.62 | 0.14 |
| lmk 12 | 0.36 | 0.19 | 0.28 | 0.33 | 0.18 | 0.12 |

TABLE A.3: Standard deviation of landmark coordinates of DI4D® and Artec® machines (unit: millimetre)

| **participant** | **1** | **2** | **3** | **4** | **5** | **6** |
|---|---|---|---|---|---|---|
| **line 1** | 0.95 | 0.13 | 0.67 | 0.39 | 0.85 | 0.22 |
| **line 2** | 0.55 | 0.57 | 0.90 | 0.38 | 0.94 | 0.43 |
| **line 3** | 0.27 | 0.69 | 0.29 | 0.17 | 0.39 | 0.22 |
| **line 4** | 0.28 | 0.61 | 0.27 | 0.29 | 0.55 | 0.46 |
| **line 5** | 0.30 | 0.31 | 0.70 | 0.35 | 0.27 | 0.32 |
| **line 6** | 0.49 | 0.73 | 0.58 | 0.92 | 1.18 | 0.34 |

TABLE A.4: Standard deviation of Euclidean distances of each pair of landmarks from all three machines (unit: millimetre)

| **participant** | **1** | **2** | **3** | **4** | **5** | **6** |
|---|---|---|---|---|---|---|
| **line 1** | 1.19 | 0.15 | 0.85 | 0.49 | 0.92 | 0.23 |
| **line 2** | 0.71 | 0.72 | 1.09 | 0.46 | 1.21 | 0.40 |
| **line 3** | 0.32 | 0.86 | 0.30 | 0.11 | 0.44 | 0.17 |
| **line 4** | 0.35 | 0.53 | 0.18 | 0.35 | 0.59 | 0.53 |
| **line 5** | 0.37 | 0.34 | 0.74 | 0.45 | 0.25 | 0.18 |
| **line 6** | 0.30 | 0.85 | 0.63 | 1.01 | 1.14 | 0.39 |

TABLE A.5: Standard deviation of Euclidean distances of each pair of landmarks from DI4D® and NCTech® machines (unit: millimetre)

| **participant** | **1** | **2** | **3** | **4** | **5** | **6** |
|---|---|---|---|---|---|---|
| **line 1** | 0.99 | 0.02 | 0.57 | 0.34 | 0.54 | 0.15 |
| **line 2** | 0.17 | 0.46 | 0.80 | 0.44 | 0.33 | 0.55 |
| **line 3** | 0.34 | 0.16 | 0.33 | 0.21 | 0.33 | 0.28 |
| **line 4** | 0.29 | 0.43 | 0.23 | 0.36 | 0.51 | 0.12 |
| **line 5** | 0.11 | 0.38 | 0.66 | 0.28 | 0.27 | 0.34 |
| **line 6** | 0.61 | 0.46 | 0.37 | 1.06 | 1.00 | 0.22 |

TABLE A.6: Standard deviation of Euclidean distances of each pair of landmarks from DI4D® and Artec® machines (unit: millimetre)

| **participant** | **1** | **2** | **3** | **4** | **5** | **sd** |
|---|---|---|---|---|---|---|
| **rmsd** | 2.172 | 1.680 | 1.727 | 1.645 | 2.218 | 0.28 |
| pn | 1.93 | 1.94 | 1.13 | 0.97 | 1.89 | 0.47 |
| acL | 1.80 | 2.67 | 1.48 | 2.55 | 4.61 | 1.21 |
| acR | 2.45 | 1.80 | 1.68 | 2.70 | 1.18 | 0.59 |
| sn | 1.26 | 0.95 | 1.80 | 0.59 | 1.32 | 0.44 |
| n | 1.77 | 0.09 | 2.22 | 0.82 | 1.78 | 0.86 |
| se | 1.67 | 1.14 | 3.32 | 0.55 | 4.90 | 1.79 |
| enL | 2.26 | 1.06 | 1.45 | 2.64 | 1.71 | 0.62 |
| enR | 2.20 | 2.15 | 1.85 | 0.98 | 1.80 | 0.49 |
| exL | 6.19 | 0.66 | 1.07 | 3.21 | 3.12 | 2.18 |
| exR | 6.92 | 1.26 | 2.22 | 3.12 | 2.23 | 2.20 |
| cheekL | 2.03 | 6.23 | 5.85 | 4.52 | 5.01 | 1.65 |
| cheekR | 6.06 | 7.85 | 5.33 | 2.57 | 6.67 | 1.99 |
| cphL | 1.41 | 2.09 | 1.55 | 1.27 | 1.09 | 0.38 |
| cphR | 0.70 | 2.18 | 0.90 | 1.48 | 2.50 | 0.79 |
| ls | 1.11 | 1.05 | 0.93 | 1.61 | 1.31 | 0.26 |
| chL | 1.65 | 1.41 | 1.28 | 2.55 | 1.20 | 0.57 |
| chR | 2.85 | 1.85 | 0.76 | 1.95 | 2.68 | 0.83 |
| st | 2.68 | 4.37 | 1.33 | 0.84 | 1.51 | 1.44 |
| li | 3.33 | 2.91 | 0.80 | 3.51 | 2.33 | 1.09 |
| sl | 3.94 | 1.23 | 3.69 | 1.15 | 1.59 | 1.38 |
| gn | 6.35 | 4.23 | 8.10 | 2.96 | 7.34 | 2.13 |

TABLE A.7: Deviance of each landmark of each participant (unit: millimetre)

# Bibliography


[1] Titus. 3D Laser Scanner Market in 2017 and Growth Prospects During 2018-2024, 2017. URL https://www.envisioninteligence.com/blog/3d-laser-scanner-market-2017-growth-prospects-2018-2024/.

[2] Darryl Motley. Examining the Reverse Engineering Workflow from 3D Scan to CAD, 2017. URL https://gomeasure3d.com/blog/reverse-engineering-workflow-scan-to-cad/.

[3] DAQRI. Depth Cameras for Mobile AR: From Iphones to Wearables and Beyond, 2018.

[4] Kevin Parsons, Eladio Márquez, and R. Albertson. Constraint and opportunity: The genetic basis and evolution of modularity in the cichlid mandible. *The American naturalist*, 179:64–78, 01 2012. doi: 10.1086/663200.

[5] Jan J. Koenderink and Andrea J. van Doorn. Surface shape and curvature scales. *Image and Vision Computing*, 10(8):557–564, 1992. ISSN 02628856. doi: 10.1016/0262-8856(92)90076-F.

[6] Paul H C Eilers, Brian D Marx, and Paul H C Ellers. Flexible Smoothing with B-splines and Penalties. *Statistical Science*, 11(2):89–102, 1996. doi: 10.1002/jwmg.567.

[7] Liberty Vittert, Adrian Bowman, and Stanislav Katina. Statistical models for manifold data with applications to the human face. pages 1–25, 2017. URL http://arxiv.org/abs/1701.07328.

[8] Liberty Vittert. *Facial Shape Analysis*. PhD thesis, 2015. URL http://theses.gla.ac.uk/6669/.

[9] Computer History Museum. 1960: METAL OXIDE SEMICONDUCTOR (MOS) TRANSISTOR DEMONSTRATED, 2019.

[10] Fujifilm. A photographic film producer develops the world's first fully digital camera. URL https://www.fujifilm.com/innovation/achievements/ds-1p/.

[11] TechTerms. Graphics. URL https://techterms.com/definition/graphics.

[12] Serdar Yegulalp. Camera phones: A look back and forward, 2012.

[13] Nicholas J. Wade. Descriptions of visual phenomena from Aristotle to Wheatstone. *Perception*, 25(10):1137–1175, 1996. ISSN 03010066. doi: 10.1068/p251137.

[14] Mostafa Ebrahim. 3D Laser Scanners: History, Applications and Future. *LAP LAMBERT Academic Publishing*, (October):104, 2016. ISSN 0161-8105. doi: 10.13140/2.1.3331.3284. URL https://www.researchgate.net/publication/267037683{_}3D{_}LASER{_}SCANNERS{_}HISTORY{_}APPLICATIONS{_}AND{_}FUTURE.

[15] M. Judith Leo and D. Manimegalai. 3D modeling of human faces- A survey. *TISC 2011 - Proceedings of the 3rd International Conference on Trendz in Information Sciences and Computing*, pages 40–45, 2011. doi: 10.1109/TISC.2011.6169081.

[16] M. M.P.A. Vermeulen, P. C.J.N. Rosielle, and P. H.J. Schellekens. Design of a high-precision 3D-coordinate measuring machine. *CIRP Annals - Manufacturing Technology*, 47(1):447–450, 1998. ISSN 00078506. doi: 10.1016/s0007-8506(07)62871-6.

[17] Chris Boehnen and Patrick Flynn. Accuracy of 3D scanning technologies in a face scanning scenario. *Proceedings of International Conference on 3-D*

*Digital Imaging and Modeling, 3DIM*, pages 310–317, 2005. ISSN 15506185. doi: 10.1109/3DIM.2005.13.

[18] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2020. URL https://www.R-project.org/.

[19] Bowman Adrian and Stanislav Katina. *Tools for the Analysis of Three-Dimensional Surface Images*, 2018. URL https://github.com/adrian-bowman/face3d. R package version 0.1.

[20] Christian Peter Klingenberg. Morphometric integration and modularity in configurations of landmarks: Tools for evaluating a priori hypotheses. *Evolution and Development*, 11(4):405–421, 2009. ISSN 1520541X. doi: 10.1111/j.1525-142X.2009.00347.x.

[21] Chengcai Leng, Hai Zhang, Bo Li, Guorong Cai, Zhao Pei, and Li He. Local Feature Descriptor for Image Matching: A Survey. *IEEE Access*, 7:6424–6434, 2019. ISSN 21693536. doi: 10.1109/ACCESS.2018.2888856.

[22] Hans P. Moravec. Rover Visual Obstacle Avoidance. 2:785–790, 1981.

[23] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. ISSN 09205691. doi: 10.1023/B:VISI.0000029664.99615.94.

[24] Liberty Vittert, Adrian Bowman, and Stanislav Katina. A hierarchical curve-based approach to the analysis of manifold data. *The Annals of Applied Statistics*, 13(4), 2019.

[25] B.N. Delaunay. Sur la sphère vide. *Bull. Acad. Sci. URSS*, 1934(6):793–800, 1934.

[26] Stanislav Katina, Kathryn Mcneil, Ashraf Ayoub, Brendan Guilfoyle, Balvinder Khambay, Paul Siebert, Federico Sukno, Mario Rojas, Liberty Vittert, John Waddington, Paul Whelan, and Adrian Bowman. The definitions of

three-dimensional landmarks on the human face: an interdisciplinary view. *Journal of Anatomy*, 2016.

[27] Jack Goldfeather and Victoria Interrante. A novel cubic-order algorithm for approximating principal direction vectors. 23:45–63, 2004. doi: 10.1145/966131.966134.

[28] Yutaka Ohtake, Alexander Belyaev, and Hans-Peter Seidel. Ridge-valley lines on meshes via implicit surface fitting. *ACM Transactions on Graphics (TOG)*, 23(3):609–612, 2004.

[29] Ian Dryden and Kanti Mardia. *Statistical Shape Analysis*. Wiley, 1998.

[30] Ian Dryden and Kanti Mardia. Multivariate Shape Analysis. *The Indian Journal of Statistics, Series A (1961-2002)*, 55(3):460–480, 1993.

[31] T.F. Cootes, C. J. Taylor, and J. Graham. Training Models of Shape from Sets of Examples. *BMVC92*, pages 9–18, 1992. doi: https://doi.org/10.1007/978-1-4471-3201-1_2.

[32] Alexey Lindo. *Multivariate Methods*. 2019. doi: 10.1002/wics.185.

[33] Richard D. DeVeaux, Adrian W. Bowman, and Adelchi Azzalini. Applied Smoothing Techniques for Data Analysis. *Technometrics*, 1999. ISSN 00401706. doi: 10.2307/1270572.

[34] Rob J Hyndman. ETC5410: Nonparametric smoothing methods. (July), 2008. URL http://www.robjhyndman.com/.

[35] Carl de Boor. A practical guide to splines. In *Applied Mathematical Sciences*, 1978.

[36] Finbarr O'Sullivan. A statistical perspective on III-posed inverse problems. *Statistical Science*, 1(4):502–518, 1986. ISSN 08834237. doi: 10.1214/ss/1177013525.

[37] Paul H.C. Eilers, Iain D. Currie, and Maria Durbán. Fast and compact smoothing on large multidimensional grids. *Computational Statistics and*

*Data Analysis*, 50(1 SPEC. ISS.):61–76, 2006. ISSN 01679473. doi: 10.1016/j.csda.2004.07.008.

[38] Yulan Guo, Mohammed Bennamoun, Ferdous Sohel, Min Lu, and Jianwei Wan. 3D object recognition in cluttered scenes with local surface features: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36 (11):2270–2287, 2014. ISSN 01628828. doi: 10.1109/TPAMI.2014.2316828.

[39] Joel Daniels, Linh K. Ha, Tilo Ochotta, and Cláudio T. Silva. Robust smooth feature extraction from point clouds. *Proceedings - IEEE International Conference on Shape Modeling and Applications 2007, SMI'07*, (June):123–133, 2007. doi: 10.1109/SMI.2007.32.

[40] Yangbin Lin, Cheng Wang, Bili Chen, Dawei Zai, and Jonathan Li. Facet Segmentation-Based Line Segment Extraction for Large-Scale Point Clouds. *IEEE Transactions on Geoscience and Remote Sensing*, 55(9):4839–4854, 2017. ISSN 01962892. doi: 10.1109/TGRS.2016.2639025.

[41] Klaus Hildebrandt, Konrad Polthier, and Max Wardetzky. Smooth feature lines on surface meshes. *ACM International Conference Proceeding Series; Vol. 255*, page 85, 2005. ISSN 1727-8384. URL http://portal.acm.org/citation.cfm?id=1281935.

[42] Weini Zhang, Linwei Chen, Zhangyue Xiong, Yu Zang, Jonathan Li, and Lei Zhao. Large-scale point cloud contour extraction via 3D guided multi-conditional generative adversarial network. *ISPRS Journal of Photogrammetry and Remote Sensing*, 164(December 2019):97–105, 2020. ISSN 09242716. doi: 10.1016/j.isprsjprs.2020.04.003. URL https://doi.org/10.1016/j.isprsjprs.2020.04.003.

[43] Rhodri H. Davies, Tim F. Cootes, and Chris J. Taylor. A minimum description length approach to statistical shape modelling. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2082(5):50–63, 2001. ISSN 16113349.

[44] James S. W. Wong. On vector representation of rigid body rotation. *Mathematics Magazine*, 41(1):28–29, 1968. ISSN 0025570X, 19300980. URL http://www.jstor.org/stable/2687956.

[45] Brian D. Segal, Michael R. Elliott, Thomas Braun, and Hui Jiang. P-splines with an l1 penalty for repeated measures. pages 1–44, 2017. URL http://arxiv.org/abs/1707.08933.

[46] Ariel Shamir. A survey on mesh segmentation techniques. *Computer Graphics Forum*, 27(6):1539–1556, 2008. ISSN 14678659. doi: 10.1111/j.1467-8659.2007.01103.x.

[47] Panagiotis Theologou, Ioannis Pratikakis, and Theoharis Theoharis. A comprehensive overview of methodologies and performance evaluation frameworks in 3D mesh segmentation. *Computer Vision and Image Understanding*, 135: 49–82, 2015. ISSN 1090235X. doi: 10.1016/j.cviu.2014.12.008. URL http://dx.doi.org/10.1016/j.cviu.2014.12.008.

[48] Adrien Maglo, Guillaume Lavoué, Florent Dupont, and Céline Hudelot. 3D mesh compression: Survey, comparisons, and emerging trends. *ACM Computing Surveys*, 47(3), 2015. ISSN 15577341. doi: 10.1145/2693443.

[49] Rui S.V. Rodrigues, José F.M. Morgado, and Abel J.P. Gomes. A contour-based segmentation algorithm for triangle meshes in 3D space. *Computers and Graphics (Pergamon)*, 49:24–35, 2015. ISSN 00978493. doi: 10.1016/j.cag.2015.04.003. URL http://dx.doi.org/10.1016/j.cag.2015.04.003.

[50] Panagiotis Theologou, Ioannis Pratikakis, Senior Member, and Theoharis Theoharis. Unsupervised Spectral Mesh Segmentation Driven by Heterogeneous Graphs. 39(2):397–410, 2017.

[51] Xue Jiao, Tieru Wu, and Xuzhou Qin. Mesh segmentation by combining mesh saliency with spectral clustering. *Journal of Computational and Applied Mathematics*, 329:134–146, 2018. ISSN 03770427. doi: 10.1016/j.cam.2017.05.007. URL http://dx.doi.org/10.1016/j.cam.2017.05.007.

[52] Markus Schoeler, Jeremie Papon, and Florentin Wörgötter. Constrained planar cuts - Object partitioning for point clouds. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 07-12-June:5207–5215, 2015. ISSN 10636919. doi: 10.1109/CVPR.2015.7299157.

[53] Rodrigo Minetto, Neri Volpato, Jorge Stolfi, Rodrigo M.M.H. Gregori, and Murilo V.G. da Silva. An optimal algorithm for 3D triangle mesh slicing. *CAD Computer Aided Design*, 92:1–10, 2017. ISSN 00104485. doi: 10.1016/j.cad.2017.07.001. URL http://dx.doi.org/10.1016/j.cad.2017.07.001.

[54] Andrew Nealen, Takeo Igarashi, Olga Sorkine, and Marc Alexa. FiberMesh: Designing Freeform Surfaces with 3D Curves. *ACM Transactions on Graphics*, 26(3):41, 2007. ISSN 07300301. doi: 10.1145/1239451.1239492.

[55] Vitaly Surazhsky and Craig Gotsman. Explicit Surface Remeshing. *Eurographics Symposium on Geometry Processing*, (June 2003):1–11, 2003.

[56] Kirsty McWhinnie. *Exploring Mechanisms of Adaptive Divergence in an Exemplar Adaptive Radiation*. PhD thesis, 2020.

[57] Christian Peter Klingenberg. Analyzing fluctuating asymmetry with geometric morphometrics: Concepts, methods, and applications, 2015. ISSN 20738994.

[58] Mitchum T. Bock and Adrian W. Bowman. On the measurement and analysis of asymmetry with applications to facial modelling. *Journal of the Royal Statistical Society. Series C: Applied Statistics*, 55(1):77–91, 2006. ISSN 00359254. doi: 10.1111/j.1467-9876.2005.00532.x.

[59] Miriam Zelditch, Donald Swiderski, H. Sheets, and William Fink. *Geometric Morphometrics for Biologists*. 2004. ISBN 9780127784601. doi: 10.1016/B978-0-12-778460-1.X5000-5.

[60] Miriam Leah Zelditch, Donald L Swiderski, and H David Sheets. A Practical Companion to Geometric Morphometrics for Biologists: Running analyses in freely-available software. (1):1–233, 2012.

[61] Christian Peter Klingenberg, Marta Barluenga, and Axel Meyer. Shape Analysis of Symmetric Structures : Quantifying Variation among Individuals and Asymmetry Author ( s ): Christian Peter Klingenberg , Marta Barluenga and Axel Meyer Published by : Society for the Study of Evolution Stable URL : http://www.jstor.org/s. *Evolution*, 56(10):1909–1920, 2002. ISSN 0014-3820. doi: 10.1554/0014-3820(2002)056[1909:SAOSSQ]2.0.CO;2.

[62] Philipp Mitteroecker, Philipp Gunz, Sonja Windhager, and Katrin Schaefer. A brief review of shape, form, and allometry in geometric morphometrics, with applications to human facial morphology. *Hystrix*, 24(1):59–66, 2013. ISSN 03941914. doi: 10.4404/hystrix-24.1-6369.

[63] James Miller. Shape curve analysis using curvature. *University of Glasgow*, 2009.

[64] Alois Kneip and James Ramsay. Combining registration and fitting for functional models. *Journal of the American Statistical Association*, 103(483): 1155–1165, 2008.

[65] Donatello Telesca and Lurdes Y T Inoue. Bayesian Hierarchical Curve Registration. 1459(2008), 2008. doi: 10.1198/016214507000001139.

[66] Wen Cheng, Ian L Dryden, and Xianzheng Huang. Bayesian Registration of Functions and Curves. (2):447–475, 2016. doi: 10.1214/15-BA957.

[67] Maik Keller, Damien Lefloch, Martin Lambers, Tim Weyrich, and Andreas Kolb. Real-time 3D Reconstruction in Dynamic Scenes using Point-based Fusion. pages 1–8, 2013. doi: 10.1109/3DV.2013.9.

[68] Damien Lefloch, Markus Kluge, Hamed Sarbolandi, Tim Weyrich, and Andreas Kolb. Comprehensive Use of Curvature for Robust and Accurate Online Surface Reconstruction. 39(12):2349–2365, 2017.

[69] Dae Wook, Dae Wook Kim, Byoungchang Kim, Chunyu Zhao, Changjin Oh, and H James. Algorithms for surface reconstruction from curvature data for freeform aspherics. (September 2013), 2013. doi: 10.1117/12.2024285.

[70] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deep convolutional network cascade for facial point detection. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3476–3483, 2013. ISBN 978-0-7695-4989-7. doi: 10.1109/CVPR.2013.446.

[71] Weihong Deng, Yuke Fang, Zhenqi Xu, and Jiani Hu. Neurocomputing Facial landmark localization by enhanced convolutional neural network. *Neurocomputing*, 273:222–229, 2018. ISSN 0925-2312. doi: 10.1016/j.neucom.2017.07.052. URL https://doi.org/10.1016/j.neucom.2017.07.052.

[72] Nannan Wang, Xinbo Gao, Dacheng Tao, Heng Yang, and Xuelong Li. Neurocomputing Facial feature point detection : A comprehensive survey. 275: 50–65, 2018. doi: 10.1016/j.neucom.2017.05.013.

[73] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active Appearance Models. *Proc. European Conference on Computer Vision (ICCV)*, 2:484–498, 1998. ISSN 0162-8828. doi: 10.1109/34.927467.

[74] Guosheng Hu, Xiaojiang Peng, Yongxin Yang, Timothy M Hospedales, and Jakob Verbeek. Frankenstein : Learning Deep Face Representations Using Small Data. 27(1):293–303, 2018.

[75] Hongwen Zhang, Qi Li, Zhenan Sun, and Yunfan Liu. Combining Data-driven and Model-driven Methods for Robust Facial Landmark Detection. 6013(c): 1–14, 2018. doi: 10.1109/TIFS.2018.2800901.

[76] Daniel Alberto Molinari. *Spatiotemporal Modelling of Groundwater Contaminants by*. PhD thesis, 2014. URL http://theses.gla.ac.uk/5873/.

[77] Brandon RichardWebster, So Yon Kwon, Christopher Clarizio, Samuel E. Anthony, and Walter J. Scheirer. Visual psychophysics for making face recognition algorithms more explainable. *Lecture Notes in Computer Science*, 11219 LNCS:263–281, 2018. ISSN 16113349. doi: 10.1007/978-3-030-01267-0_16.