3D Facial Feature Extraction and Recognition

Sokyna M. S. Al-Qatawneh

PhD

2010

3D Facial Feature Extraction and Recognition

An investigation of 3D face recognition: correction and normalisation of the facial data, extraction of facial features and classification using machine learning techniques

Sokyna M. S. Al-Qatawneh, MSc
A thesis submitted for the degree of
Doctor of Philosophy

School of Computing, Informatics & Media

University of Bradford

2010

To

My mother and father,
Your unlimited love and unconditional support are
among my most cherished memories

Abstract

Face recognition research using automatic or semi-automatic techniques has emerged over the last two decades. One reason for growing interest in this topic is the wide range of possible applications for face recognition systems. Another reason is the emergence of affordable hardware, supporting digital photography and video, which have made the acquisition of high-quality and high resolution 2D images much more ubiquitous. However, 2D recognition systems are sensitive to subject pose and illumination variations and 3D face recognition which is not directly affected by such environmental changes, could be used alone, or in combination with 2D recognition.

Recently with the development of more affordable 3D acquisition systems and the availability of 3D face databases, 3D face recognition has been attracting interest to tackle the limitations in performance of most existing 2D systems. In this research, we introduce a robust automated 3D Face recognition system that implements 3D data of faces with different facial expressions, hair, shoulders, clothing, etc., extracts features for discrimination and uses machine learning techniques to make the final decision.

A novel system for automatic processing for 3D facial data has been implemented using multi stage architecture; in a pre-processing and registration stage the data was standardized, spikes were removed, holes were filled and the face area was extracted. Then the nose region, which is

relatively more rigid than other facial regions in an anatomical sense, was automatically located and analysed by computing the precise location of the symmetry plane. Then useful facial features and a set of effective 3D curves were extracted. Finally, the recognition and matching stage was implemented by using cascade correlation neural networks and support vector machine for classification, and the nearest neighbour algorithms for matching.

It is worth noting that the FRGC data set is the most challenging data set available supporting research on 3D face recognition and machine learning techniques are widely recognised as appropriate and efficient classification methods.

Acknowledgment

I would like to begin a limitless praise to “Almighty Allah”, who bestowed me the knowledge means and approaches towards the accomplishment of the present research.

I am heartily thankful to my principal supervisor Dr. Stanly Ipson, whose constant encouragement, guidance and considerable support from the initial to the final stage enabled me to develop an understanding of the subject. Dr. Ipson is a supervisor, who truly made a difference in my life, I appreciate his vast knowledge and skill in many areas, whose expertise, understanding, and patience, added considerably to my graduate experience, and without him the completion of this work could not have been achieved. Beside of being an excellent supervisor, Dr. Ipson encouraging me by his personal kindness, and he always was available when I needed his advises.

I also want to express deep gratitude and thanks to my second supervisor, Dr. Rami Qahwaji, whose guidance, keen interest and continuous efforts enabled me to complete this work. Dr. Qahwaji has offered much advice and insight throughout my research. I also wish to extend my thanks and appreciation to Prof. Hassan Ugail for his kind cooperation during my work.

I owe my deepest gratitude and special thanks to Mr Nadhmi Auchi and Mrs Maha Al Fakier for their genuine support which enabled me to undertake this work.

Besides, I offer my regards and blessings to all of my friends and colleagues who supported me in any respect during the completion of the degree. Furthermore, I would like to thank all the staff at SCIM who helped me during my three years study especially Miss Rona Wilson and Mr Mark Tympalski.

Finally, I dedicated my thesis to my parents who fill my life with all affection one can wish, and my most sincere appreciation goes to my brothers and sisters especially to Khirat for the tremendous amount of support. You have a very special place in my heart. I also would like to remember Abdel-Raheem; you will be always in my prayers.

Table of Contents

List of Figures

List of Tables

LIST OF ABBREVIATIONS

AAM	Active Appearance Model
AFM	Annotated Face Model
AR	Aleix Martinez and Robert Benavente face database
AUs	Action Units
BEE	Biometric Experimentation Environment
CCNN	Cascade Correlation Neural Network
CMU-PIE	Carnegie Mellon University- Pose, Illumination, and Expression
Cov	Covariance matrix
DCT	Discrete Cosine Transform
EER	Equal Error Rate
EGI	Extended Gaussian Images
EHMMs	Embedded Hidden Markov Models
FACS	Facial Action Coding System
FAR	False Acceptance Rate
FERET	Facial Evaluation REcognition Test
FLD	Fisher's Linear Discriminant
FMTD	Fast Matching on Triangulated Domains
FN	False Negative
FP	False Positive
FR	Face Recognition
FRGC	Face Recognition Grand Challenge
FRR	False Rejection Rate
FRVT	Facial Recognition Vendor Test
GD	Geodic Distance

HCInt	High Computational Intensity test
HMM	Hidden Markov Models
ICA	Independent Component Analysis
ICP	Iterative Closest Point algorithm
KLT	Karhunen Loeve Transform
KNN	K-Nearest Neighbour method
KPCA	Kernel Principle Component Analysis
LDA	Linear Discriminant Analysis
LIBSVM	Library of Support Vector Machine
MDL	Minimum Description Length
MDS	Multi Dimensional Scaling
NNs	Neural Networks
OBJ	OBJect files
PCA	Principle Component Analysis
PRISM	Partnership for Research In Spatial Modelling laboratory
RBF	Radial Basis Function
ROC	Receiver Operating Characteristic curves
SD	Standard Deviation
SFS	Shape From Shading
SVM	Support Vector Machine
TN	True Negative
TP	True Positive
VRML	Virtual Reality Modelling Language files

CHAPTER ONE

1. INTRODUCTION

1.1 General background

Recent developments in computer technology and the call for better security applications have raised interest in biometrics. A biometric is a physical property; it cannot be forgotten or mislaid like a password, and it has the potential to identify a person in very different settings. Be it for purposes of security or for human–computer interaction, there are many applications for robust biometrics. Human recognition systems fall into two categories: verification and identification. In the former, the person claims to be someone, and this claim is verified if the biometric provided is sufficiently close to the data stored for that person. This is a one-to-one comparison against the template of the person whose identity is being claimed. In the identification problem, by contrast, a match is sought in a database which could be huge. In other words, it is a one-to-many comparison which compares a person against all the templates in a database to determine the identity of the query person. A human face is perhaps the most easily acquired biometric identifier. The less intrusive nature of the process of acquiring a face image is the primary reason why face recognition based systems are preferred over other biometric systems. Signatures, handwriting, face images and fingerprints are biometrics, of long standing, used in the verification or authentication of documents. More recently, voices, gaits, retinas and iris scans, handprints, and 3D face information have all been considered as biometric identifiers. Each of these

has different merits, and applicability [1]. Table 1-1 presents a summary of these biometrics and their relative strengths [1]. Although 2D and 3D face recognition are not as accurate as iris scans, their ease of use and lower cost makes them a preferable choice in some scenarios.

Table 1-1 Biometrics and their relative strengths

	Accuracy	Cost	Privacy	Integrity	Ease of use	Development	Hiding identity	Faking identity
Iris	√		√				√	√
Retina	√		√				√	√
Hand					√	√	√	√
Signature					√	√		
Voice		√		√	√	√		
Gait			√					
2D face		√	√	√	√	√		
3D face	√		√	√	√	√		√

1.2 2D and 3D Face Identification

Automatic identification of human faces is a very challenging research topic, which has gained much attention in recent years. Until recently, most research in face recognition focused on 2D intensity or colour images of faces [2]. The primary reason for this bias towards 2D face recognition was because they are easy to acquire. Furthermore, quite

useable results have been achieved using 2D face images in constrained environments [2] where illumination is assumed to be constant. Numerous techniques have been proposed, including Eigenfaces [3], Fisherfaces [4], elastic bunch graph matching [5] and the Kernel method [6]. Great strides have been made in recent years, and the existing methods usually work very well under carefully-controlled conditions. However, a number of recent studies [7-9] have shown that the unconstrained recognition of faces from still images is a difficult problem, because illumination, pose and expression change between images of the same person create great statistical differences and the identity of the face itself becomes overshadowed by these factors. 3D face recognition is invariant to the environment changes (assuming 3D capture can handle the variance in environmental conditions) which means the face recognition in 3D is superior to 2D in that it has the potential to overcome feature localization, pose and illumination problems, and it can be used in conjunction with 2D systems. Recently with the development of relatively low cost 3D acquisition systems, 3D face recognition has attracted more and more interest for tackling the degradation in performance in most existing 2D systems caused by unconstrained capture. However, despite much effort, such as modelling illumination [10], using symmetric shape-from-shading (SFS) based view synthesis [11] and employing 3D morphable models to correct the pose [12], robust face recognition is still an uphill task. Recent advances in 3D modelling and digitizing techniques have made the acquisition of 3D human face data more feasible [13, 14].

The applications for facial recognition are varied and vast. For example, the face recognition technology could be used as a security measure at ATM's; instead of using a

bank card or personal identification number, the ATM would capture an image of a face, and compare it to the face photo in the bank database to confirm the users identity. In addition, it could be used for document control (e.g. digital chip in passports, driver's licenses). As well as it would be useful for computer security (user access verification), besides that it could be used to prevent voter fraud (election accuracy). Moreover, it can be used for time and attendance applications (entry and exit verification) and in computer games.

In both 2D and 3D face recognition systems, alignment (registration) of the query and the template is generally necessary [15]. Registration based on feature point correspondence is one of the most popular methods [16]. To make a face recognition system fully automatic, robust facial feature extraction is one of the crucial steps. Facial features can be of different types: region [17, 18], landmark [19, 20], and contour [21, 22]. Generally, landmarks provide more accurate and consistent representation for alignment purposes than region-based features and have a lower complexity and computational burden than contour feature extraction.

In order to assess how well proposed methods work when dealing with face recognition issues, many research groups generate multi-view face databases which provide as many variations as possible in their images. FERET[9], CMU-PIE [23], and AR Faces [24] database represent one of the most popular 2D face image database collections. Each database is designed to address specific challenges covering a wide range of scenarios. By contrast, there are very few 3D face model databases and most of these contain relatively

little data. The 3D_RMA (clouds of points)[25], University of York 3D Face Database (Range images)[26], and GavabDB (Tri-Mesh)[27] are examples of 3D face model databases. Where 3D_RMA database was based on structured-light, being constructed using a camera and a projector, and generating the 3D coordinates of the surface points with a high precision. Glasses and dark parts of faces could not be captured. It contains 120 individuals captured in two different sessions, separated by 2 months. Digitisations consisted in three shots grabbing different and limited orientations of the heads. The whole individuals belonged to the same Caucasian race. While 3D face database of York University [15] contained a reduced set of face images per subject. This database has images corresponding to 97 individuals. It contains 10 captures per individual including different poses. However, only two of these views of each individual present light facial expressions (happiness and frown), and one presents face occlusion. GavabDB contains 3D face images of 61 people including nine images for each person. The whole set of individuals are Caucasian and most of them are aged between 18 to 40 years. And there are systematic variations over the pose and facial expression of each person. As it can be observed, there are not many varied changes in the image modifications among the different images of each individual in the databases. Some of them offer variations related to some aspects but not to others. However, when some database has a certain richness of systematic changes, the range of variation is very limited and does not consider the gender, the ethnic, and pose or expression variation.

The Face Recognition Grand Challenge (FRGC)[28], which includes a substantial amount of 3D face data with corresponding 2D texture images is an effort to promote and advance

face recognition designed to support existing face recognition systems. The FRGC consists of a set of progressively more difficult challenging problems. Each challenge problem consisted of a data set of facial images and a defined set of experiments. One of the striking characteristics of this database is the size of the FRGC in terms of data, with 50000 recordings. Another aspect is the complexity of the FRGC in terms of three image modes: high resolution still images, 3D images (range images), and multiple images of a person. This initiative is likely to intensify research into 3D face recognition and greatly improve the ability to compare and contrast various 3D face recognition methods.

Range image based 3D face recognition has been demonstrated to be effective in enhancing face recognition accuracy [29]. Since each range image only provides face data from a single viewpoint, instead of the full 3D views illustrated in Figure 1.1, the main advantages of 3D range data are explicit representation of 3D shape and association of face shape with real size. However, using the whole facial data may not be feasible considering the large computation and hardware capacity needed [30]. Therefore, it is advisable to use 3D features with fewer parameters, such as, distinctive landmarks and some effective profiles.

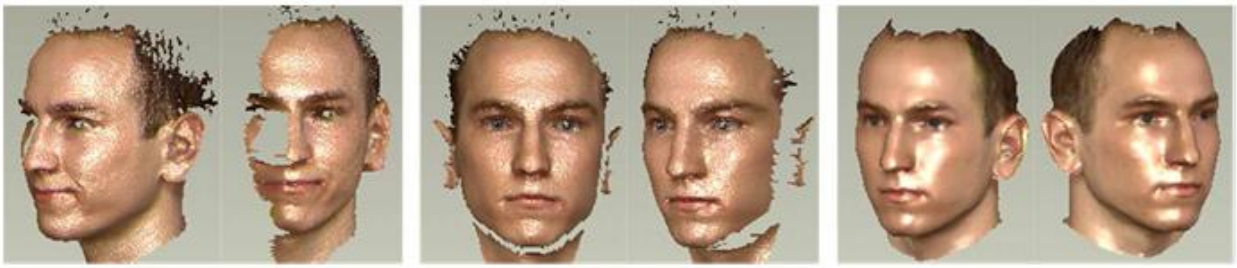


Figure 1-1 3D model and 2.5D scans (a) a 2.5D scan with large pose changes; notice the missing data due to pose change. (b) A 2.5D frontal scan. (c) A 3D face model. Each pair is the same scan/model but displayed from different viewpoints according to [31].

Face recognition algorithm performance is typically characterized by correct identification rate (the accuracy), True Positive Rate (TPR), False Acceptance Rate (FAR) and False Rejection Rate (FRR) under closed-world assumptions. However the Facial Evaluation Recognition Test (FERET) strategy [9] and the Facial Recognition Vendor Test (FRVT) [7] give useful suggestions for the standardization of the testing protocol to minimize the false alarm rate. However, the number of common benchmark databases used to test existing algorithms has risen in the last decade, and FERET is an example of this databases. In contrast, the main goal of the FRVT is the capability assessments of commercially available facial recognition systems with respect to changes in expression, illumination, pose and time delay.

1.3 Face Recognition challenges

Although a great deal of research has been dedicated to 2D and 3D face recognition systems and major advances have been achieved with some applications reporting high recognitions rate in controlled environments, the problem of recognizing faces is still a challenging one due to:

- *Variant head poses*. While this might be critical with 2D intensity images, it is less of a challenge for applications utilizing 3D facial data.
- *Illumination.* This is a real challenge for 2D system [32] unlike 3D recognition systems where it is reported that this problem is overcome because of the utilization of 3D facial data[33] .
- *Facial expression variation.* Although several approaches have been developed to address this problem such as model based face recognition techniques, and morphable 2D/3D models [34-37], this is still a challenging problem for both 2D and 3D image recognition systems. It has been estimated that the face could generate up to 55,000 different actions [38].
- *Occlusion.* Accessories (e.g. sun glasses) or other objects (beards) can obscure data.
- *Ageing.* This is clearly one of the challenges in any face recognition system, as it causes significant alterations of facial appearance of individuals [39].

1.4 Research Aims and Objectives

In this research, the aim is to work towards a fully automatic 3D face recognition system using range data from the FRGC *Ver.2.0* 3D dataset [] and introduce features to make the recognition robust to facial expression and also efficient. Figure 1.2 shows a block diagram of the proposed face recognition system.

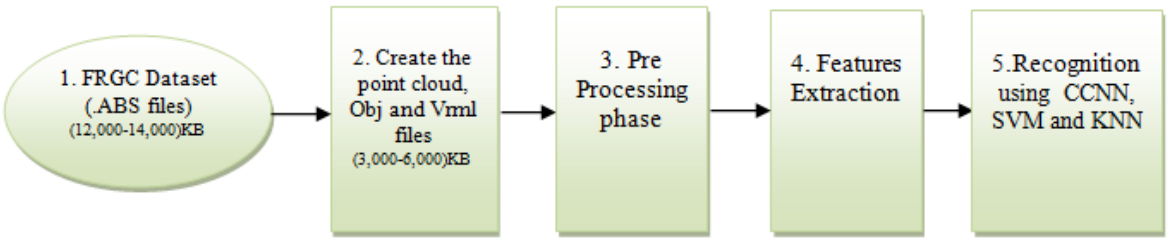


Figure 1-2 Main block diagram of the face recognition system.

More specifically, the objectives of this research outlined as follows:

- Propose a fully automated 2.5D facial feature extraction and classification system based on some machine learning techniques, and investigate the impact of different registration techniques.
- Implement a compact representation of facial data by reconstructing a 3D triangulated human face containing the coordinate and connectivity information to simplify the process of recognition, and then perform robust pre-processing techniques to tackle some of the 3D image problems and

reduce the size of input data in order to make recognition and classification of 3D faces more reliable and effective.

- Feature extraction is the most difficult problem in a face recognition system and can be considered as the heart of the system. In other words, if a system is successful in representing the face image with significant features, the whole recognition problem is likely to be largely solved and hence our third objective is to extract robust and distinguishable 3D face features by developing a novel method to automatically determine a symmetry profile for face data. This is undertaken by computing the intersection between the symmetry plane (found by an automatic search) and the facial mesh, resulting in a planar curve that accurately represents the symmetry profile.
- Determine the most effective feature points along the symmetry profile (e.g. tip of nose, nose bridge and bottom of nose) in order to compute other facial features (e.g. inner eye corners), which can then be utilized to locate the central region of the face and extract a set of profiles from that region. These feature points and profiles can be used for recognition purposes.
- Propose an efficient classification experiment by using machine learning techniques, which are widely recognised as appropriate and efficient validation methods, to validate our approach of using extracted features proposed in previous work (chapters 3 and 4).

1.5 Outline of the Thesis

This thesis is organized as follows:

- Chapter 2 provides an extended literature review for 2D and 3D face recognition techniques. In addition, the FRGC database is presented in detail with clear outline of the machine learning algorithms.
- Chapter 3 presents the 2.5D face recognition system based on range data. The interpretation of 2.5D data and the extraction of the facial region as well as the pre-processing of the face data and the feature extraction are explained in this chapter.
- Chapter 4 describes the pre processing and facial features extraction for a 3D facial mesh. Also, it gives a numerical representation for these features in order to use them for recognition and classification purpose. These features include a set of profiles and distinguishable points extracted from the central region of the face.
- Chapter 5 provides a practical implementation and an evaluation of the proposed 3D face recognition and classification systems using machine learning algorithms. It compares the performances of some learning algorithms: Cascade Correlation Neural Networks (CCNNs), Support Vector Machines (SVMs) and the K-Nearest Neighbours algorithm (KNN).

- Concluding remarks based on the proposed work and recommendations for future work are presented in Chapter 6.

1.6 Publications and papers under review

- Qatawnah S, Ipson S, Qahwaji R and Ugail H (2008): "3D Face Recognition Based on Machine Learning", *Eighth IASTED International Conference on Visualization, Imaging and Image Processing (VIIP 2008), pp. 362-366*, Palma de Mallorca, Spain.
- Qatawnah S., Ipson S. and Qahwaji R. (2011): "3D Face Identification Based On The Symmetry Profile Analysis On Nose Region", *The Twelfth IASTED International Conference on Computer Graphics and Imaging (CGIM 2011),* Innsbruck, Austria. (Under review)
- Qatawnah S., Ipson S. and Qahwaji R. (2011): "3D facial features extraction under varying facial expression and classification using machine learning techniques" The Eighth IASTED International Conference on Signal Processing, Pattern Recognition, and Applications (SPPRA 2011), Innsbruck, Austria. (Under review)

CHAPTER TWO

2. LITERATURE REVIEW

Although major advances have been made over the last decade in the automatic face recognition field such as the development of graphics techniques for manipulating images of faces, and the increased number of face recognition systems which has achieved a high recognition rates under controlled conditions, the problem is still considered hard to solve and continues to attract substantial research input from a number of different disciplines and areas including pattern recognition, machine learning and computer graphics. Evaluations such as the FRVT [32, 40] demonstrate that the current state of the art in face recognition is not yet sufficient for the more demanding applications. However, among many biometric techniques for human identification, face recognition is still considered of great importance due to its low requirements for setup and equipment, so there is significant potential demand for improved performance in face recognition.

In the following subsections, very common approaches and terminologies in the area of face recognition would be often mentioned. In Section 2.1, some of 2D image based face recognition methodologies are described and many of 3D face recognition approaches are given in Section 2.2. An overview of the Face Recognition Grand Challenge (FRGC) database is presented in Section 2.3 and in section 2.4 some of machine learning techniques are discussed.

2.1 2D face recognition systems

Research on face recognition (FR) based on 2D intensity images, has been conducted for more than three decades. As a result the current status of 2D FR is well established and advanced compared with the more recent 3D face recognition. A wide range of 2D recognition algorithms, commercial applications and image databases for testing and evaluation purposes are available. A comprehensive online source of information is available at the face recognition home page[1]. Many 2D image based face recognition methodologies are available in the literature, which can be broadly categorized as either appearance based or model based approaches.

2.1.1 Appearance Based Approaches

In this section we review methods of face recognition that use image subspace projection in order to compare face images by calculating image separation in a reduced dimensionality space. The appearance based face recognition techniques depend solely on 2D intensity images without use of 3D models of the face. Thus, the face is actually represented in terms of a relatively small number of basis images such as Eigenfaces. Among many existing linear approaches, Principle Component Analysis (PCA)[41], Independent Component Analysis (ICA)[42, 43] and Linear Discriminant Analysis (LDA)[44], are considered powerful techniques in the field of face recognition. PCA has become a standard to which other systems are often compared, as well as often being used for preliminary dimensional reduction in other methods of face recognition. It finds a set of basis vectors that are

[1] http://www.face-rec.org

uncorrelated to best represent the sample images with a small number of vectors, and so the comparison and recognition tasks are performed in a lower dimension space called the feature or face space. In principle, statistical approaches generally do not extract local facial features but instead follow a global approach to represent faces according to their whole appearance.

Turk and Pentland [41] were the first to apply PCA to face recognition. PCA extracts statistically significant information defined as eigenvectors of the covariance matrix of the set of training images, which are then used to project test faces onto a lower dimension space (face space) for reconstruction and recognition purposes. To identify a test image, it is projected onto the face space to obtain the corresponding set of weights. By comparing the weights for the test image with the set of weights of the faces in the training set, the face in the test image can be identified. The transformation onto the face space is done based on the eigenvectors corresponding to the largest eigenvalues. On the other hand, eigenvectors corresponding to the smallest eigenvalues are usually discarded or considered as noise, and not taken into consideration during the identification process. PCA is considered the most descriptive representation in terms of least square reconstruction errors. In addition it is easy to implement, yet it is not the most discriminative for class separation. However, several extensions have been made to deal with pose changes[45] and probabilistic subspace[46].

While PCA utilizes first and second-order statistics, ICA [43] explores higher order statistics. ICA has been successfully implemented in the area of face recognition and face feature extraction. However one of its drawbacks is its iterative and time-consuming nature

compared to PCA. In addition, convergence is sometimes difficult to achieve. Bartlett et al. [47] extended the work of [48] by implementing a fast ICA algorithm to reduce computation time.

Unlike the unsupervised methods PCA and ICA that construct the face space without using the face class information, the supervised LDA has also been proposed for face recognition. In LDA the goal is to find an optimal way to represent the face vector space to maximize the discrimination between different face classes. This is achieved by defining classes with different statistics, and hence allocating different classes for images in the learning set. Exploiting class variation information tends to give better recognition rates as illustrated by Belhumeur et al [44].

Various other linear subspace analysis techniques have been developed based on the above techniques and used in face recognition. Yang et al. [49] treated an image as a 2D matrix and developed a two dimensional PCA algorithm. LDA has been generalized into 2D Fisher Discriminant Analysis and applied to face recognition by Kong et al [50]. While linear subspace analysis techniques are an approximation of the non-linear face recognition manifold, other direct non-linear schemes have been also used in the context of face recognition including Neural Networks (NN) and Kernel Principle Component Analysis (KPCA), which applies nonlinear mapping from the input space to the feature space by a suitably chosen kernel functions and corresponding parameters [6, 51]. Small 2D Appearance based image face recognition system are still encountering difficulties due to the complex facial variations resulting from the inherent problems of face recognition (e.g. lighting, facial poses and facial expression). One proposed solution to this limitation is

using new methodologies to synthesise other face images from existing training sampling which simulate various facial expression and conditions [52].

2.1.2 Model Based Approaches

An alternative to global statistical approaches which represent faces according to their whole appearance, model based techniques tend to construct models of human faces that capture facial variations. A model construction of a face is carried out by localizing certain facial feature (eyes corners or centres, nose trails, noses tip, etc.) and deriving metric measurements (parameters) between these elements. For recognition purposes a set of derived parameters are matched against the parameters of a known faces 'gallery'. An early example of this approach was proposed by Kanade [53] where he utilized angles and distances between facial features such as eyes, nose and mouth to represent faces. Although some advances have been made in the automatic localization of facial points [54, 55], it is still a challenging problem in the area of 2D images based face recognition due to inherent problems(e.g. lightening, facial poses, and facial expression), especially in cluttered scenes and occluded images.

Coots et al.[21, 36] developed 2D morphable face models to address facial variations by analyzing statistical variations of shape and texture respectively and utilized these for face recognition purposes. Blanz and Vetter [56] proposed more advanced model based face recognition techniques by using 3D morphable face models to capture the true 3D geometry of the facial surface and used it along with facial appearance information for carrying out

comparisons between faces which appears to be a better representation of human faces, and give better identification rates than the 2D morphable face models.

In almost all model based schemes three main requirements need to be achieved, first of all the model construction, second fitting the model to give face image by fine-tuning the model and work out the parameters, and finally using these parameters to calculate the similarity between the query face and the faces in the gallery. In addressing these requirements various methodologies have been used. For example, faces have been represented as labelled graphs by Wiskott et al.[5], with graph nodes positioned at certain facial feature points, while recognition was carried out based on Gabor wavelet graph based similarity function.

Other approaches for constructing models have been used such as the Active Appearance Model (AAM)[21, 36, 57], where the model incorporates both shape variation and appearance variation to synthesize face images. In order to carry out comparisons a matching between a query image and the synthesized model is performed which results in a set of parameter values which best fit the image to the model.

2.1.3 Advantages and disadvantages of 2D image based systems

In addition to the above two main categories of 2D image based recognition techniques, various other approaches have been used to address the problem, including Local Feature Analysis[58], Component based schemes [59] and Statistical models based schemes such as Hidden Markov Models (HMM)[60] and Gaussian Mixture Models [61]. The availability of 2D image based algorithms; the relatively-low prices of equipments needed for setting

up such systems, and the major advances that have been achieved in this area, in addition to the legacy of 2D images database used at airports, by police, etc. have made 2D image based application widely available and popular at both academic and commercial levels with a number of popular commercial application such as Identix [62] and Cognitex [63].

Despite the above advantages, pose variations, lighting conditions and facial expression variations are still considered major factors that degrade the performance of the current 2D face recognition systems[64, 65]. Not only this, but it has been demonstrated that the performance of some of the very popular commercial 2D image based applications is very much dependant on the size of the database, and the performance time is as important as accuracy for some face recognition applications [64].

2.2 3D face recognition techniques

Although rather limited in comparison with the wealth of research applied to 2D face recognition, there are a number of investigations that have demonstrated how geometric facial structure can be used to aid recognition. 3D face recognition has attracted more attention in recent years due to two major factors. Firstly, the inherent problems with 2D face recognition systems which appear to be very sensitive to facial pose variation, variant facial expressions, lighting and illumination. On the other hand, for example, Xu et al. [66] compared 2D intensity image against depth images and concluded that depth map give a more robust face representation, because intensity images are significantly affected by

changes in illumination. The second reason is the recent developments in 3D acquisition techniques, such as 3D scanners, use of infrared and other technologies which have made obtaining 3D data much easier than it was before. For 3D face recognition applications, two main requirements have to be met. The first requirement is to provide a powerful representation modelling technique for 3D facial image. The second is to provide a matching algorithm or criterion to recognize and distinguish between these models. While the second requirement has been subject to extensive investigation and research, the first requirement is still considered an open research area. A recent paper by Bowyer et al. [67]covered this topic in detail by presenting a comparative survey of 3D face recognition algorithms. They concluded that 3D face recognition has the potential to overcome limitations of its 2D counterpart. In particular, 3D shape data of a face could be used to correct the corresponding 2D facial image, taken with a non-standard pose, to a standard pose.

Compared with 2D face recognition, face recognition based on 3D information is relatively new in terms of literature, algorithms, commercial applications and datasets used for experimentation[68]. The number of people represented in datasets for 3D face recognition experiments didn't reach 100 until 2003[67], with little experimentation explicitly incorporating pose and expression variations. For a small database consists 18 different images of 5 people, Cartoux et al. [69] reported 100% recognition rate when segmenting a range image based on principle curvatures, and allocating the symmetry profile of the face, which was used for normalizing and matching faces against each other.

Gordon [70] takes a feature based approach, based on both depth and curvature information and states that curvature data is easily derived from the range data, and has substantial advantages. It has the potential for high accuracy in describing surface based features and it is well suited to describing properties around facial regions such as the cheeks, forehead and chin. It is also viewpoint invariant. Gordon uses facial feature localisation and absolute measurements (millimetres rather than pixels) to calculate a set of feature descriptors from the 3D data. The usefulness of a feature for the purpose of classification is dependent on its ability to discriminate between people. To evaluate the set of features, a value is calculated for Fisher's linear discriminant (FLD) indicating the level of discrimination for each feature. Gordon proposed an approach with 12 features but the best score was achieved using the following six features: head width, nose height, nose depth, nose width, distance between the eyes and the maximum curvature of the nose ridge. Gordon created a face recognition system using a simple Euclidean distance measure in feature space. Several combinations of features were tested using a database of 24 facial surfaces (8 different people), defining recognition to be correct when the subject was selected as the top match from the database (lowest distance measure). Results ranged from 70.8% to 100% correct recognition (accuracy).

Zhang et al. [71] reported a profile based matching system. Their approach starts by first identifying the symmetry plane (assuming that the facial data is symmetric), then computing the symmetry profile. Based on the mean curvature plot of the facial surface and the symmetry profile, they recovered 3 feature points on the nose area to define what they called the facial intrinsic system (nose tip, nose bridge and point at the lower nose edge)

which were used to standardize the faces for comparison purposes. For individuals with a normal expression, the Equal Error Rate (EER) and the recognition rate reported are 0.8% and 96.9% respectively, but this was only tested on 32 individuals. The equal error rate value indicates that the proportion of false acceptances is equal to the proportion of false rejections. The lower the equal error rate value, the higher the accuracy of the biometric system.

Lee and Milios [72] segmented the range image into convex regions based on the sign of the mean and Gaussian curvature and created an Extended Gaussian Image (EGI). The matching is done by correlating the EGIs of the probe image and images in the gallery. EGI [73] describes the shape of an object by distribution of surface normal over the object structure. Tanaka et al. [74] also presented a correlation based face recognition approach based on the analysis of principal curvatures and their directions. They calculate the maximum and minimum principal curvatures on a face, and extract valley and ridge lines from the curvatures. Then, EGI of ridge and valley lines are constructed by mapping each of principal direction vectors onto two unit spheres for ridge and valley lines. A spherical correlation coefficient [75] was used to estimate similarity between EGI's of two faces. The algorithm was tested with 37 face range images and 100% correct recognition (accuracy) was reported.

Pan et al. [76] uses a similar approach to [77] to detect the symmetry plane of a facial data, where it is assumed that the symmetry plane of an object essentially passes through the centre of mass of the object. Planer curves resulting from the intersection between the

allocated symmetry plane and the facial surface are used for matching between faces. In order to speed up the computation they had to simplify the face images to 2000 vertices.

Pose correction using Romdhani et al. 3D morphable face model technique [78] decreases the error rates when applied to the FERET database [9]. Blanz et al [79], used a 3D morphable face model for identification of 2D face images. First they estimated the lighting direction and face shape. To minimising differences to two dimensional images they iteratively changed shape and texture parameters of the morphable face models. Then these parameters were taken as features for recognition, resulting in an 82.6% correct recognition on a test set of 68 people. The same technique has been utilised in the Huang et al. [80] component based face recognition approach using 3D morphable models from 2D images to get better recognition accuracy when pose and illumination are unknown. The face images are decomposed into smaller components connected by a flexible geometric model, to use them in an SVM approach. Huang et al. deal with the problem of obtaining a large dataset for training, which essential, by synthesising many 2D training images from 3D face models under different pose and lighting conditions. Initially, the 3D morphable model is generated from three 2D images before many face images are synthesised. Then the images are split into components and used to train a second-degree polynomial SVM classifier. A recognition accuracy of 90% was reported.

Zhao and Chellappa [81] used a generic 3D face model which is scaled and aligned to match a 2D target image. If the head is not orientated appropriately (frontal-parallel view) they used the generic 3D face model with light source direction and pose estimations, to

compensate by producing a prototype image of the corresponding frontal pose. Face recognition is then performed on this prototype image. Zhao and Chellappa performed their experiments on the Weizmann database [82] and increased the accuracy from approximately 81% (correct match within rank of 25) to 100%.

Beumier and Acheroy [83] make use of 3D surface information, by performing face recognition using a surface matching approach on the 3D facial structure. These 3D facial surfaces are obtained from a single image taken of a person's face illuminated by structured light (projected light stripes). The 3D structure of the face is computed by measuring the deformation of the stripes across the face. Some orientation normalisation is then required to ensure a consistent front-parallel view. This normalisation is performed by generating some initial parameters for the three angles of rotation and three directions of translation (based on nose detection, forehead and cheek angle), before refining the search until the minimum distance is found between the two facial surfaces being compared. They used various methods of matching the 3D facial surfaces including, extracting 15 profiles, by taking the intersection of the facial surface with evenly spaced vertical planes. These profile curves are compared by dividing the area between the curves by the arc length, giving a distance measure. Beumier and Acheroy carried out verification tests on a database of 120 3D images for 30 people, giving an EER of between 9% and 13% when using automatic alignment, but dropping to between 3.25% and 6% when manual alignment is used. Another method they used was to take one central profile and two lateral profiles. These were converted into one-dimensional vectors of local curvature values. The left and right lateral profiles were averaged to give a mean lateral profile. Facial surfaces were then

compared by calculating the differences between the curvature values of the central and mean lateral profiles. This method gave improved EERs of between 7.25% and 9% on the automatically aligned surfaces and between 6.25% and 9.5% on the manually aligned surfaces.

Similarly, Nagamine et al. [30] tackled face recognition by exploring facial profiles. They used horizontal section (extracted as an intersection of a face surface with a plane parallel to X-Z plane), vertical section (extracted as an intersection of a face surface with a plane parallel to Y-Z plane) and circular cross section (extracted as an intersection of a face surface with a cylinder (axis on Y-Z plane and parallel to Z-axis)). They extracted five feature points, and used them to standardize the face pose. For comparison between faces, Euclidean distance matching between feature vectors of different faces was used. It was concluded that vertical profiles that pass through the central region of the face give better recognition rates, circular sections which cross near the eyes and part of the nose also show some distinctiveness, while the distinctiveness of the horizontal profiles are not remarkable in themselves.

Pan and Wa [84] also explored facial profiles and surfaces from range images, and used them for recognizing faces. First, they extracted the symmetry plane of the face based on reflective symmetry hypothesis. Similar to [71], they then used two horizontal profiles that pass through the nose region, and forehead since they are sensitive to variations of facial expression. For pose standardization and alignment they used the ICP algorithms[85]. Then a statistical model was built to represent different regions on the facial surface and incorporated into a weighted distance function to measure the similarity of surfaces.

Wang et al. [84] tackled face recognition by utilizing both 3D images and 2D images. Using both types of images they extracted four 3D feature points and ten 2D feature points by means of point signatures[86], and stacked Gabor filter responses [5] respectively. Each feature point extracted was associated with a feature vector containing values of 3D and 2D features. In order to identify feature points in test images, PCA was applied using a training set with feature points manually labelled to construct the feature space. For the purpose of matching between faces, they applied two classifiers, one based on a similarity function and the other based on a support vector machine[87].

Chang et al. [88] examined the benefits of using 3D and 2D+3D approaches over 2D using PCA on both 2D intensity images and 3D depth images, and fused 2D and 3D results to obtain the final performance. They had previously made use of this multi-modal data showing how both 2D and 3D data can be utilised to greater effect than using either set of data separately. Results were presented using a database of 275 subjects in an approximately frontal pose for a single probe database; they achieved recognition rates of 89%, 94.5%, and 98.5% for 2D, 3D, and 2D+3D respectively. This clearly indicates an increase in performance can be achieved by using 3D data. Their results show that appearance based methods such as PCA can give good performance for 3D face recognition. However, they selected landmark points manually (the eye centres in 2D, and two eye tips and centre of lower chin in 3D) for facial pose normalization. In addition, their approach requires good facial alignment before matching and requires interpolation or

filtering prior to PCA encoding to handle missing data due to self-occlusion or sensor drop-outs.

Hasher et al.[89, 90] used PCA and Independent Component Analysis (ICA) to analyze range images in a similar way to 2D intensity images and estimated probability models for the coefficients. For registration, and pose standardization they used the nose tip and the nose bridge. They used a database of 37 individuals with images of 6 different facial expressions for each [89].

Bronstein et al. [91] used a photometric stereo technique to estimate facial surface shape from several 2D images, acquired under varying lighting conditions. The surface gradient is used by a Fast Matching on Triangulated Domains (FMTD) algorithm to produce a map of geodesic distances across the facial surface without the need to reconstruct an actual 3D face model. Moments of a canonical form of this geodesic map are extracted as a face signature and compared using the Euclidean distance metric. Taking images of seven subjects from the Yale face database [92], Bronstein et al. were able to show that within-class and between-class standard deviations were improved by an order of magnitude, compared with the direct comparison of facial surface gradients.

Another approach was taken by Chua et al.[93], who treated the face recognition problem as a 3D recognition problem for non-rigid surfaces. The characteristic they use to identify the rigid areas of faces and ultimately to distinguish between faces is the point signature, which describes the depth values surrounding the local region of a specific point

on the facial surface. By observing range data for various facial expressions, they noted that certain areas of the face remain rigid while other areas can deform significantly. An attempt was made to extract these rigid areas of the face for recognition, thus creating a system that is invariant to facial expression. The first stage of point signature based recognition is the registration of 3D faces, performed by finding 3 points on each of two facial surfaces for which signatures of corresponding points match. The distances between the 3 points on one face and the 3 points on the other face must be within an error tolerance. Once the three corresponding points on each face have been found the two facial surfaces can be aligned. Distances between the two facial surfaces are then measured according to a Gaussian distribution model and areas with a low distance measure are extracted as rigid areas. Two facial surfaces are compared by extracting the rigid areas of the face then generating point signatures for all points on the remaining facial surface. This set is reduced to a set of unique point signatures, which are then compared to the corresponding point signatures on the other facial surface. Chua et al. gathered results from 30 range images taken from 6 different people (four different expressions for each person, plus one used as a query). The correct scores ranged from 78.85% to 93.75%, the highest incorrect score being 76.92%.

Coombes et al. [94] presented a method of mathematically describing a facial surface shape, based on differential geometry. Curvature analysis is performed on a depth map of the facial surface, to produce segmentation into one of eight fundamental surface types. In their work the depth map of a facial surface is created from a specific viewpoint, therefore a prerequisite is some 3D facial alignment. They also describe several methods of computing the curvature of the facial surface and indicate the advantages of their proposed method,

which takes into account the possible error due to lack of detail when surfaces make a large angle to the viewpoint. Coombes et al. suggest that two faces may be distinguished by comparing which regions of the two faces are classified as the various surface types. Prior to two facial surfaces being compared, they must be registered. This is done by manually selecting several corresponding feature locations on each face, which are then used to scale, translate and rotate, so that the facial surfaces are aligned. Using this method, 3D facial surfaces have been produced for average female and average male faces, showing distinct differences in chin structure, nose shape, forehead shape and cheek bone position. A quantitative analysis of the average male and average female facial surfaces shows that the main differences are of significance in specific areas of the face.

Bronstein et al. [95] have investigated the use of texture mapped 3D face models focusing on expression invariant face recognition using geometric invariants of the face extracted from 3D face data for non-rigid surface comparisons. By using isometric surface signatures known as bending-invariant canonical forms, generated by dimensionality reduction through multidimensional scaling, they compare signatures by a weighted Euclidean distance metric. Testing the method on a database of 147 subjects, the method was shown to outperform the standard eigen-decomposition method for 10 identification operations.

The generalized Hausdorff distance, defined in Huttenlocher et al. [96] for partial matching, was employed by Achermann and Bunke [97-99] to investigate face recognition performances using both 3D point and voxel array representations. This enabled the examination of trade-offs between performance and computation time. Their best

probability of recognition on a database of 24 subjects with 10 different facial angles was obtained using a Hausdorff fraction of 0.9. This resulted in a recognition rate of 99.2% for the point representation and 100% with a 2.5% reject rate for the voxel representation. The reject rate represents cases in which ties occurred in the experiment due to data quantization caused by using the voxel array.

Moreno et al. [100] extract 86 feature vectors from segmentation of the face using Gaussian and mean curvatures. Each feature is given with a weight determined using Fisher coefficients [101]. 420 3D images of 60 individuals were tested, and the first 35 ranked features were selected to represent faces. They reported a 78% correct recognition rate when the best match was selected and a 92% recognition success rate when the five best matches were selected.

The Razdan et al. [102, 103] approach is a combination of feature points, profile curve, and partial face surface matching. Their proposed technique divided into three parts: data acquisition, feature extraction, and face recognition. The 3D dataset for this project was acquired using a combination of commercial scanning technologies and research software applications developed at Partnership for Research in Spatial Modelling (PRISM) laboratory. They represent 3D data by a triangle mesh. The feature extraction phase of face classification uses curvatures, registration of faces, finds symmetry planes, critical points, and profile curves, nose and biometrically relevant sub-face areas. For the recognition phase, they make comparison between two faces using all the facial features indicated earlier. The test set for authentication included 117 different people with 421 scans

including different facial expressions. They achieved EERs of 0.065% for normal faces and 1.13% for faces with expressions. Verification rates of 100% in normal faces and 93.12% in faces with expressions at 0.1% FAR were reported. For identification, the experiments achieved a 100% rate in normal faces and 95.6% in faces with expressions [103].

Elyan and Ugail [104] presented a method to determine the symmetry profile of the face. They computed the intersection between the symmetry plane and the facial mesh and then computed a few feature points along the symmetry profile in order to allocate the central region of the face and extract a set of profiles from that region. In this approach, they assume that the symmetry profile passes through the tip of the nose. To locate the tip of the nose they fit a bilinear blended Coon's surface patch. Coon's patch is simply a parametric surface defined by four given boundary curves[104]. These four boundaries of the Coon's patch are determined based on a boundary curve that encloses an approximated central region of interest, which is simply the region of the face that contains or is likely to contain the nose area. This region is approximated based on the centre of the mass that represents the 3D facial image. In total 365 images were used for testing and were correctly identified which corresponds to an accuracy recognition rate equal to 86.90%.

Despite all this prior work, there are still a number of areas that 3D face recognition research needs to address. For registration, automatic landmark localization, artefact removal, scaling, and elimination of errors due to occlusions, glasses, beard, etc. need to be worked out. Ways of deforming the face without losing discriminative information might also be beneficial. It is likely that information fusion is the future of 3D face recognition.

There are many ways of representing and combining texture and shape information. Publicly available 3D datasets are necessary to encourage further research on these topics.

Table 2-1 gives a comparison of selected elements of algorithms reported in this literature that use 3D facial data to recognize faces. Most papers report performance as the recognition rate, although some report the equal error rate or verification rate at a specified false acceptance rate. Previously the experimental component of work in this area was rather modest and the number of persons represented in experimental data sets did not reach 100 until 2003 [67]. Also only a few works have dealt with data sets that explicitly incorporate pose and/or expression variation [100, 105-107]. However, just comparing different 3D face recognition techniques is very challenging for a number of reasons. Firstly, there are very few standardized 3D face databases which are used for benchmarking purposes. Thus, the size and type of 3D face datasets vary across different publications. Secondly, there are differences in the experimental setups and in the metrics which are used to evaluate the performances of face recognition techniques (see Table 2-1 for details).

Table 2-1 Summary of recognition algorithms using 3D facial data

Reference	Database Size	Image Size	Data Representation	Algorithm	Reported Performance	Handle size Variation	Handle Expression	Model Type
Cartoux et al. [108], 1989	18	N/A	Profiles, surface	Curvature based Nearest Neighbor	100%	Yes	No	3D
Lee et al. [72], 1990	6	256 × 150	EGI	Correlation	N/A	No	Some	3D
Gordon, [70],1992	26	N/A	curvatures	Euclidean nearest neighbor	100%	Yes	No	3D
Nagamine et al. [30],1992	160	256× 240	Multiple profiles	Euclidean nearest neighbor	100%	Yes	No	3D
Tanaka et al.[74],1998	37	256 × 256	curvatures based EGI	Fisher's spherical Correlation	100%	No	No	3D
Zhao and Chellappa [11],2000	N/A	N/A	3D model + Texture	Produce a prototype images	81%	N/A	N/A	3D
Achermann et. al.[97], 2000	240	75 × 150	Point set	Hausdorff distance	100%	Yes	No	3D
Beumier et al.[109], 2001	120	N/A	2D and 3D vertical profiles	Minimum distance, fusion	1.4% EER	Yes	No	3D +2D
Wang et al. [84], 2002	300	128 ×512	Feature vector point signature Gabor features	PCA+SVM	>90%	No	Yes	3D +2D
Bronstein et al. [110], 2003	147	2250 points	Texture+ Range images	PCA , nearest neighbor	Not reported	Yes	Yes	3D +2D
Chang et al. [111],2003	278	480×640	Texture+ Range image	PCA	99% 3D + 2D, 93% 3D only	Yes	No	3D +2D
Hesher et al. [89],2003	222	242 × 347	Range image	ICA or PCA, nearest neighbor	97%	Yes	No	3D
Moreno et al. [100], 2003	420	2.2K points	Curvature, line, region features	Euclidean nearest neighbor	78%	Yes	Some	3D
Pan et al [99], 2003	360	3K points	Point set, range image	Hausdorff and PCA	3–5% EER	N/A	N/A	3D
Chang et al. [111], 2003	278	480×640	Texture+ Range image	PCA	99% 3D + 2D, 93% 3D only	Yes	No	3D +2D
Tsalakanidou, [112] 2003	80	100×80	Range image	PCA	99%	N/A	N/A	3D +2D
Xu et al. [113], 2004	120 and 30	N/A	Point set + feature vector	Minimum distance	96% on 30, 72% on 120	N/A	N/A	3D
Lee et al.[114], 2004	84	240 × 320	Range, curvature	Weighted Hausdorff	98%	Yes	No	3D
Lu et al. [115],2005	196	240 × 320	Surface mesh	ICP, TPS	89%	N/A	N/A	3D
Lee et al.[116], 2005	70	320×320	depth map	Feature extraction, nearest neighbor	94%	Yes	No	3D
Bronstein et al.[106], 2005	220	N/A	Point set	Canonical forms	100%	N/A	N/A	3D
Lee et al. [116], 2005	200	Various	Feature vector	SVM	96%	Yes	No	3D
Pan et al. [117],2005	720	480 × 640	Range image	PCA	95%	N/A	N/A	3D
Zhang et al. [71],2006	32	Not available	Range images	mean curvature	96.9%	No	No	3D
Razdan et al.[103], 2007	421	N/A	triangle mesh	Spin Image	>93.12%	N/A	Yes	3D
Qatawnah et al.[118],2008	56	480 × 640	Range images	Feature extraction, CCNN, SVM and KNN	100%	Yes	Yes	3D
Elyan and Ugail [104],2009	144	N/A	triangle mesh	Coon's surface patch	86.9%	No	Yes	3D

In this literature, we view our contribution as proposing a brief assessment for existing 3D facial recognition techniques. The recognition rates reported by the various works listed in Table 2-1 showed that a number of factors are combined to make direct comparisons problematic in most cases. Among these factors are different sizes of data set, different inherent levels of difficulty of the dataset, and different methods of experimental design. The results reported by Xu et al.[113] give an example of how dramatically the size of a dataset can affect reported performance. They found 96% recognition rate using a 30 person dataset, but this fell to 72% when using a 120 person dataset. Moreover, the reported performance is also greatly dependent on the inherent difficulty of the data. The presences of expression variation is one element of increased difficulty, but pose variation, time lapse between gallery and probe, presence of eyeglasses, and other factors are also important. The design of the experiment also influences the reported performance.

2.2.1 3D Face Recognition Techniques

Since the end of the last decade interest in 3D face recognition revived increased rapidly. In the following sections we have divided 3D face recognition techniques broadly into three categories: surface based, statistical based, and model based approaches.

2.2.1.1 Surface based approaches

Surface-based approaches use directly the surface geometry that describes the face. These approaches can be classified into those that extract either local or global features of the

surface (e.g. curvature), those which are based on profile lines and those which use distance based metrics between surfaces for 3D face recognition. In the following, some of local method approaches are given in Section 2.2.1.1.1 while global methods are described in Section 2.2.1.1.2.

2.2.1.1.1 Local methods

One approach for 3D face recognition uses a description of local facial characteristics based on Extended Gaussian Images (EGI) [119]. Alternatively the surface curvature can be used to segment the facial surfaces into features that can be used for matching [70]. Another approach is based on 3D descriptors of the facial surface in terms of their mean and Gaussian curvatures [100] or in terms of distances and the ratios between feature points and the angles between feature points [116]. Another locally-oriented technique is based on using point signatures, an attempt to describe complex free-form surfaces, such as faces [120]. The idea is to form a representation of the neighbourhood of a surface point. These point signatures can be used for surface comparisons by matching the signatures of data points of a "sensed" surface with the signatures of data points representing the model's surface [93]. To improve robustness towards varying facial expressions, those parts of the face that deform most non rigidly (mouth and chin) are discarded so only more rigid regions (e.g. forehead, eyes, nose) are used for face recognition.

2.2.1.1.2 Global methods

Global surface-based methods are methods that use the whole face as the input to a recognition system. One of the earliest systems is based on first locating the face's plane of

bilateral symmetry and to use this for aligning faces [69]. The facial profiles in this plane are then extracted and compared. Faces can also be represented based on the analysis of maximum and minimum principal curvatures and their directions[74]. In these approaches the entire face is represented as an EGI. Another approach uses EGIs to summarize the surface normal orientation statistics across the facial surface [121]. A different type of approach is based on distance-based techniques for face matching. For example, the Hausdorff distance has been used extensively for measuring the similarity between 3D faces [98, 99]. Other approaches propose preliminary face alignment using rigid registration algorithms such as ICP [85]. After registration the residual distances between faces are measured and used to define a similarity metric [122]. In addition, surface geometry and texture can be used jointly for registration and similarity measurement in the registration process, and measures not only distances between surfaces but also between texture [123]. Another common approach is based on the registration and analysis of 3D profiles and contours extracted from the face [83, 124, 125]. This technique can also be used in combination with texture information [109].

Finally, hybrid techniques that use both local and global geometric surface information can be employed. In one such approach, local shape information in the form of Gauss-Hermite moments is used to describe an individual face along with a 3D mesh representing the whole facial surface. Both global and local shape information are encoded as a combined vector in a low-dimensional PCA space, and matching is based on minimum distance in that space [113].

2.2.1.2 Statistical approaches

Statistical techniques such as PCA are widely used for 2D facial images. More recently, PCA-based techniques have also been applied to 3D face data [89, 126-128]. This idea can be extended to include multiple features into PCA such as colour, depth or a combination of colour and depth [112]. These PCA-based techniques can also be used in conjunction with other classification techniques such as embedded hidden Markov models (EHMM) [129]. Other approaches are based on the use of Linear Discriminant Analysis (LDA) [130] or Independent Component Analysis (ICA) [131] for analyses of 3D face data.

All of the statistical approaches discussed so far do not deal with the effects of variations in facial expressions. In order to minimize these effects, several face representations have been developed which are invariant to isometric deformations [95] such as those resulting from different expressions and postures of face. The obtained geometric invariants allow mapping 2D facial texture images into special images that incorporate the 3D geometry of the face. These signature images then decomposed into their principle components [95]. One such approach is based on flattening the face onto a plane to form a canonical image which can be used for face recognition [95, 106]. These techniques rely on multi-dimensional scaling (MDS) to flatten complex surfaces onto a plane [132]. Such an approach can be combined with techniques such as PCA for face recognition [117].

2.2.1.3 Model based approaches

The key idea of model based techniques for 3D face recognition is the so-called 3D morphable model. In these approaches, the appearance of the model is controlled by the model coefficients. These coefficients describe the 3D shape and surface colours (texture), based on the statistics observed in a training dataset. Since 3D shape and surface texture are independent of the viewing angle, the representation depends little on the specific imaging conditions [79]. Such a model can then be fitted to 2D images and the model coefficients can be used to determine the identity of the person [12]. While this approach is fairly insensitive to the viewpoint, it relies on the correct matching of the 3D morphable model to a 2D image that is computationally expensive and sensitive to initialization. To tackle these difficulties, component-based morphable models have been proposed [80, 133]. Instead of using statistical 3D face models it is also possible to use generic 3D face models. These generic 3D face models can then be made subject-specific by deforming the generic face model using feature points extracted from frontal or profile face images [134, 135]. The resulting subject-specific 3D face model is then used for comparison with other 3D face models. A related approach is based on the use of an Annotated Face Model (AFM) [107]. This model is based on an average 3D face mesh that is annotated using anatomical landmarks. This model is deformed none rigidly to a new face, and the required deformation parameters are used as features for face recognition. A similar model has been used in combination with other physiological measurements such as visible spectrum and thermal infrared sensors [136].

A common problem of 3D face models is caused by the fact that 3D capture systems can only capture parts of the facial surface. This can be addressed by integrating multiple 3D surfaces or depth maps from different viewpoints into a more complete 3D face model which is less sensitive to changes in the viewpoint [115]. Instead of using 3D capture systems for the acquisition of 3D face data, it is also possible to construct 3D models from multiple frontal and profile views [137].

In summary, just comparing different 3D face recognition techniques is very challenging for a number of reasons. Firstly, there are very few standardized 3D face databases which are used for benchmarking purposes. Thus, the size and type of 3D face datasets varies across different publications. Secondly, there are differences in the experimental setups and in the metrics which are used to evaluate the performances of face recognition techniques.

2.2.2 Comparison between 2D and 3D face recognition systems

As previously discussed, face recognition using 2D images is sensitive to illumination changes. The light collected from a face is a function of the geometry of the face, the albedo of the face, the properties of the light source and the properties of the camera. Given this complexity, it is difficult to develop models that take all these factors into account. Training of recognition systems using different illumination scenarios as well as illumination normalization of 2D images has been used, but with limited success. In 3D

images, variations in illumination only affect the texture of the face, but the captured facial shape remains intact [1].

Another differentiating factor between 2D and 3D face recognition is the effect of pose variations. In working with 2D images, effort has been put into transforming an image into a canonical position [138]. However, this relies on accurate landmark placement and does not tackle the issue of occlusion. Moreover, in 2D this task is nearly impossible due to the projective nature of 2D images. To circumvent this problem it is possible to store different views of the face [139]. However, this requires a large number of 2D images from many different views to be collected for each face. Alternative approaches to address the pose variation problem in 2D images are either based on statistical models for view interpolation or on the use of generative models [140]. Other strategies include sampling the plenoptic function of a face using light field techniques[141]. The plenoptic function is the 5-dimensional function representing the intensity or chromacity of the light observed from every position and direction in 3-dimensional space[142]. Using 3D images, this view interpolation can be solved by re-rendering the 3D face data with a new pose. This allows a 3D morphable model to be used to estimate the 3D shape of unseen faces from non-frontal 2D input images and to generate 2D frontal views of the reconstructed faces by re-rendering [12].

Another pose-related problem is that the physical dimensions of the face recorded in 2D images are often unknown. The size of a face in a 2D image is essentially a function of the distance of the subject from the sensor. However, in 3D images the physical dimensions of the face are inherently encoded in the data. In contrast to 2D images, 3D images are better at capturing the surface geometry of the face. Traditional 2D image-based face recognition

focus on areas of the face such as eyes, mouth, nose and face boundary which have high-contrast while areas such as the jaw boundary and cheeks are difficult to describe from intensity images because of low contrast [70]. 3D images, on the other hand, make no distinction between high- and low-contrast areas.

3D face recognition, however, is not without its problems. Illumination, for example, may not be an issue during the processing of 3D data, but it is still a problem during capture. Depending on the sensor technology used, oily parts of the face with high reflectance may introduce artefacts under certain lighting of the surface. The overall quality of 3D image data collected using a range camera is perhaps not as reliable as 2D image data, because 3D sensor technology is currently not as mature as 2D sensors. Another disadvantage of 3D face recognition techniques is the cost of the hardware. 3D capturing equipment is getting cheaper and more widely available but its price is still significantly higher compared to a high resolution digital camera. Moreover, the size of 3D face image is much larger than the corresponding 2D image and the current computational cost of processing 3D data is higher than for 2D data.

Finally, one of the most important disadvantages of 3D face recognition is the fact that 3D capturing technology requires the co-operation of the subject. As mentioned above, lens or laser based scanners require the subject to be at a certain distance from the sensor. Furthermore, a laser scanner requires a few seconds of complete immobility, whereas a traditional camera can capture images from far away with no cooperation from the subjects. In addition, there are currently very few high-quality 3D face databases available for testing and evaluation purposes. Those databases that are available are generally very small in size compared to the 2D face databases used for benchmarking.

2.3 Face Recognition Grand Challenge Data Base

The Face Recognition Grand Challenge (FRGC) was designed to promote the development of 2D and/or 3D algorithms and to improve error performance over FRVT 2002[64]. Determining the performances of these techniques raises three issues: (i) having sufficient data for statistical significance; (ii) challenge problems require an order of magnitude improvement; (iii) infrastructure allowing objective comparison of the contending methods. The FRGC addresses all three issues[143]. The FRGC data set consists of 50,000 images divided into training and validation partitions. The data set contains high resolution still images taken with a 4 mega pixel Canon PowerShot G2 camera under controlled lighting conditions producing unstructured illumination and with, on average, between 260 and 144 pixels respectively between the eye centres[144]. 3D images are acquired using a Minolta Vivid 900/910 series structured lighting sensor which outputs 640 by 480 range samples and a registered colour image with about 160 pixels between the eyes. Although recognizing faces from the 3D shape of a person's face has the potential to improve performance because the shape of faces is not affected by changes in lighting or pose [64], it has not yet been demonstrated that this potential can be achieved in practice.

There are three aspects of the FRGC that are new to the face recognition community. The first aspect is the size of the FRGC data which contains 50,000 recordings. The second aspect is the complexity of the FRGC. Previous face recognition data sets have been restricted to still images. The FRGC consists of three modes: high resolution still images, 3D images, and multiple-images of a person. The third new aspect is the infrastructure. The infrastructure for FRGC is provided by the Biometric Experimentation Environment (BEE),

an XML based framework for describing and documenting computational experiments [28].

The FRGC challenge problem consists of six experiments designed to promote the advancement of face recognition in general, with the emphasis on 3D and high-resolution still imagery. The experiments measure recognition performance on: 1) single front-view still images taken with controlled lighting and background; 2) four views of still images under controlled conditions like 1; 3) 3D query image with 3-D target image; 4) uncontrolled single still images; 5) single controlled 2D query image with 3D target image; 6) single uncontrolled 2D query image with 3D target images. The infrastructure ensures that results from different algorithms are computed on the same data sets and that performance scores are generated by the same procedure [28, 145].

2.3.1 Design of data set and challenge problem

The creation of the FRGC followed FRVT 2002[144], by establishing a performance goal that was an order of magnitude greater than before, and then designed a data set and a number of challenge problems that contribute to meeting the FRGC goals. The starting point for measuring the increase in performance is the High Computational Intensity test (HCInt) of the FRVT 2002 [101]. The images in the HCInt set were taken indoors under controlled lighting. The performance point selected as the reference is a verification rate of 80% (error rate of 20%) at a false acceptance rate (FAR) of 0.1%. This was the performance level of the top three FRVT 2002 participants[144]. An order of magnitude

increase in performance corresponds to a verification rate of 98% (2% error rate) at the same fixed FAR of 0.1% [143].

A challenge in creating the FRGC was collecting sufficient data to allow an error rate of 2% to be measured [146]. Verification performance is characterized by two statistics: verification and false acceptance rates[146]. The false acceptance rate is computed from comparisons between faces of different people. These comparisons are called non-matches. In most experiments, there are sufficient non-match scores because the number of non-match scores is usually quadratic in the size of the data set. The verification rate is computed from comparisons between two facial images of the same person. These comparisons are called match scores. Because the number of match scores is dependent in the data set size, generating a sufficient number of matches can be difficult[147]. The challenge is to design a data collection system that yields 50,000 match scores. This was accomplished by collecting images for a medium number of people with a medium number of replicates at a rate of about 200 images per week for a year.

The design, development, tuning and evaluation of face recognition algorithms require three data partitions: training, validation, and testing [28, 143, 146, 147]. The FRGC challenge problem provides training and validation partitions. A separate testing partition is collected and sequestered for an independent evaluation. The representation, feature selection, and classifier training is conducted on the training partition. For example, in PCA-based and LDA-based face recognition, the subspace representation is learned from the training set [147]. Challenge problem experiments are constructed from data in the validation partition. During algorithm development, repeated runs are made on the

challenge problems. This allows researchers to assess the best approaches and tune their algorithms [146, 147]. Repeated runs produce algorithms that are tuned to the validation partition. An algorithm that is not designed properly will not generalize to another data set [144].

The FRGC experimental system is based on the FERET and FRVT 2002 testing systems [144, 146, 147]. For an experiment, the input to an algorithm is two sets of images: target and query sets. Images in the target set represent facial images known to the system. Images in the query set represent unknown images presented to the system for recognition. The output from an algorithm is a similarity matrix, in which each element is a similarity score that measures the degree of similarity between two facial images.

2.4 Machine Learning Techniques

Automatic Face Recognition can be seen as a pattern recognition problem, which is very hard to solve due to its non-linearity. Particularly, it is cast as a template matching problem, where recognition has to be performed in a high-dimensional space. Since the higher the dimensionality of the space, the more the computation is needed to find a match, a dimensional reduction technique is used to project the problem into a lower dimensionality space. Indeed, the neural network can be considered as a good solution to the face recognition problem, commonly used in many other pattern recognition problems, and readapted to cope with the people authentication task. One of neural classifiers advantage is that they can reduce misclassifications among the neighbourhood classes.

2.4.1 Cascade Correlation Neural Network

The training of back-propagation neural networks is considered to be a slow process because of the step-size and moving target problems [148]. To overcome these problems cascade neural networks were developed. These are "self organizing" networks [148] with topologies which are not fixed. The supervised training begins with a minimal network topology and new hidden nodes are incrementally added to create a multi-layer construction. The new hidden nodes are added to make the most of the correlation between the new node's output and the remaining error signal that the system is being adjusted to eliminate. The weights of a new hidden node is fixed and not changed later, hence making it a permanent feature detector in the network. This feature detector can then be used to generate outputs or to create other more complex feature detectors [148].

In a CCNN, the number of input nodes is determined by the input features, while the number of output nodes is determined by the number of different output classes. The training of a CCNN starts with no hidden nodes. The direct input-output connections are trained using the entire training set with the aid of the back propagation learning algorithm. Hidden nodes are then added gradually and every new node is connected to every input node and to every pre-existing hidden node. The goal of this adjustment is to maximize *S*, the sum overall output units *o* of the magnitude of the correlation1 between *V*, the candidate unit's value, and E_o, the residual output error observed at unit *o*. *S* can be defined as:

$$S = \sum_o \left| \sum_p (V_p - \bar{V})(E_{p,o} - \overline{E_o}) \right| \qquad (2.1)$$

where o is the network output at which the error is measured and p is the training pattern. The quantities $\bar{V}$ and $\overline{E_o}$ are the values of V and E_o averaged over all patterns.

Training is carried out using the training vector and the weights of the new hidden nodes are adjusted after each pass [148]. Cascade correlation networks have a number of attractive features including a very fast training time, often a hundred times faster than a perceptron network [148]. This makes cascade correlation networks suitable for use with large training sets.

Depending on the application and number of input nodes, cascade correlation networks are fairly small, often having fewer than a dozen neurons in the hidden layer [149, 150]. This can be contrasted with probabilistic neural networks which require a hidden-layer neuron for each training case. Also, the training of CCNNs is quite robust, and good results can usually be obtained with little or no adjustment of parameters [148].

2.4.2 Support Vector Machine

Support Vector Machines are becoming popular tools for solving a variety of learning and function estimation problems. In contrast to neural networks, SVMs have a firm statistical foundation and are guaranteed to coverage to the global minimum during training [157]. They are also considered to have better generalization capabilities than neural networks and have managed to outperform neural networks in a number of applications[151-156]. SVMs employ a statistical learning theory based algorithm, introduced by V.Vapink [157], which uses the structural risk minimization principle. SVMs use non-linear-transformation kernel

functions to map the input data to a high dimensional feature space, where data can become linearly separable[157].

In the work described here, the LIBSVM tools developed by Chih-Chung Chang and Chih-Jen Lin [156,157] were used to build the classifier. A classification task usually involves training and testing data which consist of some data instances. Each instance in the training set contains one target value (class label) and several attributes (features). The goal of SVM is to produce a model which predicts target values of data instances in the testing set when provided only with the attributes. Given a training set of instance-label pairs $(xi; yi);\ i = 1; \ldots; l$ where $xi \in R^n$ and $y \in \{1, -1\}^l$, the support vector machines (SVM) [158]require the solution of the following optimization problem:

$$\min_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i \quad \text{subject to } y_i (w^T \phi(x_i) + b) \geq 1 - \xi_i, \xi_i \geq 0 \qquad (2.2)$$

Here training vectors x_i are mapped into a higher (maybe infinite) dimensional space by the function ϕ. Then SVM finds a linear separating hyperplane with the maximal margin in this higher dimensional space. $C > 0$ is the penalty parameter of the error term. Furthermore, $K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j)$ is called the kernel function.

In order to use the LIBSVM it was necessary to transform data to the format of the SVM package and then conduct simple scaling on the data afterwards. For this work the Polynomial, Radial Basis Function (RBF) and Sigmoid kernels were used. This procedure is discussed in detail in the Chapter 5.

2.4.3 Nearest Neighbours method

The K-Nearest Neighbour method (KNN) is part of a supervised learning algorithm where the result of a new instance query is classified based on the majority of K-nearest neighbour categories [159]. It has been used in many applications in the fields of data mining, statistical pattern recognition, image processing and many others. The KNN method initially introduced by J. G. Skellam, where the ratio of expected and observed mean value of the nearest neighbour distances, is used to determine if a data set is clustered [160]. Further work was done by P. J. Clark and F. C. Evans to introduce a statistical test of significance for the nearest neighbour statistic in order to quantify the departure of the pattern from random [161]. This test is of great importance because even randomly generated data can be labelled as clustered, but the significance test will indicate if the evidence for this classification is lacking.

The KNN algorithm works based on minimum distance from the query instance to the training samples to determine the K-nearest neighbours. After gathering the K nearest neighbours, a simple majority of these K-nearest neighbours are taking to be the prediction of the query instance. The data input to a KNN algorithm consist of several multivariate attributes X_i that will be used to classify Y. The KNN input data can be any measurement scale from ordinal, nominal, to quantitative. The KNN algorithm has two main advantages. Firstly, it is robust to noisy training data (especially if one uses Inverse Square of weighted distance as the "distance"). Also, it's effective if the training data is large.

In order to apply the KNN algorithm it is necessary to determine the parameter K which represents the number of nearest neighbours and calculate the distance between the query instance and all the training samples. Then the distances are sorted to determine the K^{th} nearest neighbours. The Y categories of the nearest neighbours are gathered in order to use the majority of the categories as the prediction value for the query instance. More details are included in Chapter 5.

2.4.4 Verification and Validation Techniques

The Jack-knife technique [162] is usually implemented to provide a correct statistical evaluation of the performance of a classifier when applied to a limited number of samples divided into two sets: a training set and testing set. This technique was employed to evaluate the performances of the learning system used in this work because it has proved to be more appropriate and efficient when applied to different pattern recognition applications. In addition, the use of jack-knife learning algorithms with 80% of the data for training and 20% for testing lead to a reasonably reliable results when it is applied to evaluate a similar classification problem [163].

In practice, a random number generator is used to select the samples used for training and the samples kept for testing. The classification error varies with the training and testing sample sets and, for a finite number of samples, an error-counting procedure is used to estimate the performance of the classifier [162]. In this work, 80% of the available samples were randomly selected and used for training while the remaining 20% were used for testing. The results were then analyzed to assess the performance.

The values for negative and positive results in an experiment are likely to be the most useful practically. The performance criteria used in this work are accuracy (the fraction of all correct predictions), sensitivity (the fraction of positive cases correctly classified) and specificity (the fraction of negative cases correctly classified) [164]. These performance criteria are the most common criteria to evaluate any face recognition system, and they are defined as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.3}$$

$$Specificity = \frac{TN}{TN + FP} \tag{2.4}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{2.5}$$

The numbers of true positive, true negative, false positive and false negative results are indicated by TP, TN, FP and FN respectively, which are defined as follows:

$$TP = \frac{Correct\ positive\ predictions}{Total\ positives} \tag{2.6}$$

$$FP = \frac{Incorrect\ negative\ predictions}{Total\ negatives} \tag{2.7}$$

$$TN = \frac{Correct\ negative\ predictions}{Total\ negatives} \tag{2.8}$$

$$FN = \frac{Incorrect\ positive\ predictions}{Total\ positives} \tag{2.9}$$

In these equations the numerators and denominator terms are defined as follows: “Correct positive predictions” is the total number of cases for which the system makes correct predictions; “Incorrect positive predictions” is the total number of cases for which the system makes an incorrect predicted; “Correct negative predictions” is the total number of cases for which the system correctly predicts a non match; “Incorrect negative predictions” is the total number of cases for which the system incorrectly predicts a non-match; “Total positives” is the sum of correct matching cases (Number of associated cases used in testing); “Total negatives” is the sum of correct non-matching cases (Number of un-associated cases used in testing) [164, 165].

2.5 Summary

2D face recognition systems are widely used at the commercial and academic level. Low price equipments are needed to set up such a system. In addition, it can achieve acceptable recognition rates when images are taken within controlled environment that is similar to the training set of images and when the facial data transform into feature space, then statistical classification algorithms such as NNs, SVMs and KNNs can be applied. However, these approaches suffered from various limitations, such as pose variation, illumination and facial expression which critically decline its performance.

3D face recognition approaches are relatively new trend that tends to utilize the 3D information of the human face. 3D face recognition has the potential to achieve better accuracy than its 2D counterpart by measuring the geometry of rigid features on the face. This avoids such pitfalls of 2D face recognition algorithms as change in lighting, different facial expressions, make-up and head orientation. Although, the utilization of 3D facial data has been shown to be more robust with respect to handling poses and lighting variations, 3D face recognition is still facing several challenges such as the localisation of certain face facial features points without making a prior assumption about the face pose and orientation.

As explained before, one of the more challenging problems in this work is how to compare between different 3D face recognition techniques. Hence, there are few standardized 3D face databases which are used for benchmarking purposes. Thus, the size and type of 3D face datasets varies significantly across different publications. In addition, there are many differences in the pre-processing methods and the experimental setups as well as in the metrics which are used to evaluate the performances of face recognition techniques.

In this literature, we have presented a general view about Face Recognition Grand Challenge database. FRGC was designed to promote the development of 2D and/or 3D algorithms and to improve error performance over FRVT 2002 database. The FRGC addresses the performances issues i.e. having sufficient data for statistical significance challenge problems require an order of magnitude improvement and infrastructure allowing objective comparison of the contending methods[147]. Furthermore, some of

machine learning techniques such as CCNN, SVM and KNN have provided in order to use them in this work.

In this research, we view our contribution as proposing a fully automatic, accurate and robust recognition method for frontal 3D face data, and verifying the nose region as a more promising candidate for features extraction when the expression is uncontrolled. The recognition rates reported by the various works listed in Table 2-1 showed that a number of factors are combined to make direct comparisons problematic in most cases. Among these factors are different sizes of data set, different inherent levels of difficulty of the dataset, and different methods of experimental design. The results reported by Xu et al.[113] give an example of how dramatically the size of a dataset can affect reported performance. They found 96% rank-one recognition using a 30 person dataset, but this fell to 72% when using a 120-person dataset. Moreover, the reported performance is also greatly dependent on the inherent difficulty of the data. The presences of expression variation is one element of increased difficulty, but pose variation, time lapse between gallery and probe, presence of eyeglasses, and other factors are also important. The design of the experiment also influences the reported performance.

In order to tackle the limitations explained above we have used the FRGC database which is considered as the most challenging data set available supporting research on 3D face recognition in regard to the dataset size, the expression variations, and the presence of extraneous features. Furthermore, a reliable system for automatic processing of 3D facial data has been implemented using a multi stage system taking in account several

representations of 3D data and then a set of effective facial features has been extracted. Following this, machine learning techniques are applied for the classification stage. The system performance is reported in terms of accuracy, sensitivity and specificity.

In the next chapter, the 2.5D face recognition system based on range data will be presented. The interpretation of 2.5D data and the extraction of the facial region will be discussed. Moreover, the pre-processing of the face data and the feature extraction will be explained.

CHAPTER THREE

3. 2.5D RANGE IMAGES FEATURES EXTRACTION AND RECOGNITION

3.1 Introduction

The two main representations that are commonly used to model faces in 3D face recognition applications are 2.5D and 3D representations, as illustrated in Figure 3.1 The term 2.5D image, also known as range image or pseudo-3D image, is an informal term used to describe the representation of a three dimensional environment from 2D retinal projections [166], where each pixel in the XY plane has an added "depth" channel (z coordinate) which may act like a height map [105]. The term is also used to describe 3D scenes built completely or partially from a composite of flat 2D images. In order to build a complete 3D head model, several scans have to be made from different viewpoints. Unlike range images, 3D images represent 3D face images by polygonal meshes [167], which contain a set of 3D coordinates with adjacency information. In this chapter, the focus of the discussion will be about range images. In order to investigate the impact of different registration techniques for correspondence estimation on the quality of the 2.5D model for face recognition, we have constructed a 2.5D statistical face model using 56 datasets (FRGC *Ver.2.0* Database). A typical face image consists of about 20,000 points.

(a) (b)

Figure 3-1 Examples of 3D facial representations (a) 2.5D range or depth image (b) point cloud model of 3D data

This chapter is organized as follows. The interpretation of 2.5D data and the extraction of facial region are described in Section 3.2. Nose identification and prior knowledge from anatomy are presented in Section 3.3. Standardization of the face area, the removal of spikes and the filling of holes in the data are all discussed in Section 3.4. Section 3.5 describes the feature extraction phase. The performance achieved is provided in Section 3.6. Finally, Section 3.7 draws some conclusions from the work presented in this chapter. Figure 3.2 shows a block diagram of the proposed face recognition system.

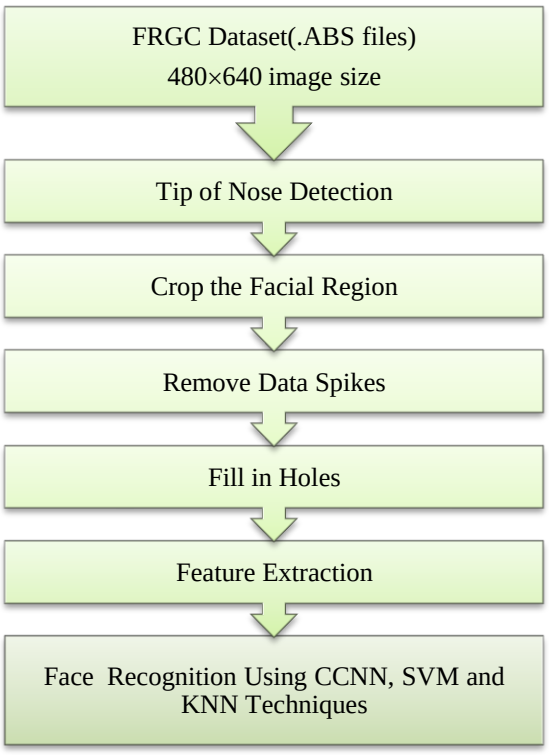


Figure 3-2 Main block diagram of the face recognition system.

3.2 Interpretation of 2.5D Data and Extraction of Facial region

The FRGC database range files have the extension ABS and use the ASCII representation. The ABS file starts with a header that indicates the numbers of rows (480) and columns (640) of the range image in the first two lines. The third line shows the order in which the data is stored in the file (flag, X, Y, Z). The rest of the file is made up of four blocks of space separated data in raster order, each of size rows × columns; the first block is a flag image (face mask), where valid pixels have value 1 and invalid pixels have value 0. The second block includes the X coordinate of each pixel, the third block includes all the Y values and the fourth block includes all the Z values. The flag, X, Y and Z data have been

extracted from the ABS files and displayed in order using the Z data as intensity in order to provide a view of the face-shape. Invalid X, Y, Z values are indicated by the ASCII code -999999.0. It was observed that invalid X or Y values sometimes occur within the face region indicated by the valid flag data and there are also occasional valid data with erroneous values. Figure 3.3 illustrates the content of an ABS file.

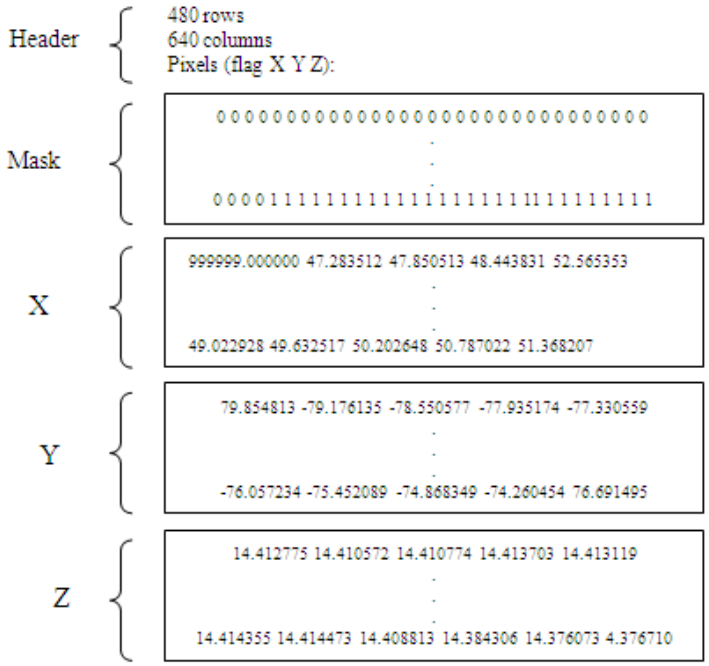


Figure 3-3 The component parts of an ABS file.

The FRGC database also provides texture files accompanying the 3D data. The texture can be mapped from the PPM file, corresponding to an ABS file. The texture can be used by mapping each pixel from the 2D image with each pixel from the range image. For

instance, the pixel (200, 200) from the texture file may be applied to the X, Y and Z data that are stored at position (200,200) in each block of the ABS file. For the most part we can simply discard the (X, Y) data as it falls fairly close to the image plane, which just gives a 2.5D range image of the subject remaining in the Z-data. At this stage of the experiments the ABS images were used without their texture. A sample of 56 3D near frontal images of faces for 7 people with 8 images to each was taken. To extract the facial area from the background given a facial scan, the invalid *Z* points were filtered out using the mask.

3.3 Nose Area Identification

The nose tip is a distinctive point of the human face, especially in the range image. Compared to the whole face region, intuitively the nose region has several advantages. Firstly, human nose, which consists mostly of cartilage, is more rigid than mouth, eyes and cheeks, all of which consist mainly of softer tissues. Secondly, the human nose does not change significantly under most facial expressions. Thirdly, the data in the nose region is usually complete and is seldom occluded by hair or beard [38] .

3.3.1 Evidence from Anatomy

The anatomy of muscles can serve as a direct tool for analyzing facial expressions. Figure 3.4 depicts the muscles of the face. From this figure, it can clearly be seen that, compared to other parts of the human face, the amount of muscle located on the nose region is least. In anatomy, the Facial Action Coding System (FACS)[168] is a *de-facto* standard for 'coding' every conceivable human facial expressions. The basic measurement units defined

in FACS are Action units (AUs). A single Au may consist of more than one muscle. Nearly all anatomically possible facial expressions can be decomposed into specific AUs that produce the expressions [168]. There are a total of 32 Aus defined in FACS. Table 3-1 according to [168] present a summary of the numbers of Aus residing on major parts of the human face. There is only one AU related to the nose region, and the associated movement defined in FACS is called the nose wrinkle [168]. By contrast, the muscles associated with other facial regions all contribute to more than one AU. As such, the nose region can be viewed as a more rigid object than any other parts of the face and is almost invariant under many facial expressions. The good face recognition performance achieved by Chang et al. [169] using multiple nose regions under varying facial expression also supports the assertion that the nose region is relatively more rigid than other parts of the human face.

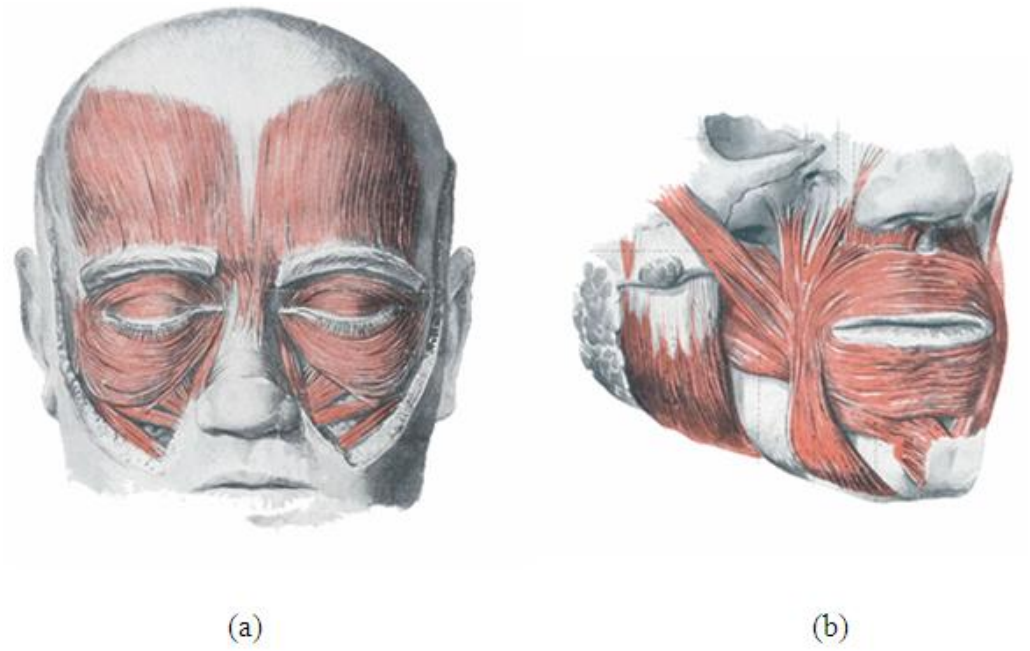


Figure 3-4 Facial muscles: a) muscles of the upper face, b) muscles of the lower face.

Table 3-1 The number of AUs located with major parts of the human face as according to [168]

	Forehead	Eyebrow	Eyes	Nose	Checks	Mouth	Lips	Chin
Aus	**>2**	**>5**	**>5**	**1**	**>5**	**>5**	**>10**	**>5**

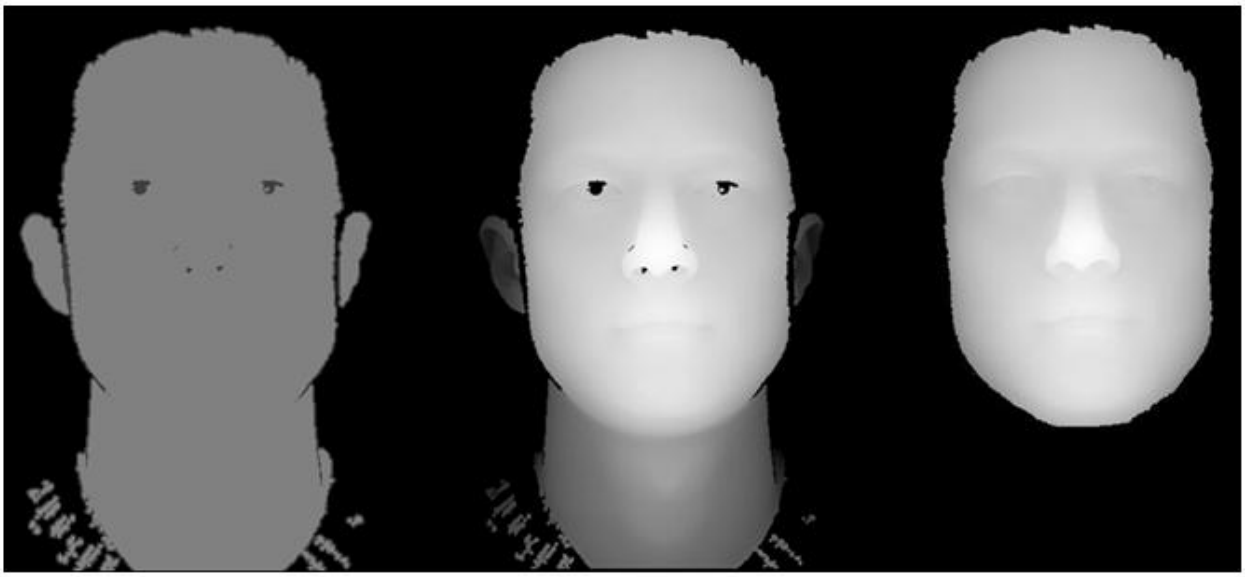
3.4 Standardisation of the Face Area and Holes filling

In the work presented in this chapter, the nose tip was manually detected as the first step in order to crop out the required facial area from the 2.5D face for further processing. This process is needed to ensure we are dealing with the correct region of face. For a frontal facial scan, the nose tip usually has the largest Z value. Then a face region is cropped from the raw 2.5D data to construct a 2.5D face image which is centred on the nose section. The size of each image is 480×640. Once the face has been cropped, the outlier values (Z values) causing spikes in the 2.5D face are removed as follows. The first removing spikes approach involves calculating the distances around each pixel(i,j), the distances are arranged in order of magnitude to determine The variance in the distances around each pixel(i,j) is computed to determine the distances between the points, if the three distances are much greater than the remaining three(by certain value such as 20 pixels) then the current pixels are considered to be part of a spike which is eliminated by identifying and correcting the outlying point. More details are described in Chapter 4 section 4.3.1.

After removing data spikes, holes which may be found in the 2.5D face data are filled using cubic interpolation. It is commonly used in image processing as a useful method that offers true continuity between the pixels [170]. With this method, the value *f(x, y)* of a function *f at a point (x, y)* is computed as a weighted average of the nearest sixteen pixels in a rectangular grid (a 4x4 array). Suppose the function values f and the derivatives f_x, f_y and f_{xy} are known at the four corners (0,0), (1,0), (0,1), and (1,1) of the unit square The equations for the cubic interpolated surface is:

$$p(x,y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} x^i y^j \qquad (3.1)$$

Instead of the two points used in linear interpolation, cubic interpolation uses four points. In general, it is agreed that the cubic interpolation yields superior results to the linear interpolation [170]. Figure 3.5 shows an example of a face which has been subjected to the spike removal and hole filling processes.



(a) (b) (c)

Figure 3-5 (a) The facial area indicated by the flag data (b) The same face with Z value displayed as intensity after spike removal (c) The face after hole filling.

3.5 Feature Extraction

One way of locating corresponding points on different faces is by using landmarks that are manually placed on the 3D features of the face. The landmarks should be placed on anatomically distinct points of the face in order to ensure proper correspondence. However, parts of the face such as the cheeks are difficult to landmark because there are no uniquely distinguishable anatomical points across all faces. It is important to choose landmarks that contain both local feature information (e.g. the size of the mouth and nose) as well as the

overall size of the face (e.g. the location of the eyebrows). Previous work on 3D face modelling for classification has shown that there is not much difference in performance between the use of 11 and 59 landmarks [171]. In the current experiments, 10 landmarks were chosen to capture the shape and size variations of the face appropriately.

Table 3-2 shows the features that are used, and Figure 3.6 shows an example of a face that was manually landmarked [172].

Table 3-2 The 10 manually selected landmarks chosen because of their anatomical distinctiveness.

Anatomical points landmarked	
Points	**Landmark Description**
Eyes	Both the inner and outer corners of the eyelids (4 landmarks).
Nasion	The intersection of the frontal and two nasal bones of the human skull where there is a clearly depressed area directly between the eyes above the bridge of the nose (1 landmark).
Nose tip	The most protruding part of the nose (1 landmark).
Subnasal	The middle point at the base of the nose (1 landmark).
Nose extremes	The outer corners of nose (2 landmarks).
Gnathion	The lowest and most protruding point on the chin (1 landmark).

Two types of features were extracted from faces and compared in order to determine which of the two approaches is more suitable for best recognition; the first uses ***the distances between the chosen landmarks*** and the second uses ***ratios of distances***.

The first approach [118] has started with the four features shown in Figure 3.6. These features are the distance between the outer corners of the eyes (**AB**), the distance between the inner corners of the eyes (**CD**), the distance between nose tip and an align point

between the eyes (**FE**), and the distance between nose extremes (**GH**), respectively. After carrying out extensive experiments, as it will be explained in Section 3.6, it was decided to increase the number of features to five, by adding the distance between the lowest point in the chin and the middle point at the base of the nose (**IJ**).

The second approach [118] used the data extracted from the same dataset but the features chosen are the ratios between the symmetry line of face **(FE)** with the outer corners of eyes **(AB)**, the ratios between symmetry line of face **(FE)** with inner corners of eyes **(CD),** and the ratios between symmetry line of face (**FE)** with the nose extremes (**GH),** respectively. Also, a new feature corresponding to the ratio between symmetry line and the line connects between the lowest point in the chin and the middle point at the base of the nose **(I J**) was added.

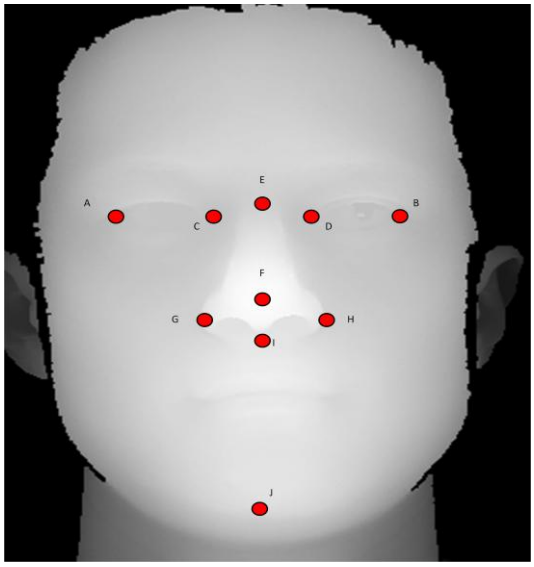


Figure 3-6 The10 manually selected features chosen because of their anatomical distinctiveness

3.6 Experimental Results

In the work presented in this chapter, Cascade Correlation Neural Network (CCNN), Support Vector Machine (SVM), and K-Nearest Neighbour (KNN) were chosen for use in experiments which were carried out to make the final decision for the recognition and classification phases; the classification problem uses a binary output for classifiers in our experiments. The Jack-knife technique was employed to evaluate the performances of the learning system, where 80% of randomly selected samples were used for training and the remaining 20% for testing. The performance criteria used in the work were accuracy, sensitivity, and specificity which are measured using the common biometric measures, namely the true positive ratio (TPR) and the false positive ratio (FPR).

In this work, experiments were conducted on different numbers of people, starting with 56 images representing 7 people, and each individual represented by 10 images that cover a range of poses and expression in which there are variations of unwanted features including hair, neck, shoulders and clothes. Numerical representations of features are used to construct the input variables for the training and testing stages of the machine learning system. The face features were calculated and normalised to be in the range between 0.1 and 0.9 for recognition using CCNN in order to find which of the features are the most significant. The output node has a numerical value of 0.9 if it is the correct class and 0.1 if it is not correct. The data was prepared in another specified format and normalised between -1 and 1 for recognition using SVM. Figure 3.8 shows an example of CCNN input file contains56 images representing 7 individuals and each individual has 8 images

Class		5 Input features					7 Output Classes						
Person 1	Image 1	252	100	42.432	87.001	9.6886	0.9	0.1	0.1	0.1	0.1	0.1	0.1
	Image 2	254	102	42.636	87.001	8.4064	0.9	0.1	0.1	0.1	0.1	0.1	0.1
	Image 3	256	101	42.204	84.001	9.8622	0.9	0.1	0.1	0.1	0.1	0.1	0.1
	Image 4	261	104	43.519	87.001	8.7652	0.9	0.1	0.1	0.1	0.1	0.1	0.1
	Image 5	259	104	44.134	86.001	9.7613	0.9	0.1	0.1	0.1	0.1	0.1	0.1
	Image 6	253	100	43.17	87	10.347	0.9	0.1	0.1	0.1	0.1	0.1	0.1
	Image 7	254	104	43.534	87.001	10.85	0.9	0.1	0.1	0.1	0.1	0.1	0.1
	Image 8	258	103	44.388	85	9.2981	0.9	0.1	0.1	0.1	0.1	0.1	0.1
Person 2	Image 1	127	41	13.463	90	1.5415	0.1	0.9	0.1	0.1	0.1	0.1	0.1
	Image 2	130	41	15.555	88.001	1.614	0.1	0.9	0.1	0.1	0.1	0.1	0.1
	Image 3	126	42	14.528	88	2.6058	0.1	0.9	0.1	0.1	0.1	0.1	0.1
	Image 4	129	41	14.869	88.001	1.8274	0.1	0.9	0.1	0.1	0.1	0.1	0.1
	Image 5	132	41	15.243	90	2.8083	0.1	0.9	0.1	0.1	0.1	0.1	0.1
	Image 6	129	42	13.591	89	2.6058	0.1	0.9	0.1	0.1	0.1	0.1	0.1
	Image 7	128	42	13.339	89	2.9267	0.1	0.9	0.1	0.1	0.1	0.1	0.1
	Image 8	126	42	15.241	89	2.4971	0.1	0.9	0.1	0.1	0.1	0.1	0.1
Person 3	Image 1	195	51	17.142	94	1.3127	0.1	0.1	0.9	0.1	0.1	0.1	0.1
	Image 2	199	53	17.658	93	1.3608	0.1	0.1	0.9	0.1	0.1	0.1	0.1
	Image 3	194	54	18.683	94.006	1.4608	0.1	0.1	0.9	0.1	0.1	0.1	0.1
	Image 4	196	51	16.114	95.627	1.3608	0.1	0.1	0.9	0.1	0.1	0.1	0.1
	Image 5	198	51	18.243	92	1.7376	0.1	0.1	0.9	0.1	0.1	0.1	0.1
	Image 6	199	52	17.221	93.001	1.3333	0.1	0.1	0.9	0.1	0.1	0.1	0.1
	Image 7	195	51	18.928	96	1.8176	0.1	0.1	0.9	0.1	0.1	0.1	0.1
	Image 8	199	52	16.409	93	1.9251	0.1	0.1	0.9	0.1	0.1	0.1	0.1
Person 4	Image 1	173	48	30.535	112	18.853	0.1	0.1	0.1	0.9	0.1	0.1	0.1
	Image 2	176	51	26.33	110	19.561	0.1	0.1	0.1	0.9	0.1	0.1	0.1
	Image 3	174	46	29.223	111	18.49	0.1	0.1	0.1	0.9	0.1	0.1	0.1
	Image 4	177	46	25.122	113	18.811	0.1	0.1	0.1	0.9	0.1	0.1	0.1
	Image 5	173	50	26.167	110	19.636	0.1	0.1	0.1	0.9	0.1	0.1	0.1
	Image 6	179	49	31.511	110	18.097	0.1	0.1	0.1	0.9	0.1	0.1	0.1
	Image 7	170	49	31.608	110	17.612	0.1	0.1	0.1	0.9	0.1	0.1	0.1
	Image 8	176	45	26.122	109	18.097	0.1	0.1	0.1	0.9	0.1	0.1	0.1
Person 5	Image 1	214	78	23.461	133	8.2716	0.1	0.1	0.1	0.1	0.9	0.1	0.1
	Image 2	219	77	19.402	129	7.4773	0.1	0.1	0.1	0.1	0.9	0.1	0.1
	Image 3	216	78	19.543	129	7.1087	0.1	0.1	0.1	0.1	0.9	0.1	0.1
	Image 4	214	79	19.981	131	8.2303	0.1	0.1	0.1	0.1	0.9	0.1	0.1
	Image 5	214	78	23.271	135	7.9692	0.1	0.1	0.1	0.1	0.9	0.1	0.1
	Image 6	220	77	21.586	135	8.6757	0.1	0.1	0.1	0.1	0.9	0.1	0.1
	Image 7	213	75	23.06	130	8.6427	0.1	0.1	0.1	0.1	0.9	0.1	0.1
	Image 8	219	76	19.992	130	9.4908	0.1	0.1	0.1	0.1	0.9	0.1	0.1
Person 6	Image 1	192	61	33.738	172	11.519	0.1	0.1	0.1	0.1	0.1	0.9	0.1
	Image 2	191	62	32.187	174	13.225	0.1	0.1	0.1	0.1	0.1	0.9	0.1
	Image 3	190	58	31.584	169	11.519	0.1	0.1	0.1	0.1	0.1	0.9	0.1
	Image 4	192	60	34.63	174	12.176	0.1	0.1	0.1	0.1	0.1	0.9	0.1
	Image 5	186	61	36.369	169	12.318	0.1	0.1	0.1	0.1	0.1	0.9	0.1
	Image 6	185	61	34.099	169	11.706	0.1	0.1	0.1	0.1	0.1	0.9	0.1
	Image 7	189	61	35.176	171	12.981	0.1	0.1	0.1	0.1	0.1	0.9	0.1
	Image 8	191	58	32.437	169	13.385	0.1	0.1	0.1	0.1	0.1	0.9	0.1
Person 7	Image 1	232	73	29.02	136	16.638	0.1	0.1	0.1	0.1	0.1	0.1	0.9
	Image 2	238	74	27.027	137	16.283	0.1	0.1	0.1	0.1	0.1	0.1	0.9
	Image 3	237	75	26.288	138	19.006	0.1	0.1	0.1	0.1	0.1	0.1	0.9
	Image 4	239	74	29.338	137	16.31	0.1	0.1	0.1	0.1	0.1	0.1	0.9
	Image 5	230	73	28.162	139	17.603	0.1	0.1	0.1	0.1	0.1	0.1	0.9
	Image 6	231	75	28.563	137	16.742	0.1	0.1	0.1	0.1	0.1	0.1	0.9
	Image 7	237	73	26.161	136	16.62	0.1	0.1	0.1	0.1	0.1	0.1	0.9
	Image 8	231	74	29.949	137	15.384	0.1	0.1	0.1	0.1	0.1	0.1	0.9

Figure 3-7 Sample for a CCNN input file that contains 56 images representing 7 individuals.

The NN has been used because it has proven to be a very useful tool for solving many real-life problems such as pattern recognition problems, and readapted to cope with the people authentication task. In addition, NN can reduce misclassifications among the neighbourhood classes. However, CCNN has a number of attractive features including a very fast training time, often a hundred times faster than a perceptron network [148]. Besides, it was found that the CCNN topology provides good performance in terms of convergence time and optimum topology [148].

On the other hand, the SVMs which were used for this work have a firm statistical foundation and are guaranteed to converge to a global minimum during training [157]. They are also considered to have better generalisation capabilities than neural networks. In addition, SVMs are known to be an excellent tool for binary classification problems, which were used for this work by decomposing the M-class problem into a series of two-class problems (one- against- all, see the seven output columns in table) by searching for the optimal separating hyper plane that provides efficient separation for the data and maximises the margin. In other words, SVM takes the closest vectors from both classes, assuming they are linearly separable, and maximises the distance between them in a hyper plane.

The KNN is a fast supervised machine learning algorithm which was used to classify the unlabeled testing set with a labelled training set. It has been used in many applications in the fields of data mining, statistical pattern recognition, image processing and many others [167]. Also, it has verified a good classification rate [168]. It was decided to

conduct further experiments to improve the recognition performance as well as to ensure of the effectiveness of the extracted features.

Intensive experiments were ran on 56 images to determine which of the machine learning algorithms is more suitable for the face recognition stage, the CCNN, SVM and KNN systems were optimised to find the best parameters and topology before making the comparisons.

3.6.1 Experimental work using CCNN

The number of input nodes in a CCNN is determined by the number of input features, extracted from the 2.5D face data, while the number of output nodes is determined by the number of different output classes. The direct input-output connections are trained using the entire training set with the aid of the back propagation learning algorithm. Hidden nodes are then added gradually and every new node is connected to every input node and to every pre-existing hidden node. Training is carried out using the training vectors and after each pass the weights of the new hidden nodes are adjusted [148]. Using a Neural Network with one hidden layer and with numbers of hidden nodes ranging from 1 to 10. For each number of hidden nodes, 8 results were generated, which were averages over 10 iterations carried out using the Jack-knife technique (80% randomly used for training and rest for testing). Consequently, a total of 45 images were used for training and 11 images were used for testing. The results generated were the number of hidden nodes, TP, FP, FN, TN,

Accuracy, Specificity and Sensitivity. Hence 100 learning and testing experiments were carried out for each case.

Four different experiments using 4 input features were carried out based on the Jack-knife technique for each CCNN configuration and the average TPR and FPR were recorded. After these experiments, the performance indicators were evaluated for every experiment, these indicators are TPR, FPR, FNR, TNR, accuracy, specificity and sensitivity. The averages of these indicators were found for each input features and are shown in Table 3-3.

Table 3-3 Average performance indicators using five input features.

Hidden Node	TPR	FPR	TNR	FNR	Accuracy	Specificity	Sensitivity
1	0.6	0.067	0.4	0.933	0.886	0.933	0.6
2	0.8	0.033	0.2	0.967	0.943	0.967	0.8
3	0.7	0.05	0.3	0.95	0.914	0.95	0.7
4	0.9	0.017	0.1	0.983	0.971	0.983	0.9
5	0.8	0.033	0.2	0.967	0.943	0.967	0.8
6	0.6	0.067	0.4	0.933	0.886	0.933	0.6
7	0.7	0.05	0.3	0.95	0.914	0.95	0.7
8	0.4	0.1	0.6	0.9	0.829	0.9	0.4
9	0.7	0.05	0.3	0.95	0.914	0.95	0.7
10	0.5	0.083	0.5	0.917	0.857	0.917	0.5

At the end of these experiments the best results were obtained for the following topologies:

- 5 input nodes (distance features) and 4 hidden nodes gave the best results for face recognition, where a value of 0.971, 0.983, and 0.9 was reached for accuracy, sensitivity and specificity respectively.
- 4 input nodes (distance features) with 2 and 7 hidden nodes, which gave 0.886, 0.933 and 0.6 for accuracy, specificity and sensitivity, respectively.
- 4 input nodes (ratio features) and 4 hidden nodes gave 0.914, 0.95 and 0.7 for accuracy, sensitivity and specificity respectively.
- 3 input nodes (ratio features) with 3 and 6 hidden nodes gave 0.857, 0.917 and 0.5 for accuracy, sensitivity and specificity, respectively.

3.6.2 Experimental work using SVM

The SVM experiments were carried out using the LIBSVM program to optimise the performance of both the kernel and its parameters, which were determined empirically because there are no known guidelines to help choose them[173]. For the work reported here, the Radial Basis Function (RBF) kernel was used. Hence, the shape of the RBF kernel is controlled by the parameter γ (Gamma).

To complete the SVM optimisation it was necessary to determine the value of γ. This was done by training and testing the SVM over 100 iterations. During each iteration, γ was incremented from 0.1 to 2.5 in steps of 0.2 and C was incremented from 1 to 20 in steps of 1. Hence, 240 experiments were conducted to test these values. The numbers of input features were 5 input features for 56 images. For each of those configurations twenty

experiments were carried out in order to find the best number of features and best parameters values.

The experiments using 56 images and 5 input features showed the best results were obtained when the γ value equals 0.5. Eight experiments for each individuals using 5 input features were carried out using the Jack-knife technique and the averages of performance indicators were found for every experiment and are shown in table 3-4, these indicators are TPR, FPR, FNR, TNR, accuracy, specificity and sensitivity.

Table 3-4 Average performance indicators for different input features.

Person Number	TPR	FPR	TNR	FNR	Accuracy	Specificity	Sensitivity
1	0.783333	0.114693	0.885307	0.216667	0.934320083	0.885306833	0.783333333
2	0.75	0.117763	0.882237	0.25	0.916118708	0.882237417	0.75
3	0.750002	0.20899	0.79101	0.249998	0.870505875	0.791010083	0.750001667
4	0.847223	0.178822	0.821178	0.152777	0.934200875	0.821178417	0.847223333
5	0.812275	0.196253	0.803747	0.187725	0.88010833	0.803746667	0.812275
6	0.965	0.200758	0.799242	0.035	0.882121	0.799241833	0.965
7	0.708333	0.413637	0.586363	0.291667	0.947348	0.586363333	0.708333333

3.6.3 Experimental work using KNN

In order to apply the KNN algorithm it was necessary to determine the parameter K which represents the number of nearest neighbours and calculate the Euclidean distances between the query instance and the training samples in order to predict the testing data classes. Then the distances are sorted based on the minimum difference to determine the K^{th} nearest neighbours.

The experiment was carried out on 56 images representing 7 individuals. Each individual has 8 images and using Jack-knife technique 45 vectors and 11 vectors were randomly selected for training and testing sets, respectively. The Euclidian distances between each vector of testing and each vector of training matrices were calculated. The experimental work showed that the best results were obtained when the K value equals 3 while the accuracy rate obtained were 0.96 for 56 images.

3.7 Conclusion

Because of the successes of machine learning appearance-based approaches to 2D face recognition, these methods were adapted for application to 3D face data. This required pre-processing work to prepare the data before applying the machine recognition phase. In general, it's not easy to compare or reproduce results of other research as many results are not reported using the same data. If there is a common database, such as the Face Recognition Grand Challenge (FRGC) database, different pre-processing operations may still be used for different methods, which make direct comparisons of the methods difficult.

Several recognition approaches are commonly applied in 3D face recognition investigations, for example, correlation, closest vector, PCA, SVM, EHMM and ICP approaches. In this work 3D face recognition with range data from FRGC Ver.2.0 data set was carried out using the machine learning techniques (CCNNs, SVMs and KNN), which had not been done before. In the machine learning stage, the training is carried out using the

training vector and the weights of the new hidden nodes are adjusted after each pass. Cascade correlation networks have a number of attractive features including a very fast training time, often a hundred times faster than a perceptron network.

Pre-processing operations are needed in order to localize the nose in the 3D face model to detect the face area then complete the pre-processing (i.e. cropping, hole filling, spikes removing). Pose correction may be needed to bring the face to the correct frontal orientation. Moreover, using ratios and distances feature didn't show a big difference in the results however, extracting distinctive features and combining various kinds of them could give better recognition rate than any single matching methods.

In the next chapter, an automated system to process the 3D facial data is developed. This system determines the symmetry profile for the face with few feature points along it, and extracts a set of effective profiles from the central region of the face. These feature points and profiles will be used for recognition and classification purposes.

CHAPTER FOUR

4. 3D FACIAL FEATURE EXTRACTION

Facial features in 2D or 3D images are usually classified as low level, high level, and semantic level features. Low level features are basic representations derived from images at any point in the image, such as the curvature, shape index, etc. High level features on the other hand are those related to human perception of faces, such as the tip of nose, eye corners, mouth corners, etc. By contrast, semantic level features at a more abstract level, such as gender, ethnicity, are often used to improve recognition rates and speed up the process of retrieving information from databases. Facial feature at the different levels provide different classes of information to analyze and recognize facial scans.

In this chapter, a novel method for the automatic processing of 3D facial data, which is obtained using 3D scanning devices, is presented. Here the input data can be in the form either of a 3D triangular facial mesh (containing the coordinate and connectivity information), or of a data point cloud. In the new approach, the first goal is to automatically determine a symmetry profile for the face. This is undertaken by computing the intersection between the symmetry plane (found by an automatic search) and the facial mesh, resulting in a planer curve that accurately represents the symmetry profile. Once the symmetry profile is successfully determined, a few feature points along the symmetry profile are computed. These features points are essential to the computation of other facial features, which can then be utilized to allocate the central region of the face and extract a set of

profiles from that region. These profiles can be used for recognition purposes as explained in the following chapter.

4.1 Revision of Feature Extraction

In face recognition applications based on either 2D or 3D images, alignment between the query and template is necessary in order to conduct matching between faces. In general, feature extraction based on landmarks provides an accurate and consistent representation for alignment and recognition purposes than other facial features. Such landmarks include the tip of the nose, eye corners, etc, as shown in Figure 4.1. However, these landmark points are not easy to recover automatically and incur different levels of complexity (e.g. the nose tip is usually the easiest to recover, while eye corners, for example, tend to be more difficult to acquire).

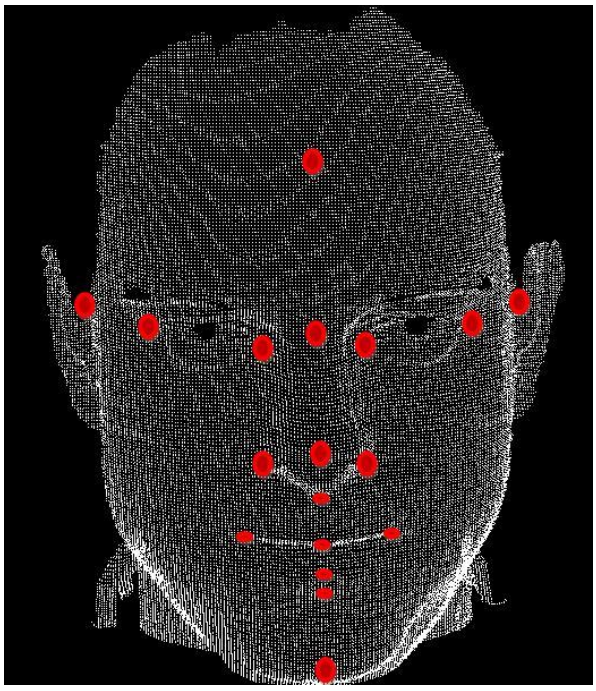


Figure 4-1 A subset of landmarks on a frontal view of a face.

For face recognition applications, 3D images captured using 3D scanning devices usually require some form of pre processing before they can be exploited for purposes such as the accurate and efficient extraction of certain facial feature. For example, given 3D facial geometric data, it is usually necessary to identify from the data the central region of the face known as the facial mask. The facial mask of a 3D face is the region of interest for various applications such as 3D face recognition and authentication.

A common method for determining the facial mask from raw 3D facial data and extract certain facial features is based on measuring the surface curvature [174], such as Gaussian curvature. However, such techniques are often prone to errors since the computation of accurate Gaussian curvatures requires sufficiently accurate data which provides relatively smooth surfaces and such data cannot be always be made available by the available scanning devices.

3D data may have a coordinate value missing in some rows and columns (visible as vertical or horizontal stripes in the face examples in Figure 4.2); these need to be automatically identified and replaced by interpolation at the pre-processing stage. Another of the challenges in processing and allocating certain facial features for given raw 3D facial data, as illustrated in Figure 4.2, is due to the scanned image containing unwanted geometry (such as neck, shoulders and hair). This need to be identified and discarded at the pre-processing stage. Thus, in some applications, semi-automatic approaches have been utilized to overcome this challenge [30, 71, 175, 176]. For example in [176] seven

landmark points have been manually selected. Similarly in [30] nose tip and eyes corners were manually identified to register faces.

Another challenging issue in processing and characterizing 3D raw facial images is that a captured face could have any one of a range of orientations or poses, and hence it becomes crucial to determine or standardise the facial pose, to simplify the allocation of characteristic facial features. Various techniques are used to recover or estimate the facial pose. For instant, Principal Component Analysis (PCA) was used in [71] to estimate the pose.

Figure 4-2 Example for 4 different scanned images of human faces, with different poses.

Other approaches are based on some reasonable assumptions that are made about pose variations. For example, in [177] it is assumed that the input data contain a frontal view of the face with pose variation limited $\pm 15^{\circ}$ along the x, y and z axes. Additionally, scanned images usually are very dense datasets and therefore simplification algorithms need to be deployed in order to reduce the size of the captured data to a computationally feasible status.

Many real world objects have symmetry characteristics, where the symmetry could be rotational, reflective or translational. A key component of a face is its symmetry characteristics associated with a plane which divides the face into two halves related by a reflective transformation. Faces rarely if ever possess exact reflection symmetry but it is a useful approximation to assume. The reflection symmetry of facial data can be destroyed by non-uniform data boundaries or by facial features (e.g., hair) so it is not a trivial matter to estimate the plane of symmetry. Various techniques have been used to identify the symmetry plane of 3D scanned images[71, 76, 77, 178, 179]. For instance, Sun et.al.[77], assumes that the symmetry plane passes through the centre of mass of a given object and uses an Extended Gaussian Image (EGI) based technique to detect reflection and rotational symmetry of objects. Similarly Pan et al.[76] used an EGI technique similar to [77, 125] to detect the symmetry plane of the face. However, for facial data such assumptions might not hold, especially since 3D facial data acquired by 3D laser scanners might be highly asymmetric since it could contain noise, and undesired geometry such as neck and shoulder (see Figure 4.2). Wu et.al. [125] used a profile matching approach for face authentication

and for their symmetry analysis an initial position of the symmetry plane has to be interactively identified. Colbry and Stockman [179] identified the symmetry plane of a facial scan by matching that scan with a mirror image of itself using face surface alignment algorithm assuming that pose variation ranges from 10 to 30 degrees in the coordinate system.

Several other analysis techniques can be found in the literature which address the detection of the symmetry of a human face and other feature extraction, such as the use of principle axes of inertia of the object [180],the extended Gaussian image of the object [77],and the ‘mirror’ plane method [181]. Another more general approach to determine the symmetry plane of 3D object is by minimising the symmetry value over all possible planes[182].A reflective symmetry descriptor[183], and point signature techniques have also been used to allocate facial features for face recognition purposes and 3D shape registration [93].

To summarize, the automatic processing and characterization of 3D scanned images is still considered a challenging problem with a relatively few papers in the public domain addressing it. Semi automatic approaches, where user intervention is assumed, or initial assumptions are made about the pose of the face are often encountered to simplify the problem.

4.2 Point Cloud data and Object and VRML files

Viewable 3D image files of facial geometry are produced either as a point clouds or as polygonal meshes. A point cloud is simply a set of n vertices $V = \{p_i | p_i \in R^3, 1 \leq i \leq n\}$. On the other hand a polygonal geometry uses points and faces to define objects. A triangular mesh S includes the set of vertices with its x y z coordinates, which defines the position of the vertices in three dimensions. In addition, it includes information about the set of facets of these vertices, and is defined as $F = \{(p_i, p_j, p_k) | \ p_i, p_j, p_k \in V\}$.

A normalized and registered raw mesh means that all values of the vertices are scaled to be in the range between 0.0 and 1.0. In addition, the facial data is aligned with the Cartesian coordinate, such that the nose tip is located at the origin and the face is looking towards the positive z-axis as shown in Figure 4.3.

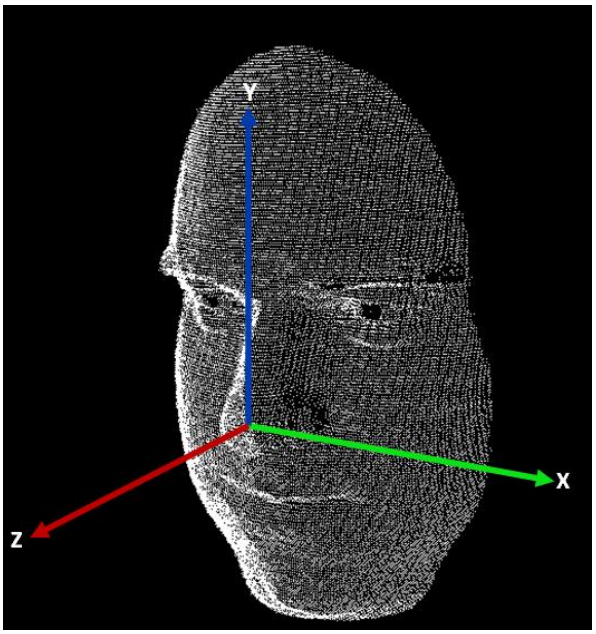


Figure 4-3 Normalized and registered facial scanned image.

Object files (OBJ) and the Virtual Reality Modelling Language files (VRML) are formats for storing a description of the surface of a 3D object, composed of triangles or higher degree polygons.

Object files define the geometry and other properties of objects for Wave Front's advanced visualizer. Object files can also be used to transfer geometric data back and forth between the advanced visualizer and other applications. Object files can be in ASCII format, when they have the extension OBJ, or in binary format, when they have the .mod extension. The OBJ file format supports both *polygonal* objects and *free-form* objects. Polygonal geometry uses an element such as points, lines, and faces to define objects while free-form geometry uses curves and surfaces [184].

On other hand, VRML files are designed particularly for use with the World Wide Web [185]. URLs can be associated with graphical components so that a web browser might fetch a web page or a new VRML file from the Internet when the user clicks on the specific graphical component. VRML files are commonly called "worlds" and have the WRL extension. Although VRML worlds use a text format, they may often be compressed using gzip so that they can be transferred over the internet more quickly[185].

In order to have a clear face representation, the cloud points must be joined together to form the mesh, thus, a better face observation is obtained. 3D face data in the FRGC dataset are provided in ABS (ASCII raster) format and in the current work OBJ and VRML files have been generated from FRGC database in order to use them in further work. And to make it available for other researcher to use

Vertex data provides coordinates for geometric vertices, and the right-hand coordinate system is used to specify the coordinate locations. The statement syntax for the geometric vertex is $\mathbf{v}\ \boldsymbol{x}\ \boldsymbol{y}\ \boldsymbol{z}$. Where $\boldsymbol{x}\ \boldsymbol{y}$ and $\boldsymbol{z}$ are coordinate values. These are floating point numbers that define the position of the vertex in three dimensions. When vertices are loaded into the visualization system, they are sequentially numbered, starting with 1. These reference numbers are used to identify geometric vertices and to create faces using the statement syntax $\mathbf{f}\ \mathbf{v_i}\ \mathbf{v_j}\ \mathbf{v_k}$ where $\mathbf{v_i}$,$\mathbf{v_j}$ and $\mathbf{v_k}$ represent the reference numbers for vertices. Notice that every three vertices generate a triangle; providing that they are not on a straight line. This method of creating faces is called triangulation.

The triangulation approach used is incremental, i.e. it looks for the nearest three points and joins them together to form a triangle then it moves to the next set and so on until a complete mesh that covers the face is formed. Figure 4.4 shows portions of an OBJ file and a VRML file that contain vertices and facets information. Figure 4.5 shows an example of an OBJ files before and after applying the triangulation method, which also presents a comparison between the point clouds face and our simplified model.

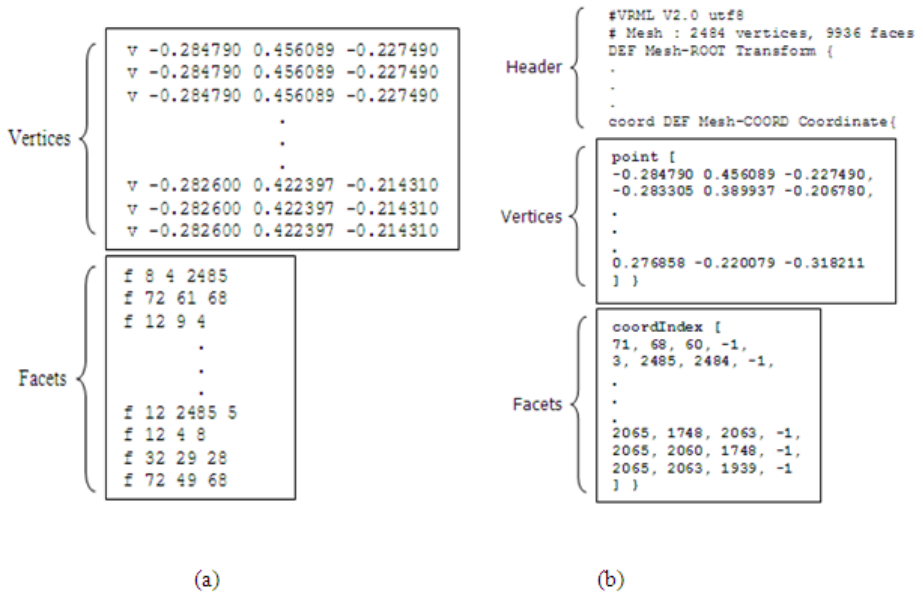


Figure 4-4 Samples for a portion of an OBJ file (a) and a VRML file (b) that contain vertices and facets information.

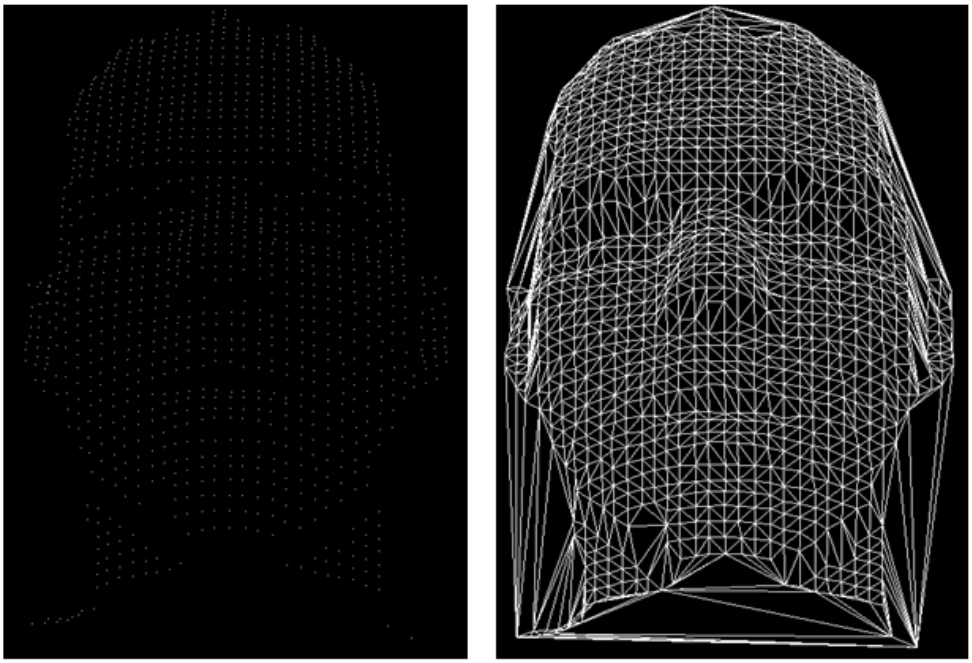


Figure 4-5 Face comparisons between: a) the face point clouds, and b) after applying the triangulation method.

Moreover, triangulation can be applied to any simple polygons with a number of vertices n [186]. The possibility of applying triangulation can be justified by the fact that every polygon has a diagonal if the polygon contains at least one vertex [186]. Knowing that every polygon has at least one vertex [186] and has at least one diagonal, a polygon with $n \geq 4$ can be divided into triangles by placing those diagonals [186](See Appendix B for more details).

There are many ways for triangulating a polygon but all of them share the fact that the number of the generated diagonals $ND = n - 3$, and the number of resultant triangles $NT = n - 2$ [186].

4.3 Pre-processing and Registration Phases

As indicated earlier in this chapter, the human face is a very interesting but complex pattern with characteristics, which include the following:

- The face object is a natural pattern unlike a simple artificial pattern such as a cube or pyramid.
- Although the face pattern has a small number of elements, the structures of elements like the eyes, nose, mouth, chin ... etc. are complex.
- The exact positions of these components with respect to each other are given by geometrical relationships and their arrangement within the face is constant.

- The extraction of face features like lines, curves and corner points are not trivial operations.
- The artificial additions like spectacles, earrings and make-up as well as natural additions like hairstyles, moustaches and beards increase the complexity of the face pattern.

The 3D faces provided by the FRGC version 2.0 dataset are noisy and contain spikes. Therefore, the pre-processing step is an important one for face recognition because it determines the regions of useful data and removes noise and background regions. This step aims at reducing the amount of data to be processed, to remove noise spikes, to fill holes and to produce useful structural information about object boundaries.

The raw face data are assumed to be stored with known adjacency relations among the 3D points and the faces are captured roughly in the normal top-down posture (the front direction of the face can be somewhat arbitrary), in the FRGC v2 and 3D-BUFE databases. Many commercial 3D scanners can generate such data [67]. The following subsections discuss the pre-processing steps including removal of sharp spikes, filling of holes, smoothing of mesh surface and facial region extraction.

4.3.1 Removal of sharp spikes

A noise spike is defined as a random variation in values that may corrupt the data content, and usually occurs during data capture, transmission or processing [33, 85]. Figure 4.6 shows examples of faces with sharp spikes.

Once the OBJ and VRML files have been created, outlier points causing spikes are removed from these files using two approaches. Figure 4.7 illustrates the relation between pixel$(i, j)$and its neighbours $(i+1, j), (i-1, j), (i, j-1), (i, j+1), (i+1, j+1)$ which our removing spikes techniques are based on.

The first removing spikes approach involves calculating the distances around each pixel(i, j), the distances are arranged in order of magnitude to determine if the three distances are much greater than the remaining three(by certain value such as 20 pixels). If so, then the current pixels are considered to be part of a spike which is eliminated by identifying and correcting the outlying point. The pseudo-code for this procedure is given in Figure 4.8.

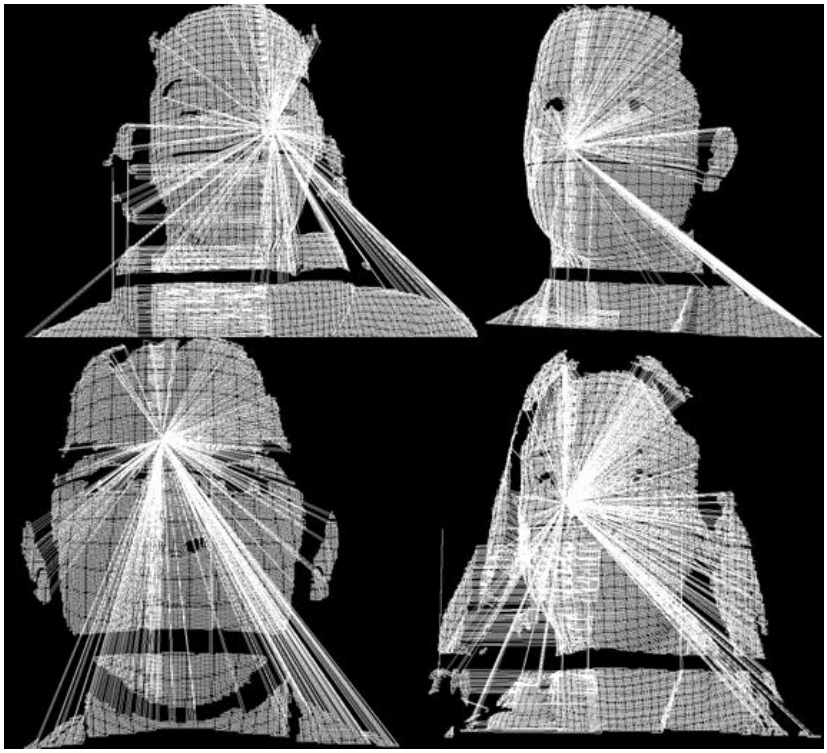


Figure 4-6 Four different 3D faces with sharp spikes.

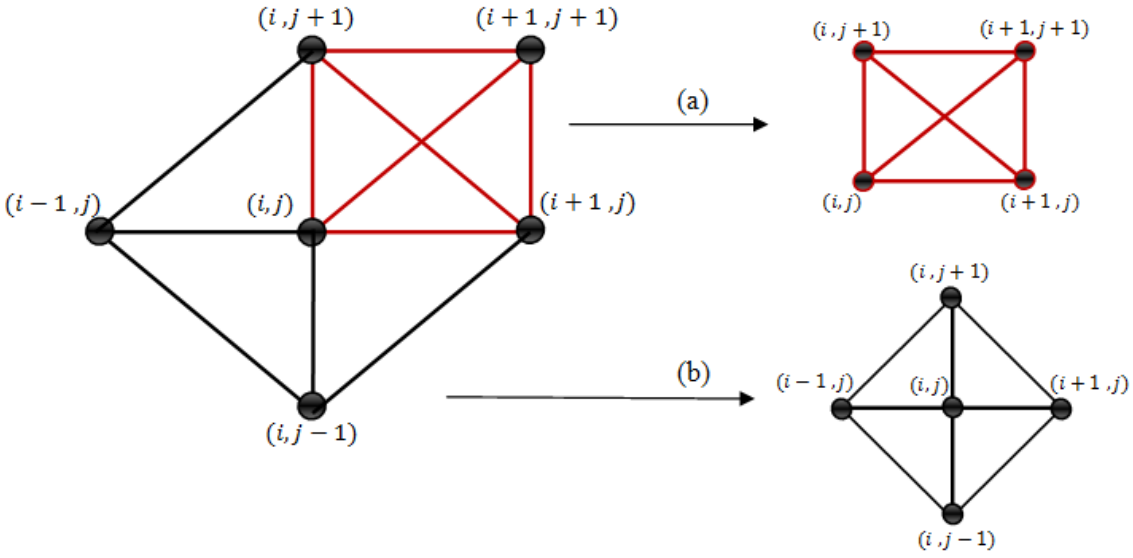


Figure 4-7 The relation between pixel$(i, j)$and its neighbours (a) correcting the distances between points approach. (b) correcting the angles between points approach.

Algorithm 1: Remove spikes by correcting the distances between points.

Step 1: Calculate the variance distances around each pixel (i,j).

Step 2: Compute the six distances between four points: $(i, j), (i, j + 1), (i + 1, j + 1), (i + 1, j)$

Step 3: Order the distances

Step 4: Determine which of the three cases applies

(i) All distances are similar

(ii) Four distances much greater than the other two

(iii) Three distances much greater than the other three

Step 5: If case (iii) applies // the current pixels are considered to be part of a spike

5.1 Initialise order of distances,

5.2 Reorder them in ascending order and keep an index of their original positions.

5.3 Delete the face either because at least two points coincide or because one of the distances is too large.

Step 6: Check which point is not present in the three lines with smallest distance,

6.1 Initialise counter for the points

6.2 Use the point opposite to the one to be corrected as a reference.

6.3 The order of points is: point opposite one to correct, two on both side and the point to correct.

6.4 Initialised the array and overwrite the interior points to smooth,

6.5 Move origin to point xR, yR, zR

6.6 Move origin back to original position.

Step 7: Return the sum of the sides of the triangle defined by the three points in 3D space.

Figure 4-8 The pseudo-code for remove spikes by correcting the distances between points algorithm.(see case a in figure 4.7)

However, in some cases calculate the distances around each pixel fail to determine the spike pixel, especially if this pixel close to the face mesh and the distances between the points are almost equal. To tackle this problem we have added a new technique to remove the sharp spikes by correcting the angle between points. In this method, we considered each of the lines joining point (i, j) to its four neighbours then determined the average angle between the valid lines, if the angle is less than a certain value such as 15 or 30 degrees we replaced the point under investigation by the average of the valid neighbours. Figure 4.9 shows the pseudo-code for this algorithm.

Algorithm 2: Remove spikes by correcting the angles between points.

Step1: Consider each of the lines joining point (i, j) to its four neighbours (i, j+1), (i+1, j), (i, j-1), (i-1, j)

1.1 Calculate the angle between the two lines which are connected between the two vertices (i+1, j) and (i, j+1) to the point (i, j).

1.2 Calculate the angle between the two lines which are connected between the two vertices (i, j+1) and (i-1, j) to the point (i, j).

1.3 Calculate the angle between the two lines which are connected between the two vertices (i, j-1) and (i-1, j) to the point (i, j).

1.4 Calculate the angle between the two lines which are connected between the two vertices (i+1, j) and (i, j-1) to the point (i, j).

Step 2:If any angle is less than 15 degree

2.1 Replace point by average of the valid neighbours

Figure 4-9 The pseudo-code for remove spikes by correcting the angles between points algorithm (see case b in Figure 4.7).

4.3.2 Filling in missing data

Data are inevitably missing due to the complexity of the scanned region or to an imperfect scanning process. As mentioned in the previous subsection, holes and missing points in the data can also occur during the removal of sharp spikes. Missing scanned data cause holes in the triangular mesh created, but a hole-free mesh model is a prerequisite for the feature extraction process.

Although a number of hole filling algorithms have been investigated such as using curvature information[187], they enable the filling of holes only in the smooth regions of a model. They are not always robust in case of complex holes.. In this work and in order to fill in these holes, two methods were used. The first method is linear interpolation which was used to fill in gaps in X or Y data in cases where the Z value exists. The second method was a cubic interpolation which was used to fill the gaps in Z data as explained early in Chapter 3 section 3.4.

Linear interpolation is a method of curve fitting using linear polynomials. It has numerous applications including computer graphics [188]. Figure 4.10 illustrates the linear interpolation between two known points (x_0,y_0) and (x_1,y_1).

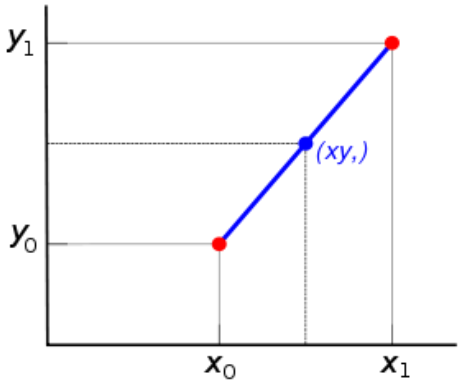


Figure 4-10 Linear interpolation between two known points

If the two known points are given by the coordinates (x_0, y_0) and(x_1, y_1), the linear interpolant is the straight line between these points and the value y at x is found on this line. For a value x in the interval (x_0, x_1) , the value y along the straight line is given from the equation 4.1.

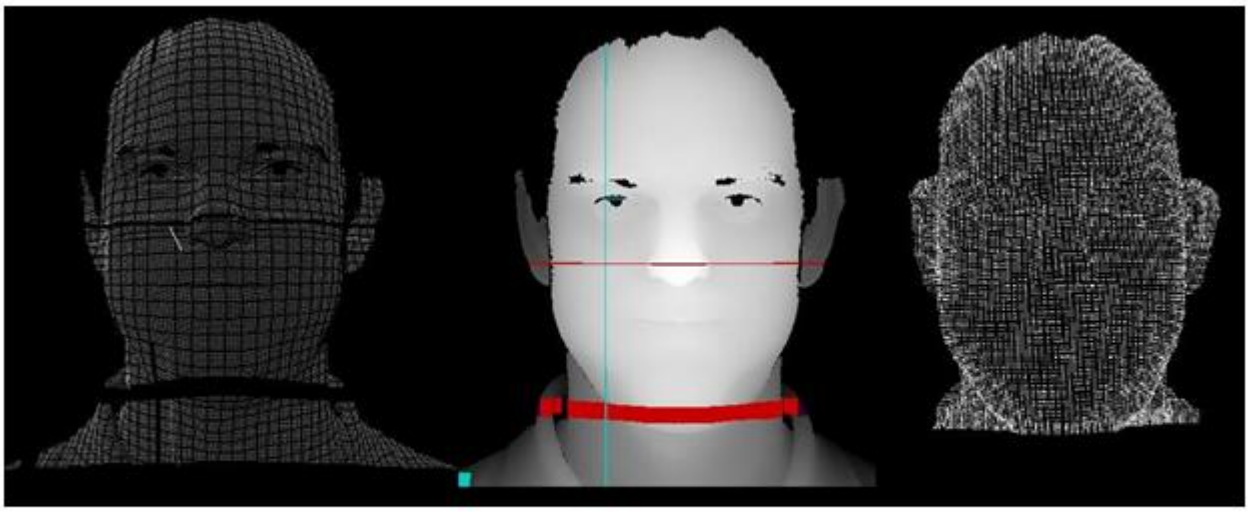
$$\frac{y - y_0}{x - x_0} = \frac{y_1 - y_0}{x_1 - x_0} \qquad (4.1)$$

which follows from the geometry of Figure 4.10. Solving this equation for y, which is the unknown value at x, gives

$$y = y_0 + (x - x_0)\frac{y_1 - y_0}{x_1 - x_0} \qquad (4.2)$$

Which is the formula for linear interpolation in the interval (x_0, x_1). Outside this interval, the formula is identical to linear extrapolation. On the other hand, the linear interpolation on a set of data points $(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)$ is defined as the concatenation of linear interpolants between each pair of data points. Figure 4.11 shows filtered X , Y and Z data

using linear interpolation to fill in missing X and Y data (high-lighted by red and cyan colours) and cubic interpolation to fill in missing Z data (for example near the eyes).



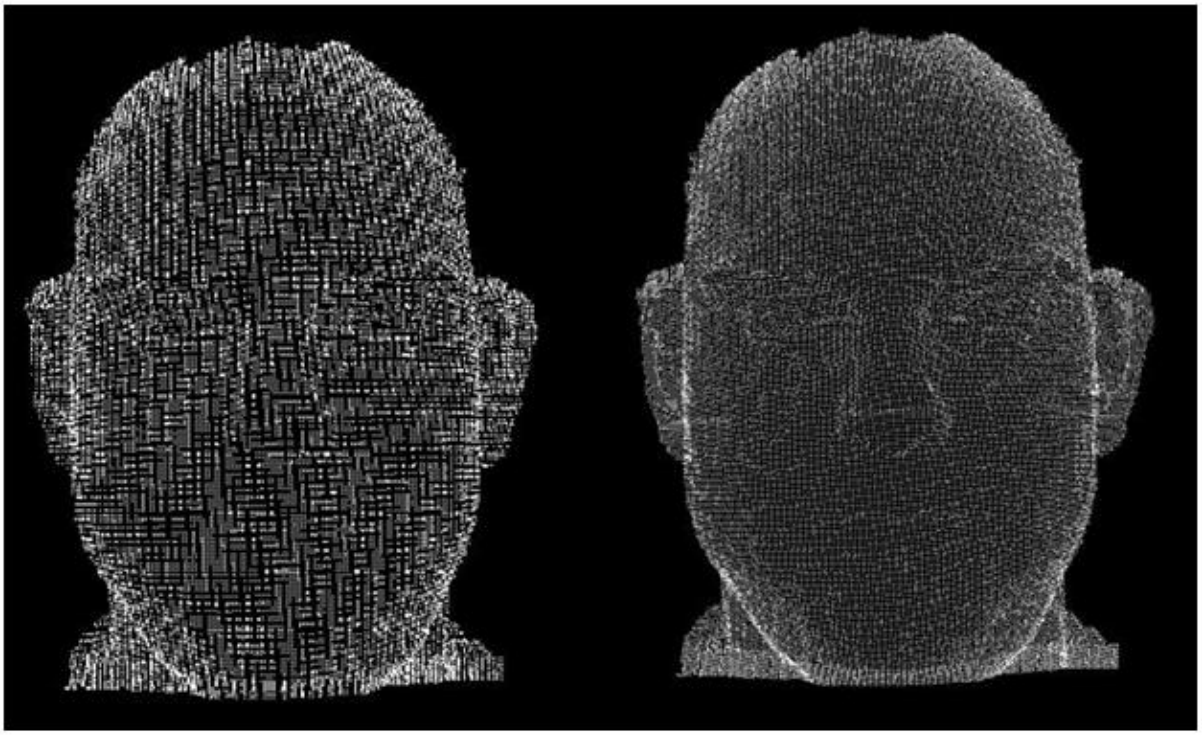
(a) (b) (c)

Figure 4-11 Filtered X and Y and Z data using linear interpolation and cubic interpolation, respectively, where a) shows X, Y and Z holes, b) shows X and Y holes filled (in red and cyan) c) filing the z holes.

4.3.3 Smoothing 3D data surfaces

Finally smoothing techniques are applied to enhance the hole triangles. The newly created vertices and triangles are added to their respective lists and the topology information is updated. A Gaussian filter is used for smoothing the Z data with a variance of ¾, so that the original value and the closest neighbours in the block have most influence on the filtered value.

The 3D smoothing function was used in these experiments. This function smoothes the input using a convolution kernel which is a ‘Gaussian filter’. The size of the convolution kernel is specified as [3 3 3] and the convolution kernel standard deviation (SD) has a default value of 0.65. Figure 4.12 shows the data before and after applying the smooth function.



(a) (b)

Figure 4-12 The data before (a) and after (b) applying the smooth function.

4.3.4 Extraction of facial regions

Previously, an effective method of facial region extraction, based on a symmetry plane analysis of the whole 3D face data has been proposed by Pan et al.[189]. This algorithm and its variants have been adopted by many other researchers in their face registration work [71, 117, 179]. This class of methods makes the assumption that the human face is bilaterally symmetric. Nevertheless, small pose variations, different hairstyles and facial expressions can all lead to violations of this assumption. This means that the symmetry plane results for multiple facial scans from the same person are not always stable. As explained in Chapter 2, we are inspired to find a more rigid part on the human face for the symmetry plane analysis. Compared to the whole face region, intuitively the nose region has several advantages. For instance, the human nose does not change significantly under most facial expressions.

In the automatic 3D face features extraction method proposed here, the first stage is to extract a facial region from a set of raw 3D face data and removing all unwanted data (e.g. shoulders, neck, hair, hat, etc) by locating the four boundaries of the face region coarsely and then crop an initial face region. The second stage is to compute an estimate of the symmetry plane of the extracted 3D face region and find the nose region in order to extract the tip of the nose, the nose bridge and the bottom of nose. The third stage is to locate the eye plane in order to find the inner eye corners. The fourth stage is to locate effective curves around the nose area in order to increase the number of extracted features.

In this section, we outline the method for estimating the symmetry plane for the face data. The approach adopted to solve this problem is as follows; define a numerical measure for the degree of reflection symmetry for face data about a given plane, specified in terms of a small number of parameters and search for the parameter values which minimises the measure. A reflection measure could be defined by computing the perpendicular distances of the data points on either side of the chosen test plane, summing the distances on either side of the plane and taking the difference between the two sums. This will be zero for a reflection plane but can also be zero in other circumstance (such as the case of the diagonal of a rectangle in 2D). Figure 4.13 shows examples for such problem. To avoid this case, the reflection measure adopted here is the sum of the individual differences between two distances on either side of the test plane, on the same perpendicular line. This measure will be zero for the true reflection plane and generally larger in other cases. It is true that for a complicated surface, such as a face, another non-reflection plane can give a minimum (see Figure 4.13 for two examples). The result of the search can be checked (manually, or automatically by analysing the frontal line profile) and, if necessary, the search repeated from a new starting position. In practice this process has been found to give reliable results.

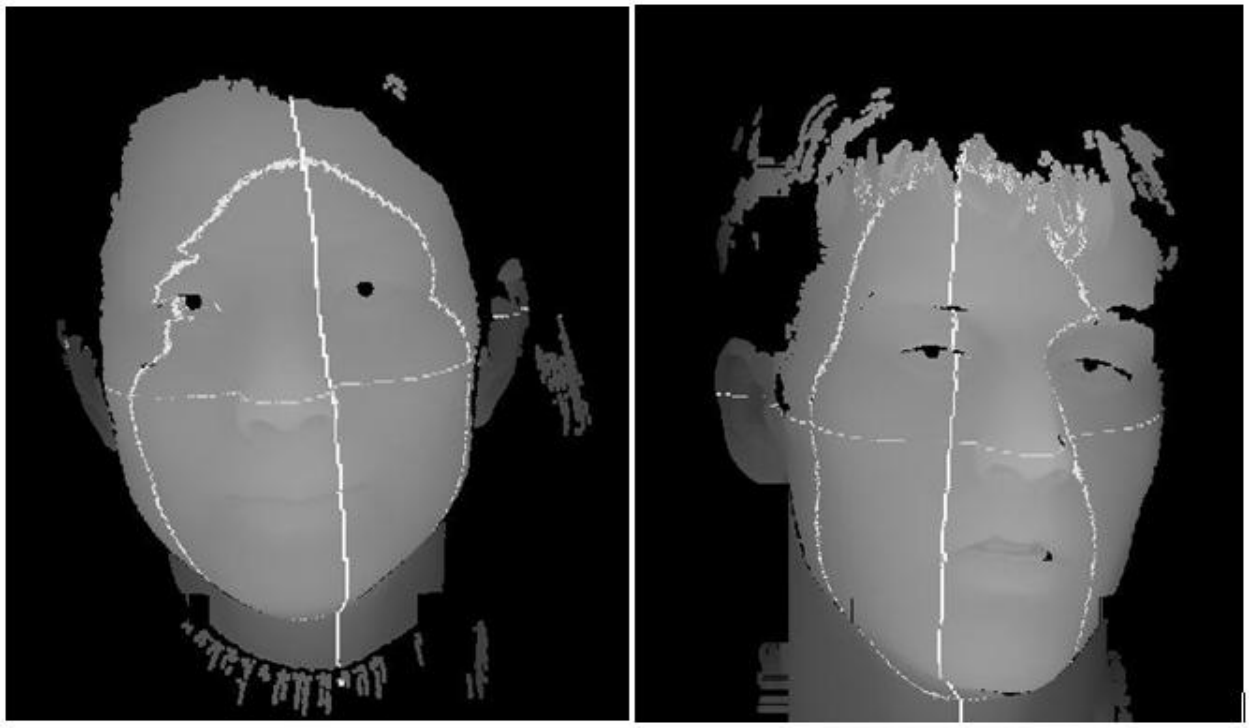


Figure 4-13 A false reflection measure results after summing the perpendicular distances of the data points on either side of the chosen test plane, and taking the difference between the two sums.

To implement this approach it is necessary to take account of the possibilities that the face data will have different densities of points in different regions. For example, there could be holes in the data on one side but not the other and the boundary to the captured data could not be reflection symmetric. The presence of data noise will also cause the symmetry measure to be greater than zero but a minimum at the plane of reflection.

To handle the first two problems in implementing a reflection measure, instead of taking all data points into account a set of lines parallel to the normal of the test plane, and spaced equally apart in the orthogonal directions are defined. Each of these lines intersects the face

data on either side of the test plane and two average distances of these intersections are computed. If only one can be computed (because of missing data for example) then the contribution of that line to the summed difference distances is ignored in the measure. The third problem is tackled by applying the symmetry measure only to the part of the face in front of a plane through the data centroid orthogonal to the test plane. This removes the irregular data boundary towards the back of the head.

The approach then is to define an initial test plane approximating the true reflection plane and the perpendicular data bounding plane and to compute the symmetry measure. An automatic search is then conducted (using the simplex minimisation algorithm), adjusting up to three parameters defining the location and orientation of the test plane to find the minimum condition.

Since the head pose for 3D face data used in this work are all near frontal (the angle for pose is within $\pm 15°$, of straight forward) Principle Component analysis (PCA) has been applied to define the starting symmetry and data bounding planes.

For example for a given three dimensional data set P of N points $\mathrm{p_i} = (\mathrm{x_i}, \mathrm{y_i}, \mathrm{z_i})$, to apply PCA the first step is to subtract the mean values $\bar{X}, \bar{Y}\ and\ \bar{Z}$ across each dimension x, y and z respectively. The next step is to construct the covariance matrix, which in this case would be 3×3 matrix, defined as follows:

$$\mathrm{cov} = \begin{Bmatrix} \mathrm{cov}(x,x) & \mathrm{cov}(x,y) & \mathrm{cov}(x,z) \\ \mathrm{cov}(y,x) & \mathrm{cov}(y,y) & \mathrm{cov}(y.z) \\ \mathrm{cov}(z,x) & \mathrm{cov}(z,y) & \mathrm{cov}(z,z) \end{Bmatrix} \quad (4.3)$$

where the matrix element is defined as:

$$\mathrm{cov}(x,y) = \sum_{i=1}^{n} \frac{(X_i - \bar{X})(Y_i - \bar{Y})}{n} \quad (4.4)$$

The next step is to compute the eigenvalues and eigenvectors of the covariance matrix. This can be accomplished by solving for the roots of the characteristic polynomial equation resulting from the covariance matrix [190] and then substituting these back into the eigenvalues equation and solving for the corresponding eigenvectors. Alternatively, a standard algorithm, such as the Jacobi algorithm, can be used to give the eigenvalues and eigenvectors directly.

The principal directions P_X, P_Y, P_Z resulting from the face data are similar to the X, Y, Z axes respectively, to which the original data are referenced. The data used to define the reflection measure are bounded by the $P_Z = 0$ plane (data in front of the face centroid) and the starting reflection plane is the $P_X = 0$ plane. Figure 4.14 shows an example to which PCA has been applied. The intersections of the three principal planes with the data are curves. The curve closest to the vertical is the symmetry profile corresponding to the starting estimate of the reflection curve. The closed curve shows the position of the initial data bounding plane. It is close to the reflection symmetric when the estimated reflection plane is correct and can be used to check this. The third line shows the intersection of the data and the principal plane closest to the horizontal which is of less interest at this point.

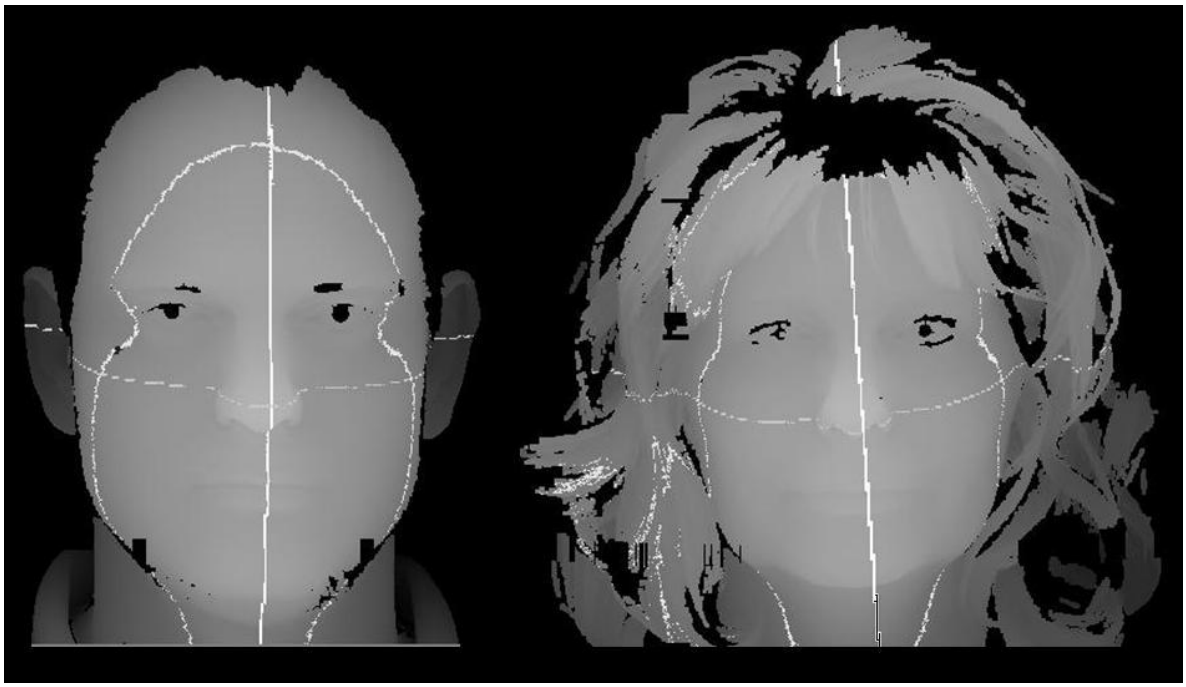


Figure 4-14 The face data and intersection with all three principle planes; hence the initial symmetry plane divides the face asymmetrically.

4.4 Symmetry plane identification

To create the algorithms used to calculate the intersection between a plane and a face surface data set and the reflection measure for a given set of data points and a plane, the geometrical relations described in below are used. Figure 4.15 shows a point ***P*** (X, Y, Z) and a plane through the origin defined by its normal vector $\hat{a}$ (A, B, C). The vector $\boldsymbol{\lambda\hat{a}}$ from the plane to ***P***, perpendicular to the plane is also shown, where λ, the signed perpendicular distance from point to plane is given by

$$\lambda = \hat{a} \cdot P = Ax + By + Cz \qquad (4.5)$$

The condition that a point $\boldsymbol{P}$ is within a distance ±d of the plane is

$$|AX + BY + CZ| < d \tag{4.6}$$

and if the plane is not through the origin but is displaced a distance *D* from the origin in the direction â**,** this condition becomes

$$|AX + BY + CZ - D| < d \tag{4.7}$$

The vector $\boldsymbol{r}$ in the plane to the foot of the perpendicular from the point P to the plane is given by:

$$\boldsymbol{r} = (X - \lambda A,\ Y - \lambda B,\ Z - \lambda C) \tag{4.8}$$

Equation (4.7) is used to find the points in the face data within a specified distance (about 1 mm) of a particular plane (since few if any data points will be exactly in any given plane) and when used with two perpendicular planes finds the points within a square tube about the line of intersection of the planes. Similarly, equation (4.8) is used to project the points close to a particular plane into the plane, when finding the intersection of face data with a particular plane. The resulting points are then guaranteed to be on a plane curve.

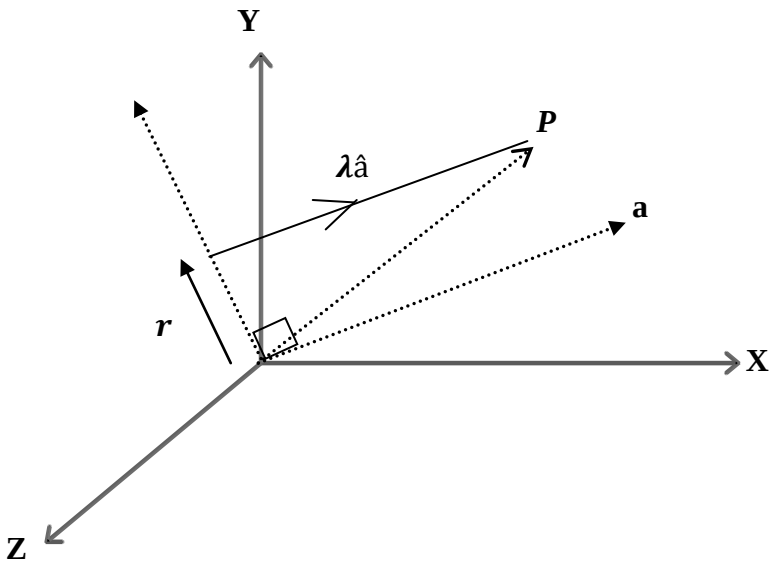
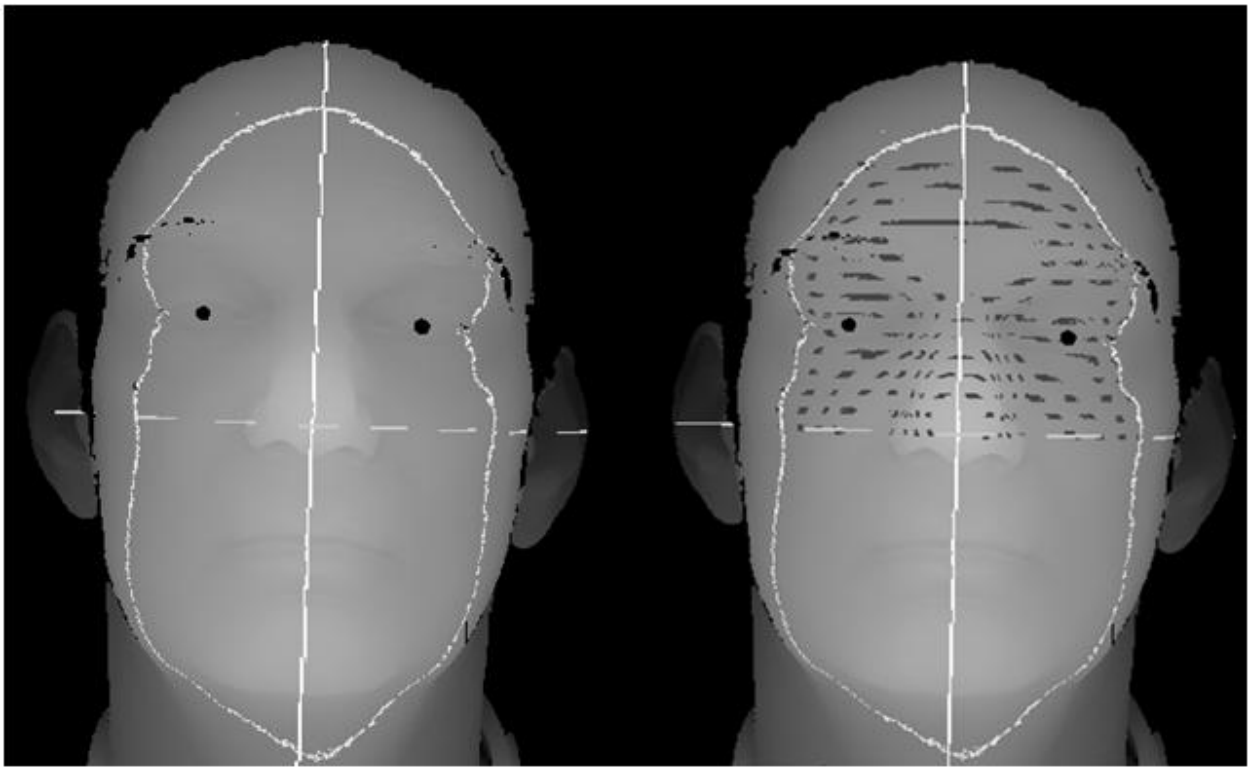


Figure 4-15 The normal vector a (unit vector â) to a plane through the origin is shown together with an arbitrary point in space P. The vector r is the foot of the perpendicular λâ from the point P to the plane.

Figure 4.16 shows an example of face data with *Z* values relative to the data centroid displayed in grey. The intersections of the face data with the principal planes are the three white curves in Figure 4.16 and the points used to calculate the symmetry measure are shown in darker grey than the neighbouring face data. These are the intersections of the face data with a regularly spaced set of tubes with axes in the P_Y directions.



(a) (b)

Figure 4-16 a) Face data with Z values displayed in grey and intersections of the principal planes with the data shown in white. B) The patches of darker grey on the upper half of the face show points within a square tube included in calculating the reflection measure.

The reflection measure for given face data and test plane can be regarded as a function of the parameters which specify the position and orientation of the test plane. These are translations in the P_X direction, and rotations about the P_Y and P_Z axes. The downhill simplex method [191] is employed to find the parameter values which minimise the reflection measure. This method was chosen because it is a multi-dimensional function minimisation routine which requires only function evaluations and not derivatives. It is not the fastest method available but it can be extremely robust [191]. The downhill simplex

method must be started not just with a single point, but also with N + 1 vertices, defining an initial simplex. Taking one of these vertices $\mathbf{P_0}$ as our initial starting point , then we take the other N points to be

$$\mathbf{P_i} = \mathbf{P_0} + \boldsymbol{\lambda_i}\hat{\mathbf{a}}_\mathbf{i} \qquad (4.9)$$

where the $\hat{\mathbf{a}}_\mathbf{i}$ are N unit vectors, and where $\boldsymbol{\lambda_i}\lambda$ are constants, which are the problem's characteristic length scales.

The Simplex algorithm consists of four movements of an original simplex: Reflection, Reflection and expansion, Contraction and Multiple contraction. Suppose we have a simplex (a)(b)(c). The basic steps in the simplex method are shown in Figure 4.17, assuming that (a) is the highest and (c) is the lowest.

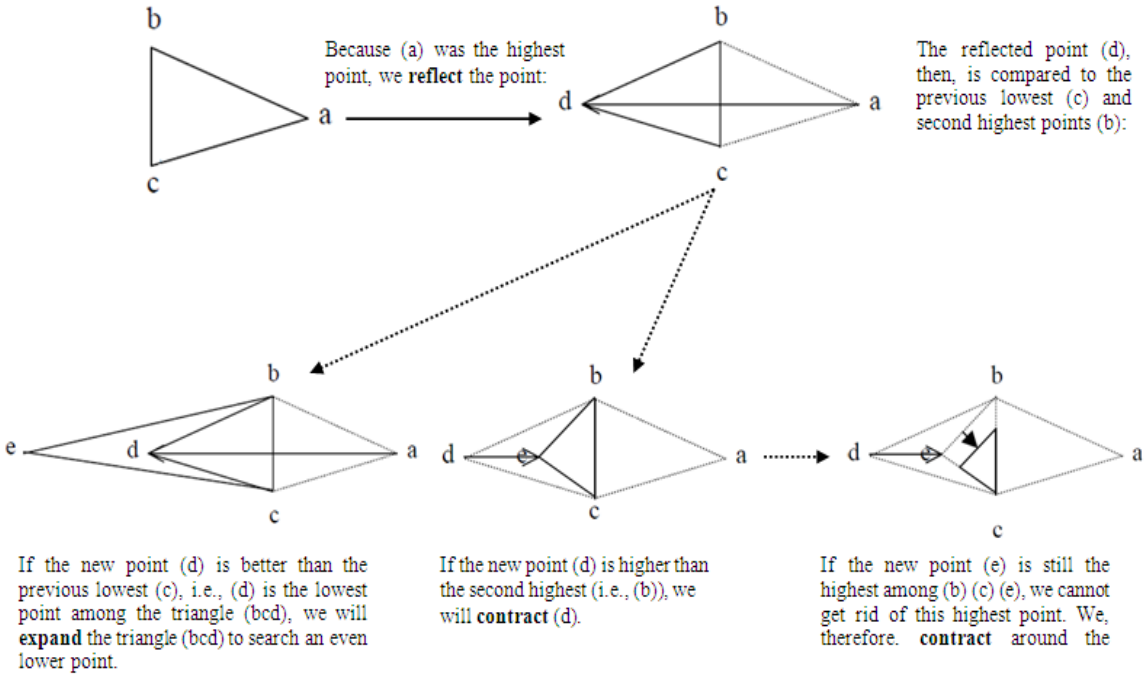
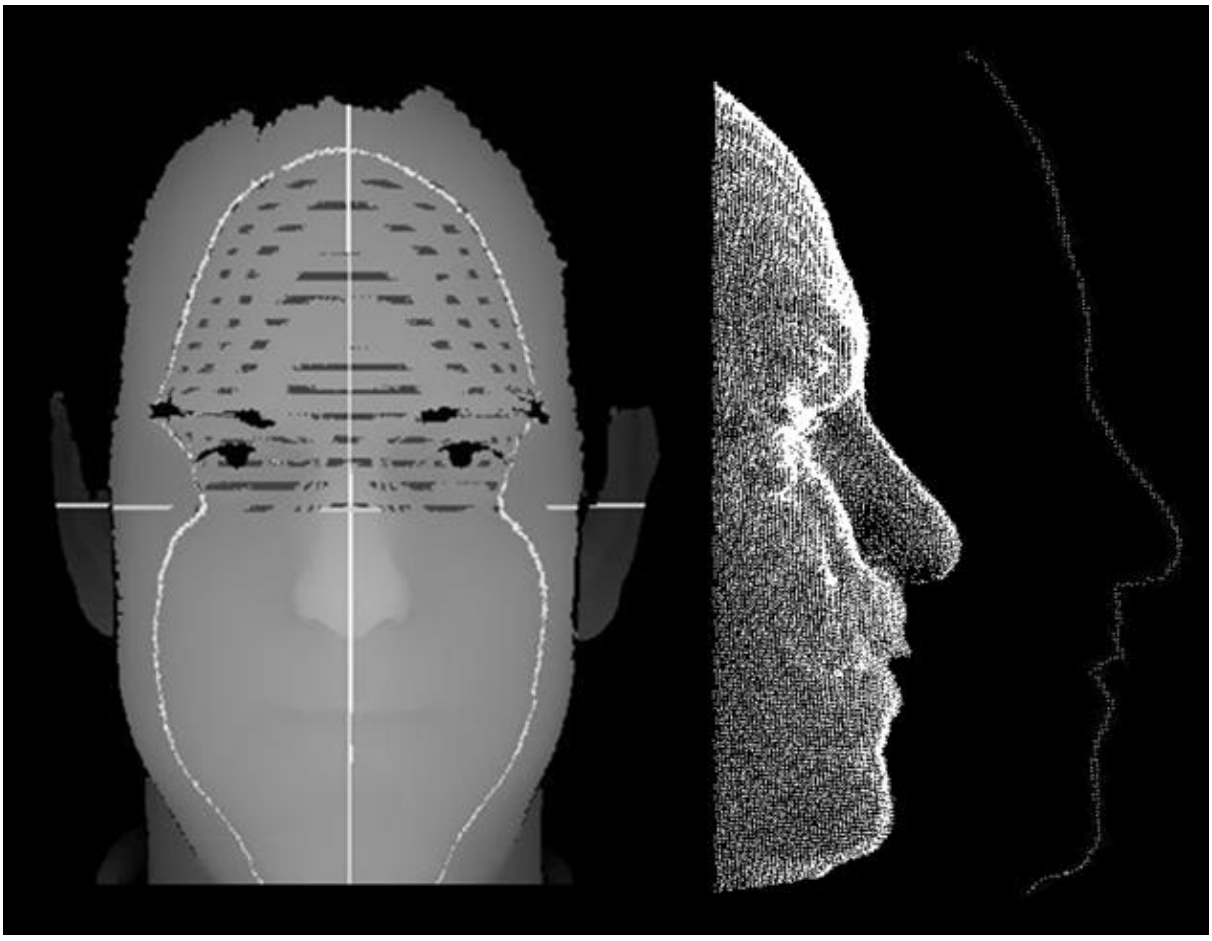


Figure 4-17 The simplex method movements

This summarizes all the moves we need for finding the lowest point with a given starting simplex. Once we find the lowest point with a given starting simplex, we should move to another set of starting simplex and start the entire process again. An example shown in Figure 4.18. Hence, the output profiles will be stored as an OBJ and VRML files in order to use it to extract further features.



.

Figure 4-18 Final result for the symmetry plane identification.

4.5 Tip of Nose, Bottom of Nose and Nose Bridge Identifications

Once the symmetry plane has been extracted and saved as an OBJ and VRML files, the profile is analyzed to identify a local extreme point that corresponds to the tip of nose, the nose bridge and the bottom of nose. The nose bridge is also used to extract the eye profile and the inner eyes corner points. In order to avoid false alarms, before starting the analysis process the curve is filtered to get a smooth symmetry plane passing through the discrete data.

Figure 4.19 shows a symmetry profile with the tip of nose, nose bridge and the bottom of nose points These are identified by computing their distances relative to the line joining the symmetry curve extremes as described below.

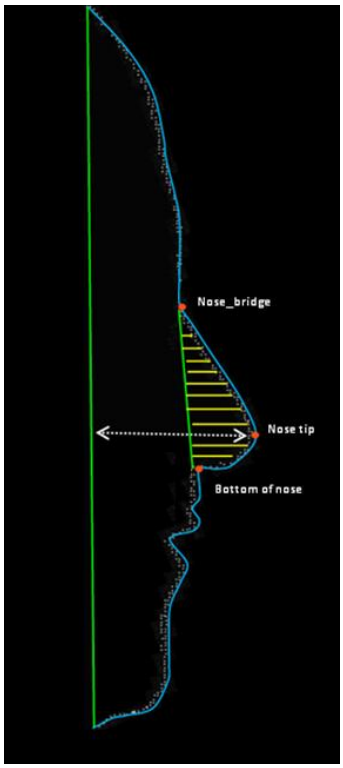


Figure 4-19 Feature points on the symmetry profile, the tip of nose, nose bridge and the bottom of nose.

The 3D location of the nose tip is a crucial step in the recognition of 3D face data. In order to find the tip of the nose, the two ends of the line profile have been joined by a line. To simplify the calculations of the accurate distances between the symmetry curve and the line (which connected between the two ends) we have rotated the line about one end point through an angle which brings the other end point to the same y value as first end point, as shown in Figure 4.20.

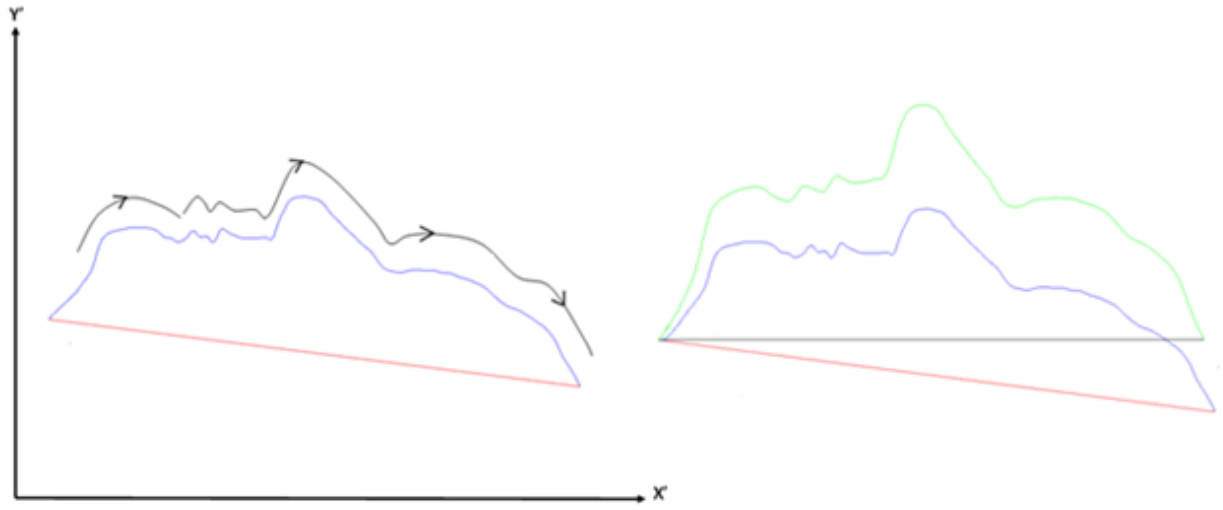


Figure 4-20 The symmetry profile connected with a line, which then moved to be perpendicular to the Y′ axis.

The distances between the line and the symmetry curve are then calculated. The perpendicular distance between any point *i* in the face curve and the point *p* in the line that connects the two ends is:

$$D = \sqrt{(X(i) - X_p)^2 + (Y(i) - Y_p)^2} \qquad (4.10)$$

where

$$X_p = \frac{(Y(i) - (\tan(\tan^{-1} b + \frac{\pi}{2})) \times X_p) - a}{b - \tan(\tan^{-1} b + \frac{\pi}{2})} \ , \qquad Y_p = b \times X_p + a$$

and

$$a = Y(1) - b\, X(1) \ , \qquad b = \frac{Y(i) - Y(1)}{X(i) - X(1)}$$

The nose tip is determined as the point on the facial profile with maximum perpendicular distance from a certain line (the horizontal line connected between the two extremes). So the point in the face curve with maximum distance is found. From the tip of nose, the nose bridge and the bottom of nose points on the symmetry profile are determined by searching all the points around the tip of nose in both directions to find the lowest point (decreasing) before the curve starts to increase again. In order to extract these features the face curve has been divided into three parts. The first and third parts are discarded because we know that the tip of nose is in the middle part. However, in order to ensure that the above algorithm produces correct results, the point positions (i.e. nose tip, nose bridge and bottom of nose points) on the symmetry curve have to be identified correctly. Thus, in case the data is not smooth enough another condition is included before selecting the point which represents the change in the curve behaviour (from decreasing to increasing). The number of points in the direction of increasing in the distance values should be large, and

this number varies depending on the symmetry curve length. Figure 4.21 shows the symmetry curve features allocation method, which was used to allocate the set of distinguishable points, including the tip of nose, nose bridge and the bottom of nose.

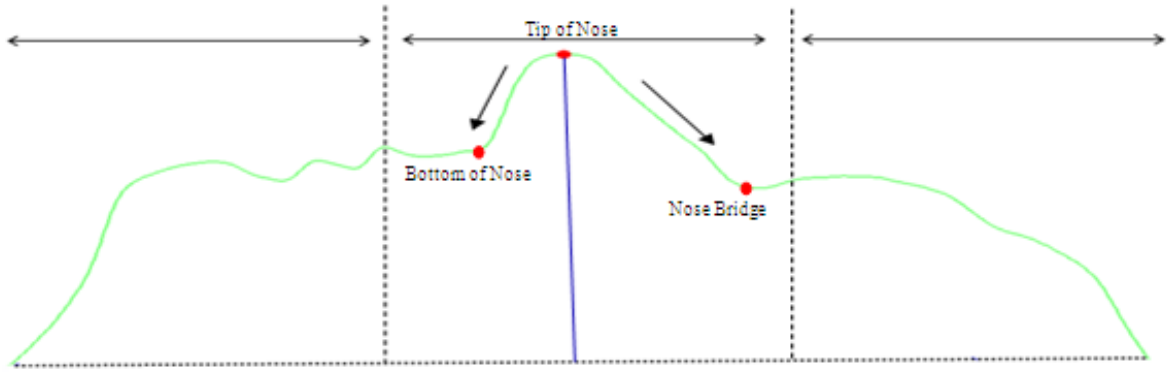
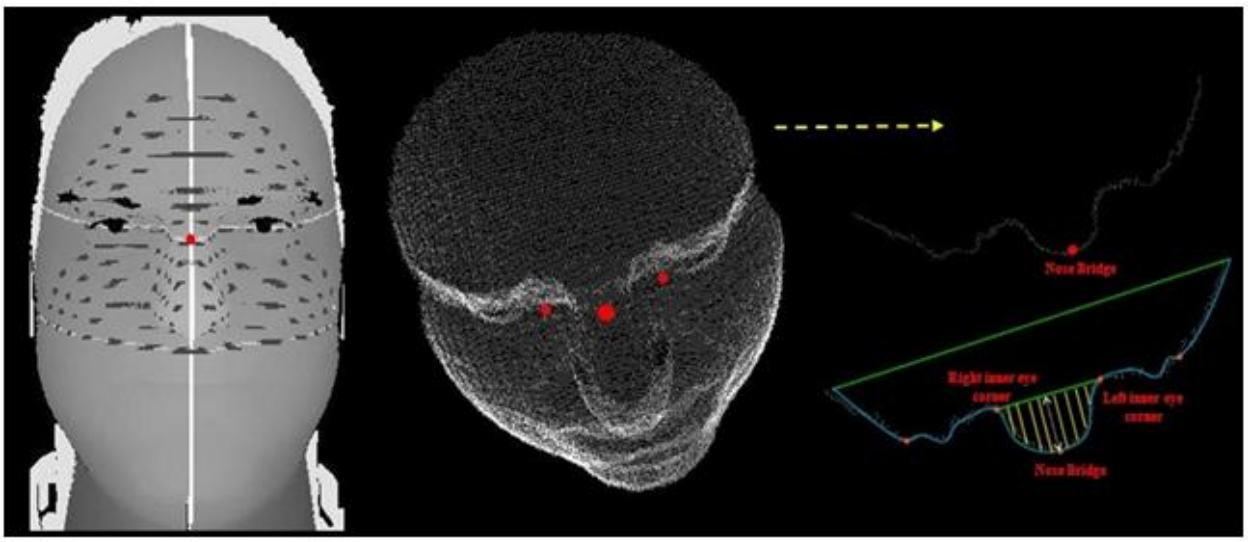


Figure 4-21 Symmetry curve features allocation method.

4.6 Eye profile and inner eye corners identification

As mentioned before, the symmetry plane plays the major role in the feature extraction process. After extracting the nose bridge point on the symmetry profile, the eye profile intersecting the nose bridge point is found. The origin of the P_Y principal axis is shifted to the nose bridge point and the new associated plane of intersection with the face data gives a new curve that represent the eyes profile. This contains new distinguishable features including the inner corner of the left eye and the inner corner of the right eye as shown in

Figure 4.22. The techniques used to identify symmetry plane features were used again. Firstly, points on the curve passing through the nose bridge point were identified and the eye curve was smoothed in order to avoid any false detection. Then the two ends are joined by a straight line. Using equation 4.10, the distances between the line and the eyes curve are calculated in order to extract further features. The face curve has been divided into three parts, with the focus on the second part containing the nose bridge point. From the nose bridge the two inner eyes corners are determined by searching all the points around the nose bridge in the both directions to find the lowest point (decreasing) before the curve behaviour changes, as shown in Figure 4.23. Moreover, the extracted eye profile was saved as an OBJ and VRML files in order to identify the inner eyes corners.



(a) (b)

Figure 4-22 a) The intersection point between the symmetry profile and the eyes profile through the nose bridge point. b) the extracted eyes curve which contains the nose bridge, the inner corner of the left eye and the inner corner of the right eye.

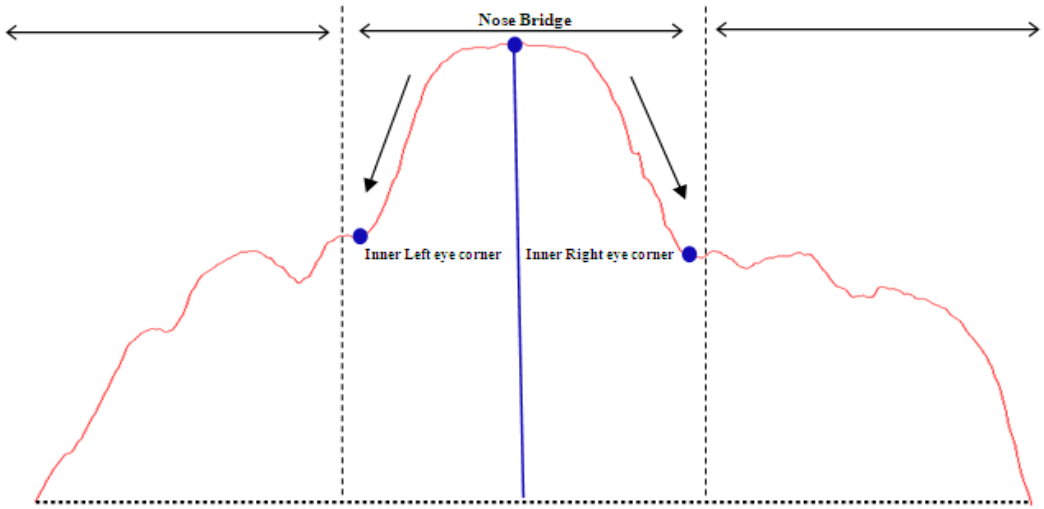


Figure 4-23 Eyes curve features allocation method.

4.7 Facial feature numerical representation

After the main feature points, (nose tip, bottom of nose, nose-bridge and two inner eye corners) have been extracted, it is necessary to find a numerical representation in order to use these features effectively with machine learning algorithms for classification purpose.

4.7.1 Geodesic distances

Firstly, the geodesic distance between the tip of nose and the nose bridge points, and the geodesic distance between the bottom of nose and the tip of nose points were calculated using the symmetry profile. Using the eyes curve, the geodesic distance between the right inner eye corner and the left eye corner was calculated.

The geodesic distance (GD) is the shortest distance between any two points on a surface measured along a path on the surface [192, 193]. For example, between any two points on a sphere, there is a unique great circle, and the two points separate the great circle into two arcs. The length of the shorter arc is the great circle distance between the points i.e. the shortest path from p to q suppose $P = p_1, p_2, \dots, p_n$ is a path between points p_i and p_n, where p_i and p_{i+1} are connected neighbours for $i \in \{1, 2, \dots, n-1\}$ and p_i belong to the domain for all i. The geodesic distance is defined as:

$$GD = \sum_{i=1}^{n-1} \sqrt{1 + (y_i - y_{i-1})^2} \qquad (4.11)$$

Where $\sqrt{1 + (y_i - y_{i-1})^2}$ represent the values for each pair of connected neighbours.

4.7.2 Surface Areas

In an attempt to find more features which can be accurately defined, the nose area has been calculated using the composite Simpson's rule. The interval [*a*,*b*] has been broken into a number of smaller subintervals, Simpson's rule is applied to each subinterval and the results are summed to produce an approximation for the integral over the entire interval. Suppose that the interval [*a*,*b*] is split up in *n* subintervals, with *n* an even number. Then, the composite Simpson's rule is given by:

$$\int_a^b f(x).dx \approx \frac{h}{3}\left[f(x_0) + 2\sum_{j=1}^{n/2-1} f(x_{2j}) + 4\sum_{j=1}^{n/2} f(x_{2j-1}) + f(x_n)\right] \qquad (4.12)$$

Figure 4.24 illustrates the method used to calculate the nose area. As shown in diagram (a) the whole area under the curve between the nose bridge and the bottom of the nose was calculated, then the trapezium area between the nose bridge and the bottom of nose shown blue in diagram (b) was calculated. Afterwards, the trapezium area between nose bridge and the bottom of nose was subtracted from the whole area under the curve between the nose bridge and the bottom of nose as shown in diagram (c). We have calculated the area between the two inner eye corners using the same method.

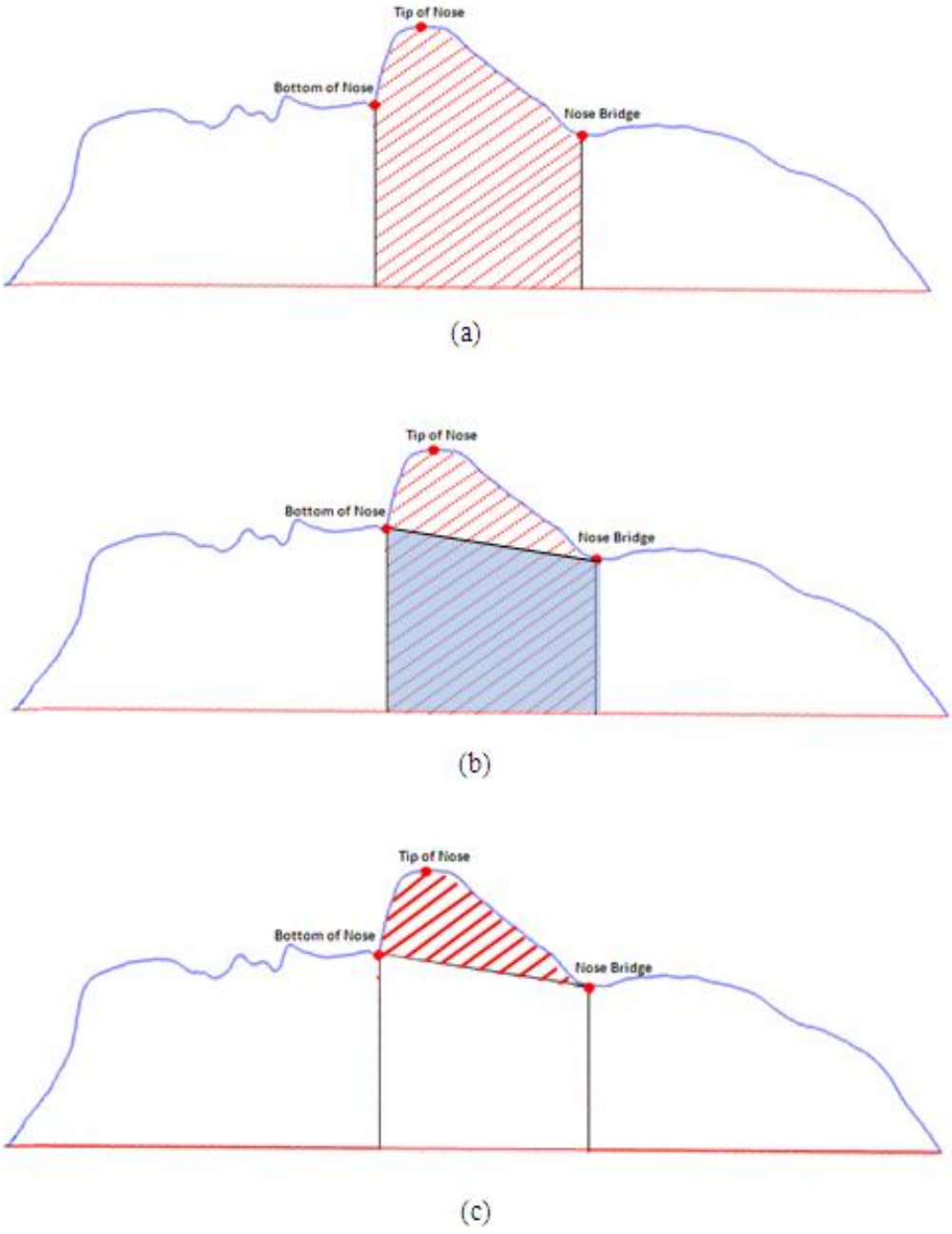


Figure 4-24 The nose area calculation process.

4.7.3 Discrete cosine transform (DCT) for six effective curves

In this section, the focus is on the use of the plane curves extracted from the 3D geometry of the face. Several facial curve based methods have been proposed in the past whose common goal is to extract set of curves that can be effectively used for 3D face recognition.

In the work of Li et al. [194], the authors extract the central profile curve and a depth contour curve from 2D depth images. Gökberk et al.[195] published recognition results based on sets of seven vertical profile curves. Instead of geodesic contour curves, Berretti et al.[196] use geodesic stripes and their spatial relationship to identify faces. All of the above-mentioned curve based methods require one or more reference points, such as the tip of nose, to start the extraction of facial curves from the 3D data.

In the current work and in order to obtain corresponding samples, a set of six profiles have been extracted. A profile is defined as a plane curve that passes through one or more of the points that were extracted previously, either vertically or horizontally and follows a path over the surface mesh.

Figure 4.25 indicates the positions of the curves that were extracted; a horizontal curve through the tip of nose, a horizontal curve through the bottom of nose point and a third horizontal curve through the eyes inner corner points and the nose bridge point, which is the eye profile. Vertical curves extracted include curves through the left and right eye

inner corners and a third curve through the nose bridge point, the tip of nose point and the bottom of nose, which is the symmetry curve.

The Discrete Cosine Transform (DCT) has been widely used in image data compression (jpeg) because it is highly effective at de-correlating the image data and packing the information into the smallest possible number of transform coefficients. The DCT closely approximates the information packing properties of the Karhunen-Loeve Transform (KLT), which is the optimum transform in a mean-square sense but much harder to calculate. The DCT is similar to the discrete Fourier transform; it transforms a signal or image from the spatial domain to the frequency domain.

The most common DCT definition of a 1-D sequence of length N is defined by the following equation and the one used is:

$$C(u) = \alpha(u) \sum_{x=0}^{N-1} f(x) \cos\left[\frac{\pi(2x+1)u}{2N}\right] \qquad (4.13)$$

For $u = 0,1,2,\dots,N-1$, and the corresponding inverse transformation is defined as

$$f(x) = \sum_{u=0}^{N-1} \alpha(u) C(u) \cos\left[\frac{\pi(2x+1)u}{2N}\right] \qquad (4.14)$$

For $x = 0,1,2,\dots,N-1$, in both equations $\alpha(u)$ is defined as

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}} & for \quad u = 0 \\ \sqrt{\frac{2}{N}} & for \quad u \neq 0 \end{cases}$$

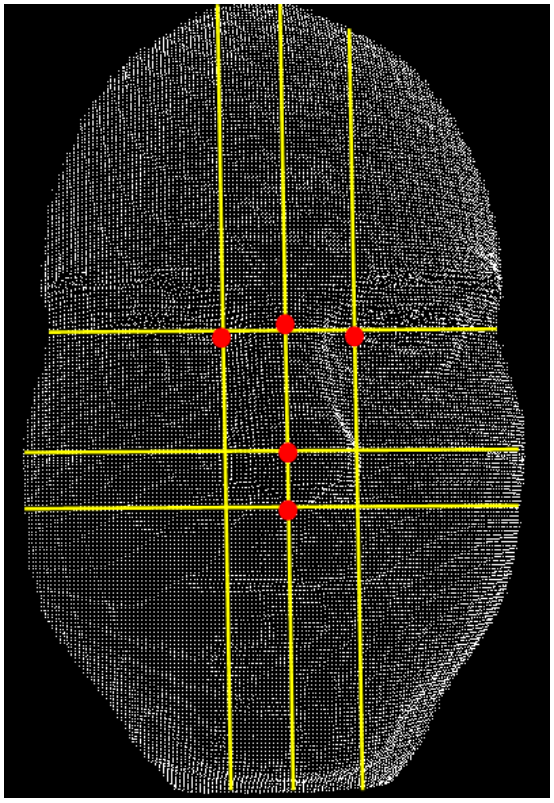


Figure 4-25 The extracted horizontal and vertical curves.

4.8 Summary

In this chapter, we have provided an automatic registration technique to process raw scanned 3D face data. Different from previous works using the whole face region for locating the symmetry plane of a face, we use the nose region to conduct our work because the nose region is relatively more rigid than other parts of a face and is almost expression invariant in anatomy sense. In this work we have produced the input data in the form of a 3D triangular facial mesh, the OBJ and VRML files have been generated from FRGC database in order to use them in further work and to make it available for other researcher to use. Robust pre-processing methods were used to reduce the amount of data to be processed, to remove noise spikes, to fill holes and to extract a facial region for feature extraction stage. In the new approach, the symmetry profile for the face was successfully determined automatically. This was undertaken by computing the intersection between the symmetry plane (found by an automatic search) and the facial mesh, resulting in a planer curve that accurately represents the symmetry profile.

Once the symmetry profile is successfully determined, a few feature points along the symmetry profile are computed. These features points are essential to the computation of other facial features, which can then be utilized to allocate the central region of the face and extract a set of effective profiles from that region. After the main feature points and curves around the nose area have been extracted, the output profiles saved as an OBJ and VRML files in order to use them for further work.

A numerical representation was calculated in order to use these features effectively with machine learning algorithms for recognition and classification purpose as explained in the following chapter (See Appendix C for an example).

CHAPTER FIVE

5. FACE RECOGNITION USING MACHINE LEARNING TECHNIQUES

A wide range of algorithms has been proposed for face recognition applications that utilize 2D or 3D facial information. Standard approaches that deal with the face as rigid object, such as the eigenfaces or standard ICP approached appeared to be ineffective in the presences of lighting, pose or facial expression variations.

In this chapter, face recognition is performed by selecting rigid regions that are less sensitive to variation of facial expression. In particular, we consider the part of the symmetry profile which goes through the nose region, and the part of the eyes profile as discussed in Chapter 4.

Three different machine-learning techniques: Cascade Correlation Neural Networks (CCNNs), Support Vector Machines (SVMs), and K-Nearest Neighbour (KNN) were used to make the final decision for the recognition and classification phases. The Jack-knife technique was employed to evaluate the performances of the learning system, where 80% of randomly selected samples were used for training and the remaining 20% for testing. The performance criteria used in the work were accuracy, sensitivity, and

specificity which are measured using the common biometric measures, namely the true positive ratio (TPR) and the false positive ratio (FPR) as explained in Section 2.4.

5.1 Experimental Dataset

As discussed in Chapter 3, a set of 3D images have been taken from the Face Recognition Grand Challenge Database (FRGC). For verification and recognition purposes 240 different sets of facial data with various densities have been processed and tested. The density of facial models varies from faces with 60,000 vertices up to 76,000 vertices.

The chosen 240 images contain several challenging problems. Most of these challenging issues of the entire database are represented well in the subset we used for research, which supports our decision to validate firstly our contributions on a smaller range. Moreover, running further experiments using more dataset items needs more processing and evaluation time due to the complexity for each case without addressing other challenges; as we started our experiments with 56 images, then by using the experience gained we increased the number of images to 120 from which the results motivated us to increase the amount to 240 images. The resulting system shows that we are in the right direction to construct a robust platform for a 3D facial recognition system capable to take the whole data set.

In this work, experiments were conducted on different numbers of people, starting with 120 images representing 12 people, including 10 images for each. Then the number was doubled to 240 images representing 24 different individuals, each individual represented by 10 images that cover a range of poses and expression in which there are variations of unwanted geometry including hair, neck, shoulders and clothes. As an example, a typical set of individual images is given in Figure 5.1.

Figure 5-1 A set of 3D facial images for one individual that includes different poses and expressions.

Numerical representations of features are used to construct the input parameters for the training and testing stages of the machine learning system. As the classification problem uses a multi classes input feature and a binary output for classifiers. The face features shown in Table 5-1 were calculated and normalised to be in the range between 0.1 and 0.9 for recognition using CCNN in order to find which of the features are the most significant. The output node has a numerical value of 0.9 if it is the correct class and 0.1 if it is not correct. The data was prepared in another specified format and scaled between -1 and 1 for recognition using SVM.

Table 5-1 The features extracted for 3D facial images.

<table>
<tr><th>Machine learning techniques</th><th colspan="4">Features used for recognition</th></tr>
<tr><td rowspan="6">CCNN, SVM, and KNN</td><td rowspan="6">125 Input Features</td><td rowspan="5">5 Input Features</td><td rowspan="3">3 Input Features</td><td>The Geodesic distance between Tip of nose and Nose Bridge.</td></tr>
<tr><td>The Geodesic distance between Tip of nose and Bottom of Nose.</td></tr>
<tr><td>The Area between Bottom of nose and Nose bridge.</td></tr>
<tr><td colspan="2">The Geodesic distance between Left inner eye corner and Right inner eye corner via Nose bridge.</td></tr>
<tr><td colspan="2">The Area between the left and the right inner eyes corners</td></tr>
<tr><td colspan="3">DCT for six horizontal and vertical curves.</td></tr>
</table>

To determine which of the machine learning algorithms is more suitable for the face recognition stage, the CCNN, SVM and KNN systems were optimised to find the best

parameters and topology, as described in the next subsections, before making the comparisons.

5.2 Experimental work using CCNN

The NN has proven to be a very useful tool for solving many real-life problems. However, efficient implementation of the NN usually requires long training sessions [148] which depend on the training vector and on the topology of the NN [148]. However, it was found that the CCNN topology provides good performance in terms of convergence time and optimum topology. In this network, the first layer has connecting weights to the input layer and each subsequent layer has weights connecting it to all previous layers including the input layer. As explained earlier, experiments were conducted with 120 images representing 12 people and then 240 images representing 24 different individuals, including 10 images for each individual.

In the CCNN experiments, the number of input nodes and the number of hidden nodes in each experiment were changed to find the best inputs and their related topologies. The numbers of input features used for 120 images were 3 and 5 while 3, 5 and 125 input features were used for 240 images. In addition, 30 CCNN configurations were created for each input feature by changing the number of hidden nodes from 1 to 10. Three experiments were carried out using the Jack-knife technique for each CCNN configuration and the average *TPR* and *FPR* were recorded.

The initial experiment was carried using 120 images with 2 input features (3 input features and 5 input features). It was found that a CCNN with 5 input nodes and 9 hidden nodes gave the best results for face prediction as it provided an accuracy of 0.975, a specificity of 0.987 and a sensitivity of 0.7 as shown in Table 5-2.

Table 5-2 Average performance indicators using five input features.

Hidden Node	TPR	FPR	TNR	FNR	Accuracy	Specificity	Sensitivity
1	0.1	0.039	0.9	0.961	0.925	0.961	0.1
2	0.5	0.022	0.5	0.978	0.958	0.978	0.5
3	0.5	0.022	0.5	0.978	0.958	0.978	0.5
4	0.4	0.026	0.6	0.974	0.95	0.974	0.4
5	0.2	0.035	0.8	0.965	0.933	0.965	0.2
6	0.6	0.017	0.4	0.983	0.967	0.983	0.6
7	0.5	0.022	0.5	0.978	0.958	0.978	0.5
8	0.6	0.017	0.4	0.983	0.967	0.983	0.6
9	0.7	0.013	0.3	0.987	0.975	0.987	0.7
10	0.8	0.009	0.2	0.991	0.983	0.991	0.8

These results motivated increasing the number of data samples and the number of features by conducting further learning experiments with the classifiers to ascertain the effectiveness for the extracted features. The new experiments were applied using 240 images and extra input features (125 features). As shown in Table 5-3, the CCNN with 4 input nodes gave the best results for face prediction and it provided an accuracy of 0.983, a sensitivity of 0.9 and a specificity of 0.983.

Table 5-3 Average performance indicators using five input features.

Hidden Node	TPR	FPR	TNR	FNR	Accuracy	Specificity	Sensitivity
1	0.9	0.1	0.1	0.9	0.9	0.9	0.9
2	0.8	0.2	0.2	0.8	0.8	0.8	0.8
3	0.7	0.05	0.3	0.95	0.914	0.95	0.7
4	0.9	0.017	0.1	0.983	0.971	0.983	0.9
5	0.8	0.033	0.2	0.967	0.943	0.967	0.8
6	0.6	0.067	0.4	0.933	0.886	0.933	0.6
7	0.7	0.05	0.3	0.95	0.914	0.95	0.7
8	0.4	0.1	0.6	0.9	0.829	0.9	0.4
9	0.7	0.05	0.3	0.95	0.914	0.95	0.7
10	0.5	0.083	0.5	0.917	0.857	0.917	0.5

However, it was concluded that using more input features such as 120 DCT coefficient combined with 5 input features did not improve the classification rate for CCNN over the use of 3 input features or 5 input features alone.

5.3 Experimental work using SVM

The SVM experiments were carried out using the LIBSVM program to optimise the performance of both the kernel and its parameters, which were determined empirically because there are no known guidelines to help choose them [173]. For the work reported here, the kernel types tested included Polynomial, Radial Basis Function (RBF) and Sigmoid and 5 input features were used. The results obtained are illustrated in Figure 5.2, which shows that the RBF kernel gave the best results. As explained in Section 2.4.2, the shape of the RBF kernel is controlled by the parameter γ (Gamma).

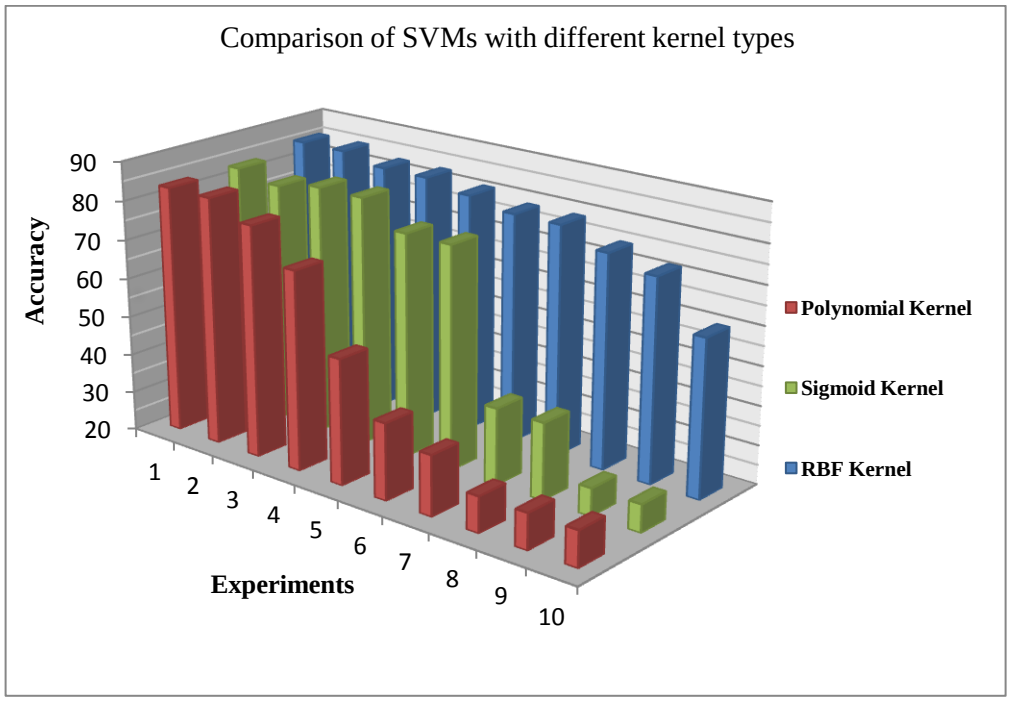


Figure 5-2 Comparisons of SVMs with different kernel types to recognize faces.

To complete the SVM optimisation it was necessary to determine the values of these parameters. This was done by training and testing the SVM over 100 iterations. During each iteration, γ was incremented from 0.1 to 2.5 in steps of 0.1 and C was incremented from 1 to 20 in steps of 1. Hence, 500 experiments were conducted to test these values. The numbers of input features were 3 and 5 for 120 images while 3, 5 and 125 input features were used for 240 images. For each of those configurations twenty experiments were carried out in order to find the best number of features and best parameters values.

The initial experiments using 120 images and 3 and 5 input features showed the best results were obtained when the γ value equals 0.2 for both of 3 features and 5 features. Figure 5.3 compares the accuracies of correct face predictions using 120 images for SVM

configurations for 3 and 5 input feature when the value of γ is fixed at 0.2 and the value of C is varied between 1 and 20. As shown in this figure, optimum performance is obtained when 3 input features were used with γ equal to 0.2 and C varied between 7 to12. Furthermore, twelve experiments for each individuals using 3 input features were carried out using the Jack-knife technique and the average TPR and FPR values were recorded. After these experiments, the performance indicators were found for every experiment, these indicators are TPR, FPR, FNR, TNR, accuracy, specificity and sensitivity.

The averages of these indicators were found for each input features and are shown in Table 5-4 after applying a "one against the rest" technique by decomposing a multiclass problem into a series of two-class problems. 144 experiments were carried out with 144 SVM configurations, resulting in 12 averages TPR and FPR values being produced. Using the RBF kernel with optimum C and γ values and optimum inputs configuration shows that the best values obtained were 0.900, 1 and 0.799 for accuracy, sensitivity and specificity, respectively.

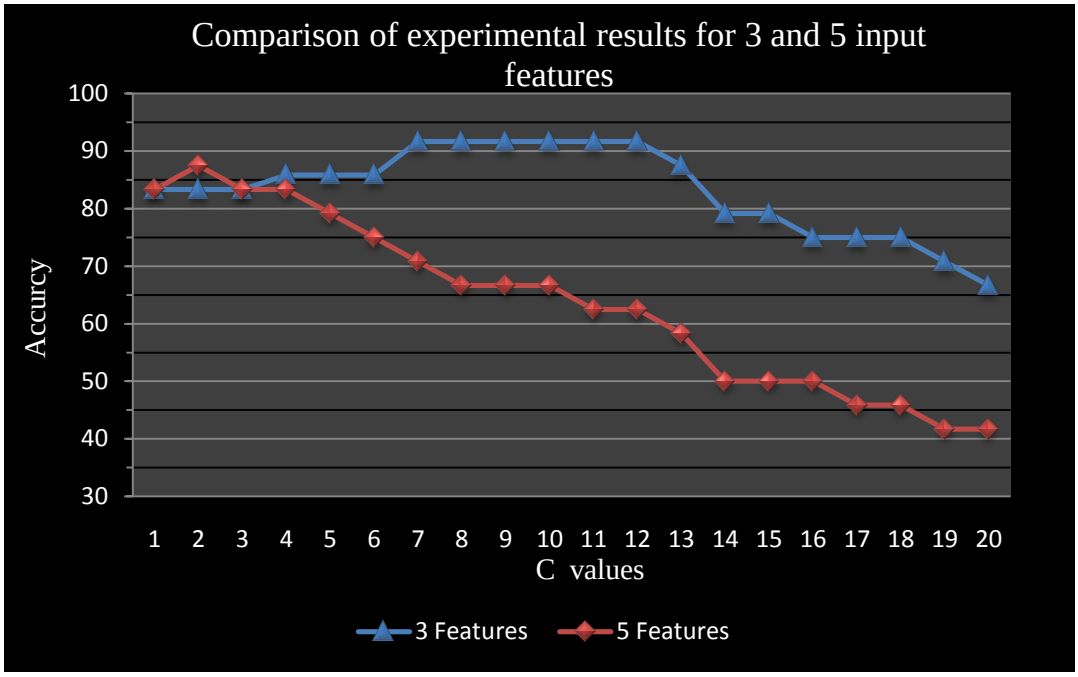


Figure 5-3 Percentage of correct face prediction by SVM, for 3 and 5 input features when C varied between 1 and 20 and γ set to 0.2.

Table 5-4 Average performance indicators for different input features.

Person Number	TPR	FPR	TNR	FNR	Accuracy	Specificity	Sensitivity
1	1	0.200758	0.799242	0	0.899620917	0.799241833	1
2	0.801074	0.405303	0.594697	0.198926	0.697885417	0.594696667	0.801074167
3	0.965	0.200758	0.799242	0.035	0.882120917	0.799241833	0.965
4	0.708333	0.413637	0.586363	0.291667	0.647348333	0.586363333	0.708333333
5	0.833333	0.398107	0.601893	0.166667	0.717613333	0.601893333	0.833333333
6	0.75	0.291667	0.708333	0.25	0.729166667	0.708333333	0.75
7	0.8	0.209471	0.790529	0.2	0.795264708	0.790529417	0.8
8	0.783333	0.114693	0.885307	0.216667	0.834320083	0.885306833	0.783333333
9	0.75	0.117763	0.882237	0.25	0.816118708	0.882237417	0.75
10	0.750002	0.20899	0.79101	0.249998	0.770505875	0.791010083	0.750001667
11	0.847223	0.178822	0.821178	0.152777	0.834200875	0.821178417	0.847223333
12	0.812275	0.196253	0.803747	0.187725	0.808010833	0.803746667	0.812275

Further experiments were conducted after doubling the number of images to 240 and increasing the number of input features to three. The accuracy of correct face prediction was calculated when the value of γ was fixed at 0.2, 0.5 and 0.008 for 3 features, 5 features and 125 features respectively and the value of C was varied between 1 and 20.

As shown in Figure 5.4 which compares the accuracies for the three kinds of input features, the optimum performance was obtained when 5 input features were used with γ equal to 0.5 and C varied between 1 and 3. Moreover, using the RBF kernel with these optimised parameters, twelve experiments for each individual using 5 input features were carried out and the average TPR and FPR values were calculated. Later on the average TPR and FPR values obtained for 288 SVM configurations were produced for each individual after applying a "one against the rest" technique, where the best values obtained were 0.882 for accuracy and 0.965, for sensitivity and 0.799 for specificity. The averages of indicators for each input features are also shown in Table 5-5.

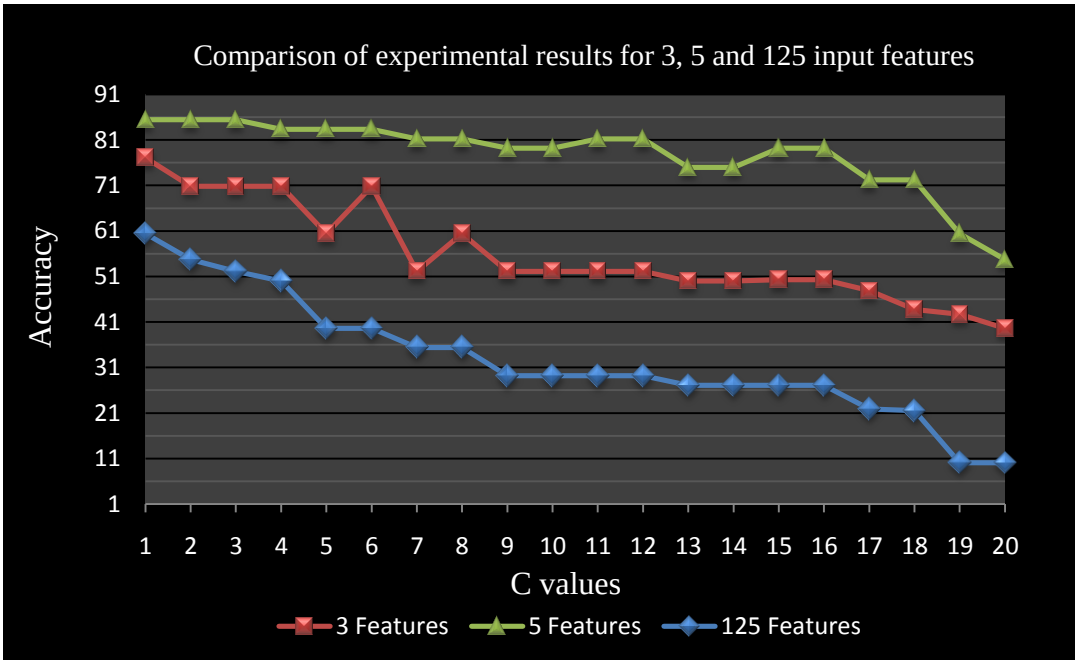


Figure 5.4 Percentage of correct face prediction by SVM, for 3 sets of input features when C varied between 1 and 20 and γ set to 0.2, 0.5 and 0.008 for 3 features, 5 features and 125 features respectively.

Table 5-5 Average ROC performance indicators for different input features.

Person Number	TPR	FPR	TNR	FNR	Accuracy	Specificity	Sensitivity
1	0.75	0.291667	0.708333	0.25	0.729167	0.708333333	0.75
2	0.801074	0.405303	0.594697	0.198926	0.697885417	0.594696667	0.801074167
3	0.965	0.200758	0.799242	0.035	0.882120917	0.799241833	0.965
4	0.708333	0.413637	0.586363	0.291667	0.647348333	0.586363333	0.708333333
5	0.833333	0.398107	0.601893	0.166667	0.717613333	0.601893333	0.833333333
6	0.75	0.291667	0.708333	0.25	0.729166667	0.708333333	0.75
7	0.8	0.209471	0.790529	0.2	0.795264708	0.790529417	0.8
8	0.783333	0.114693	0.885307	0.216667	0.834320083	0.885306833	0.783333333
9	0.75	0.117763	0.882237	0.25	0.816118708	0.882237417	0.75
10	0.750002	0.20899	0.79101	0.249998	0.770505875	0.791010083	0.750001667
11	0.847223	0.178822	0.821178	0.152777	0.834200875	0.821178417	0.847223333
12	0.812275	0.196253	0.803747	0.187725	0.808010833	0.803746667	0.812275
13	0.965	0.200758	0.799242	0.035	0.882121	0.799241833	0.965
14	0.708333	0.413637	0.586363	0.291667	0.647348	0.586363333	0.708333333
15	0.833333	0.398107	0.601893	0.166667	0.717613	0.601893333	0.833333333
16	0.75	0.291667	0.708333	0.25	0.729167	0.708333333	0.75
17	0.8	0.209471	0.790529	0.2	0.795265	0.790529417	0.8
18	0.783333	0.114693	0.885307	0.216667	0.83432	0.885306833	0.783333333
19	0.75	0.117763	0.882237	0.25	0.816119	0.882237417	0.75
20	0.595437	0.481852	0.518148	0.404563	0.556793	0.518148306	0.595437463
21	0.653519	0.383825	0.616175	0.346481	0.634847	0.616174685	0.653518574
22	0.709577	0.326253	0.673747	0.290423	0.691662	0.673746719	0.709577292
23	0.703971	0.33201	0.66799	0.296029	0.68598	0.667989516	0.70397142
24	0.754853	0.293802	0.706198	0.245147	0.730525	0.706197521	0.754853333

5.4 Experimental work using K-Nearest Neighbours

The KNN is a fast supervised machine learning algorithm which was used to classify the unlabeled testing set with a labeled training set. It was decided to conduct further experiments to improve the recognition performance as well as to ensure of the effectiveness of the extracted curves, since it was observed that their use with CCNN and SVM reduced the performance rates. As explained in Chapter 4 section 4.5.3, the DCT can be used to pack curve information into a much smaller number of transform coefficients. Once a set of six profiles have been identified, the DCT coefficients were calculated for each curve and the first 20 coefficients were used to make the matching. Checks were made to ensure that the 20 coefficients represent the original curve by using the inverse DCT function to restore the original curve.

In order to apply the KNN algorithm it is necessary to find the parameter K which represents the number of nearest neighbours and calculate the Euclidean distances between the query instance and the training samples in order to predict the testing data classes. Then the distances are sorted based on the minimum difference to determine the K^{th} nearest neighbours.

The first experiment was carried out on 120 images representing 12 individuals. Each individual has 10 images and each image has 6 curves, while 20 coefficients from each curve were used. A total of 120 coefficients were extracted to represent each vector beside 5 extracted landmark features were used. Using jack-knife technique 96 vectors and 24

vectors were randomly selected for training and testing sets, respectively. For the second experiment, 192 vectors and 48 vectors were randomly chosen for training and testing sets, correspondingly. The Euclidian distances between each vector of testing and each vector of training matrices were calculated. The experimental work showed that the best results were obtained when the *K* value equals 5 for 120 images and 3 for 240 images while the accuracy rate obtained were 0.79 for 120 images and 0.69 240 images respectively. Figure 5.5 illustrate the KNN results.

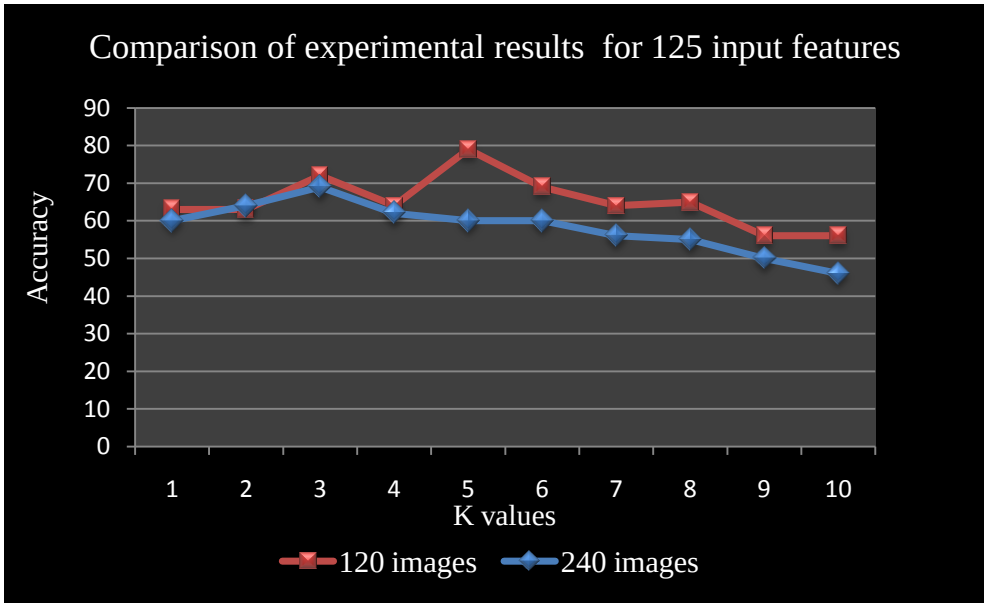


Figure 5.5 Percentage of correct face recognition by KNN, for 125 input features when K varied between 1 and 10.

5.5 Discussions and conclusions

The optimisation and learning experiments using CCNNs and SVMs were carried out as explained in the previous two subsections. For all the training and testing experiments, the jack-knife technique was applied to obtain the training and testing sets. At the end of these experiments the optimum configuration obtained for an SVM for 120 images with 3 input features was 0.2 and between 7 and 12 for γ and C, respectively. This configuration provided *TPR* and *FPR* rates of 1 and 0.2, respectively. On the other hand, the optimum topology for a CCNN using 120 images is 5 input nodes with 3 and 4 hidden nodes. This topology generated *TPR* and *FPR* values of 0.9822 and 0.0 respectively.

As indicated before for 240 images experiments, the best results were obtained using 5 input feature nodes, where the SVM uses the RBF kernel with γ and C parameters set to 0.5 and 3 respectively and the CCNN with 1 hidden layer and 5 hidden nodes. This configuration provides TPR and FPR rates of 0.965 and 0.2, respectively for SVM and generates TPR and FPR values of 0.9 and 0.004, respectively for CCNN.

It is clear from these results that the prediction performances for both CCNNs and SVMs have been improved and the FPR has been reduced. The best performance achieved using 5 input features were 0.89 for SVM and 0.981 for CCNN using 120 images. Moreover, values of 0.882 and 0.92 for successful prediction of faces for SVM and CCNN respectively were obtained using 240 images.

The use of 120 DCT coefficient combined with 5 input features didn't produce any improvement in the prediction performance for CCNN and SVM compared with the use of 3 input features or 5 input features alone. It was difficult for the SVM and CCNN classifiers to distinguish between different curves for different individuals because they are represented by values that are not separated enough for successful learning and output class separation.

It is shown by the validation experiments that the use of KNN technique to evaluate the extracted DCT coefficients gave better results than training them using SVM and CCNN techniques. The best results were obtained using jack-knife technique with *K* value equal to 5 for 120 images and 3 for 240 images while the accuracy rates obtained were 0.79 and 0.69 for 120 images and 240 images respectively.

The SVM is a binary classifier. The conventional way to extend it to a multiclass scenario is to decompose an *M*-class problem into a series of two-class problems, for which one-against-all is one of the most widely used implementations. The experiments showed that using the SVM was very fast compared to the CCNN which required longer training times.

As explained before, one of the more challenging problems in this work is how to compare different 3D face recognition techniques; there are few standardized 3D face databases which are used for benchmarking purposes. Thus, the size and type of 3D face datasets varies significantly across different publications. In addition, there are many

differences in the pre-processing methods and the experimental setups as well as in the metrics which are used to evaluate the performances of the face recognition techniques.

However, the algorithms developed in this work significantly outperforms some of the existing algorithms and have tackled many of crucial problems in this field such as the dataset size, the expressions variations, the presence of extraneous features, and the 3D face representations. We have conducted experiments using data from the FRGC database, the largest and most established data corpus for face recognition. Our results indicate that our algorithm exhibits high levels of accuracy and robustness and these remarks are justified by the contents of Table 5-4 which presents the characteristics and performance of the proposed system with respect to other existing systems.

In order to tackle the limitations explained in the literature we have used a subset of 240 images taken from the FRGC database which is considered as the most challenging dataset available for supporting research on 3D face recognition in regard to the expression and pose variations, and the presence of extraneous features [28]. The chosen 240 images contain several challenging problems. Most of these challenging issues of the entire database are represented well in the subset we used for research, which supports our decision to validate firstly our contributions on a smaller range. Moreover, running further experiments using more dataset items needs more processing and evaluation time due to the complexity for each case without addressing other challenges; as we started our experiments with 56 images, then by using the experience gained we increased the number of images to 120 from which the results motivated us to increase the amount to 240 images. The resulting system shows that we are in the right direction to construct a robust platform for a 3D facial recognition system capable to take the whole data set.

Table 5-6 Comparison between our proposed system and some of current recognition algorithms

Reference	Database	Image size	Data Representation	Algorithm		Reported Performance (Accuracy)			Handle size variation	Handle Expression	Model Type
Wang et al.[84] 2002	300	128 ×512	Feature vector point signature Gabor features	PCA+SVM		>90%			No	Yes	3D +2D
Tsalakanidou [112], 2003	80	100×80	Range image	PCA		99%			N/A	N/A	3D +2D
Chang et al. [111] 2003	Set of FRGC	480×640	Texture+ Range image	PCA		99% 3D + 2D, 93% 3D only			Yes	No	3D +2D
Moreno et al. [100] 2003	420	2.2K points	Curvature, line, region features	Euclidean nearest neighbor		78%			Yes	Some	3D
Lee et al.[116] 2004	70	320×320	depth map	Feature extraction, Nearest neighbor		94%			Yes	No	3D
Chang et al.[197] 2005	Set of FRGC	480 × 640	Point set	Multi-ICP		92%			Yes	No	3D+ 2D
Lee et al. [116] 2005	200	Various	Feature vector	SVM		96%			Yes	No	3D
Pan et al. [117]2005	Set of FRGC	480 × 640	Range image	PCA		95%			N/A	N/A	3D
Zhang et al. [71] 2005	32	N/A	Range images	mean curvature		96.9%			No	No	3D
Lu et al. [115],2005	196	240 × 320	Surface mesh	ICP, TPS		89%			N/A	N/A	3D
Passalis et al.[107] 2005	Set of FRGC	480 × 640	Surface mesh	Deformable model		90 %			N/A	Yes	3D
Husken et al.[198] 2005	Set of FRGC	480 × 640	Heir. graph	Graph match		93%			N/A	N/A	3D + 2D
Razdan et al.[103] 2007	421	N/A	triangle mesh	Spin Image		>93.12%			N/A	Yes	3D
Elyan and Ugail [104]2009	144	N/A	triangle mesh	Coon's surface patch		86.9%			No	Yes	3D
Our proposed System	240 FRGC v2.0	480 × 640	Range images, Point set, Triangle mesh (.Obj +.Wrl)	Feature Extraction (PCA, Downhill Simplex method, Reflection Method)		**Accuracy**	**Sensitivity**	**Specificity**	Yes	Yes	3D
				CCNN	120	98%	98.2%	100%			
					240	92%	90 %	99.6%			
				SVM	120	89.96%	79.9%	100%			
					240	88.2%	80 %	96.5%			
				KNN	120	79 %	N/A	N/A			
					240	69 %	N/A	N/A			

CHAPTER SIX

6. FINAL CONCLUSIONS AND FUTURE WORK

6.1 Overview

The most commonly used approach in face recognition applications is based on 2D intensity images. These applications are widely used in commercial and academic work. The relatively low price to set up such a system, and the availability of a wide range of algorithms (Chapter 2 section 2.2) made it very popular. These systems achieved acceptable recognition rates in the absence of facial pose variations, expression variations and within controlled environments.

3D facial models have been used extensively in the past two decades for face recognition purposes. The use of those models in biometric applications is due to the inherent problems of the classical 2D image based face recognition systems, which mainly result from pose, light, and facial expression variations. Because of these limitations, the trend has shifted toward utilizing 3D images for face recognition applications which gave better recognition rates than the 2D images approach, especially in the presence of facial pose and expression variations, as discussed in the literature review.

In spite of the improvements in terms of recognition rates and accuracy, 3D image based face recognition are still faced with several challenges, such as the localisation of facial feature points, the quality of the 3D images, and the availability of public benchmark 3D face databases. In addition, it is not easy to compare or reproduce results of other research as many results are not reported using the same data. However, when there is a common database, different pre-processing operations may still be used for different methods, which makes direct comparisons of the methods difficult.

In this research study, the focus is on the use of 3D FRGC database in a recognition system using the machine learning techniques (CCNN, SVM, and KNN) and the major components of the research discussed in this thesis are outlined as follows:

- Chapter 2 provides a detailed literature review for 2D and 3D face recognition techniques. In addition, the FRGC database is also presented in detail with clear outline of the machine learning algorithms.
- Chapter 3 presents the 2.5D face recognition system based on range data. The interpretation of 2.5D data and the extraction of the facial region are described in this chapter. Furthermore, the pre-processing of the face data and the feature extraction are explained.
- Chapter 4 describes the pre processing and facial features extraction for a 3D facial mesh. It also gives a numerical representation for these features in order to use them for recognition purpose. These features include a set of

profiles and distinguishable points extracted from the central region of the face.

- Chapter 5 provides a practical implementation and an evaluation of the proposed 3D face recognition and classification systems using machine learning algorithms. It compares the performances of some learning algorithms: Cascade-Correlation Neural Networks (CCNNs), Support Vector Machines (SVMs) and the K-Nearest Neighbours algorithm (KNN).

6.2 Detailed Conclusions

- In Chapter 2 the current literature regarding 2D and 3d face recognition technology is reviewed and a number of methods worthy of further investigation identified. Also, a general view about Face Recognition Grand Challenge database which was used in this research is presented. Furthermore, some machine learning techniques such as CCNN, SVM and KNN were used for classification purposes.

- In the work discussed in Chapter 3, 3D face recognition with range data from FRGC Ver.2.0 data set was carried out using the CCNN, SVM and KNN machine learning techniques, which had not been done before. Pre-processing operations were needed in order to extract effective features. At this stage the 2.5D data was interpreted from ABS files then the facial region was extracted. Standardization of the face area, the removal of spikes and the filling of holes in the data were essential steps in order to extract the features. Moreover, it was important to choose anatomical

distinctive landmark features that contain both local feature information and the overall size of the face. Applying the machine recognition algorithm was the last stage to evaluate the extracted features. Cascade correlation networks were used to make the final decision; it has a number of attractive characteristics including a very fast training time. Using this approach has been found appropriate yielding promising results for 2.5D face recognition which has encouraged expanding the work to include more dataset and extract new features along with using further machine learning techniques.

- Improve the effectiveness of 2.5 face recognition algorithm, by presenting a novel method for automatic processing and developing feature extraction for 3D facial data. This research direction has continued using the experience gained in 2.5D face recognition investigations to provide an automatic facial feature extraction technique as discussed in Chapter 4. Various pre-processing methods were applied to produce numerous 3D mesh representations from ABS raw data, namely object files, VRML files and point clouds representation. These files have been generated from FRGC Ver.2.0 database in order to use them in further work and to make it available for other researcher to use. Hence the database is available only in .ABS format. The resulting system was able to accomplish the first goal to automatically determine a symmetry profile for the face. This was undertaken by computing the intersection between the symmetry plane (found by an automatic search) and the facial mesh, resulting in a planar curve that accurately represents the symmetry profile.

- In order to address all the problems in implementing a reflection measure, an effective reflection measure process has been adopted, which has proved to give reliable results. The sum of the individual absolute differences between two distances on either side of the test plane, for face points on the same perpendicular line were used to calculate the reflection measure. To handle the possibilities of having a different densities of points and holes in different regions of face data, a set of lines perpendicular to the test plane, and spaced equally apart in the test plane are defined. For these lines which intersect the face data on both sides of the test plane a contribution to the reflection measures are computed. The problem of the irregular boundary of the face data is tackled by applying the symmetry measure only to the part of the face in front of a plane through the data centroid orthogonal to the test plane. This removes the irregular data boundary towards the back of the head.

- Facial feature extraction to a high level of accuracy, with much improved robustness to symmetry profile identification presented in chapter 4, was achieved based on using the nose region to extract a set of effective facial features from what is a relatively more rigid part of the human face than the symmetry plane analysis of the whole face region. Once the symmetry profile is successfully determined, a few feature points along the symmetry profile are computed automatically. These features points are the tip of nose, the nose bridge, the bottom of nose, and the two inner eyes corners. Extracting these features is essential to the computation of other

facial features, which are utilized to allocate the central region of the face and extract a set of distinctive profiles from that region.

- Finally, in an attempt to utilize the extracted features effectively with machine learning algorithms for recognition purpose, a numerical representation for them was computed. The geodesic distance between the tip of nose and the nose bridge points, as well as the geodesic distance between the bottom of nose and the tip of nose points were calculated using the symmetry profile. Along with using the eyes curve, the geodesic distance between the right inner eye corner and the left eye corner was determined. Moreover, the nose area and the area between the two inner eye corners have been calculated using the composite Simpson's rule and the DCT for six facial profiles have been computed.
- Evaluate the extracted facial features using several machine-learning techniques in order to produce a fully 3D face recognition system. In Chapter 5, three different machine learning algorithms (CCNN, LIBSVM and KNN) and several sets of input features (3 features, 5 features and 125 features) were tested and compared for faces prediction. The extracted features were arranged in appropriate numerical formats so that they could be processed by machine learning algorithms. It was concluded that using five input features for recognition purposes were the most effective means of recognizing faces.

- For the first time, CCNNs and SVMs were used to recognize and classify a 3D dataset taking from FRGC Ver.2.0 database. In order to evaluate the performances of the learning system, the Jack-knife technique was employed with the use of 80% randomly selected samples for training and the remaining 20% for testing.

- After conducting extensive experiments using different number of images, it was found that CCNNs provided more accurate results for face recognition than using the SVM, where the optimum configuration obtained for CCNN using 5 input features achieved 0.981, 0.9822 and 0.01 for accuracy, TPR and FPR respectively. In comparison, the optimum topology for SVM generated values of 0.89, 1 and 0.2 for accuracy, TPR and FPR respectively. For 240 image experiments, the best results obtained using 5 input feature nodes using the SVM were 0.882, 0.965 and 0.2 for accuracy, TPR and FPR rates respectively and values of 0.92, 0.9 and 0.004 for accuracy, TPR and FPR respectively for CCNN.

- It was decided to conduct further experiments using the KNN algorithm to try to improve the recognition performance. As well as to ensure of the efficiency of the extracted curves, especially that used them within CCNN and SVM has reduced the accuracy. However, the use of KNN technique to evaluate the extracted DCT coefficients gave better results than trained them using SVM and CCNN techniques. The best results obtained using the jack-knife technique with *K* value equal to 5 for 120 images and 3 for 240 images achieved accuracy rates of 0.79 and 0.69 for 120 images and 240 images, respectively.

6.3 Original Contributions

The main original contributions presented in this thesis can be summarised as follows:

- A fully automatic recognition method with high accuracy and robustness to facial expressions for nearly frontal views using 3D face data is presented. Unlike previous work using the whole face region for feature extractions, we use the nose region, which is automatically located and analyzed to conduct our work because the nose region is relatively less affected by facial expression compare to other facial regions in the anatomical sense.

- A new method of determining the symmetry plane of 3D face data has been developed by defining a numerical measure for the degree of reflection symmetry for face data around a given plane. The reflection measure adopted is the sum of the individual differences between two distances on either side of the test plane, on the same perpendicular line. This technique has proved to give reliable and accurate results.

- New feature extraction algorithms extract discriminative feature points and a set of effective profile curves over the nose region with numerical representations in order to use these features effectively with machine learning algorithms. This method overcomes the limitations described in the literature, and can accept data in several 3D formats (depth image, point cloud, object files, and VRML files).

- Machine learning algorithms are used for the first time with FRGC dataset. The system design implements different machine learning algorithms, which have been optimised to determine which of the machine learning algorithms is more suitable for face recognition. The results are presented in the form of ROC curves which are used to analyze more than two classes of prediction problems.
- To verify our claims, the FRGC v2.0 benchmark 3D face dataset was utilized for evaluating the performance of our method in registering 3D facial datasets. Results achieved using our automatic recognition method are promising and demonstrate that our system is not only effective but also efficient.

6.4 Future Work

Future work to improve the outcome of the current work should include more accurate and efficient techniques for improving the facial extraction and the recognition techniques in the areas indicated in the following suggestions.

In general, comparing different 3D face recognition techniques or reproduce results is very challenging for a number of reasons. Firstly, there are very few standardized 3D face databases which are used for benchmarking purposes. Thus, the size and type of 3D face datasets varies significantly across different publications. Secondly, there are differences in

the experimental setups and in the pre-processing operations, as well as differences in the metrics which are used to evaluate the performances of face recognition techniques. Therefore, in order to make comparisons between our proposed system and those of other researchers who used 3D FRGC *Ver.2.0* database the current work needs to increase the number of faces used to the whole 3D FRGC *Ver.2.0* dataset.

The proposed facial feature extraction is designed to identify certain facial features by identifying the symmetry profile to begin with then determine the some points along it, namely the tip of nose, the nose bridge, and the bottom of nose. However, this algorithm is designed to work in frontal facial scans. More work in this area needs to be done to improve the algorithm to work in frontal left and frontal right facial images.

The downhill simplex search method which is used to determine the facial symmetry plane is efficient and easy to use, as well as being very generally applicable. However, this technique is slower than other less general methods, and is not very efficient in terms of the number of function evaluations that it requires. Faster alternatives could be investigated.

The algorithm for extracting facial features from scanned images assumes that the nose tip lies on the symmetry profile. In extreme cases where such assumption might not hold, the results may not be valid. Hence, further investigation is needed to address such cases.

The automatic feature extraction approach proposed in this study depends on the proper identification of the symmetry plane. Incorrect identification of the symmetry plane

would essentially lead to incorrect identification of the other facial features. This is considered as a drawback that could be overcome by investigating the possibility of identifying other facial feature points rather than just relying on the symmetry profile as an initial step to recover other important features such as the tip of nose. In addition an automatic (rather than a manual) verification that the search has yielded a symmetry plane could be achieved by checking that the plane closed curve associated with the intersection of the new Z principal plane with the face data is in fact symmetric about the reflection plane.

Another possible improvement to this algorithm is to consider cases where pose angles are more than ± 15^{o}, the pose correction may be needed to bring the face to the correct frontal orientation. Moreover, extracting distinctive features and combining various kinds of them should be useful and could give better recognition rate than any single matching methods.

Although using a machine learning techniques with Jack-knife (20, 80) shows a good recognition rate, there is still a great deal of development that may be done in this area. One brief experiment has already demonstrated that using 40% of data for testing and 60% for training can produce good TP, FP ratios and lower error rates than using (20, 80) for SVM using 10 images for 24 people. This suggests that altering the testing and training ratios may make the system flexible and more effective.

Future work considering various semantic factors, namely, age, gender and ethnicity, needs to be investigated. This might include considering other facial metric features to be integrated in the comparisons algorithm to improve the recognition rate. These improvements are mainly dependent on the improvement of the automatic facial feature extraction, where more feature points might be automatically identified, which would result in deriving more facial metric measurements.

Another possible improvement is to consider the utilization of the texture data associated with the 3D images. In other words combining both 3D and 2D facial data, which will allow the system to make use of the wide range of 2D image based algorithm along with 3D algorithm to improve the recognition rate.

References

1. Akarun, L., B. Gökberk, A.A. Salah. *3D Face Recognition for Biometric Applications*. in *European Signal Processing Conference*. 2005. Antalya.
2. W. Zhao, R.C., P. J. Phillips, A. Rosenfeld, *Face recognition: a literature survey* ACM Computing Surveys, 2003. **35**(4): p. 399–458.
3. M. Turk , A.P., *Eigenfaces for recognition.* Journal of Cognitive Neuroscience 1991. **3**(1): p. 71-86.
4. P. N. Belhumeur, J.P.H.a.D.J.K., *Eigenfaces vs. fisherfaces: Recognition using class specific linear projection.* IEEE Trans. Pattern Analysis and Machine Intelligence, 1997. **19**(7): p. 711–720
5. L. Wiskott, J.F., N. Kruger, and C. von der Malsburg, *Face recognition by elastic bunch graph matching.* IEEE Trans. Pattern Analysis and Machine Intelligence, 1997. **19**(7): p. 775–779.
6. Yang, M.H. *Kernel eigenfaces vs. kernel fisherfaces: Face recognition using kernel methods*. in *IEEE International Conference on Automatic Face and Gesture Recognition*. 2002.
7. P. Phillips, P.G., R. Micheals, D. Blackburn, E. Tabassi and M. Bone, *Face Recognition Vendor Test 2002*, in *NIST Technical Report*. 2003, NIST.
8. S. Rizvi, P.P.a.H.M., *The FERET Verification Testing Protocol for Face*. 1998, NIST Technical Report.
9. P. Phillips, H.M., S. Rizvi, and P. Rauss, *The FERET Evaluation Methodology for Face-Recognition Algorithms.* IEEE Trans. Pattern Analysis and Machine Intelligence, 2000. **22**: p. 1090-1103.
10. A. S. Georghiades, P.N.B.a.D.J.K., *From few to many: Illumination cone models for face recognition under variable lighting and pose.* IEEE Trans. Pattern Analysis and Machine Intelligence, 2001. **23**(6): p. 643–660
11. Chellappa, W.Z.a.R. *Illumination-insensitive face recognition using symmetric shape-from-shading*. in *IEEE International Conference on Computer Vision*. 2000.
12. Blanz, V., Romdhani, S., and Vetter, T. *Face Identification across Different Poses and Illuminations with a 3D Morphable Model*. in *International Conference on Automatic Face and Gesture Recognition*. 2002.
13. Rusinkiewics, O.H.-H.a.S. *Stripe boundary codes for real-time structured light range scanning of moving objects*. in *IEEE International Conference on Computer Vision* 2001.
14. S. Rusinkiewicz, O.H.-H.a.M.L., *Real-time 3D model acquisition.* Computer Graphics Proceedings SIGGRAPH, 2002: p. 438–446.
15. *Face Recognition Vendor Test (FRVT).*
16. Jain, S.L.a.A., *Handbook of Face Recognition*. 2005: Springer.
17. Oh, Y.R.a.S., *Automatic extraction of eye and mouth fields from a face image using eigenfeatures and multiplayer perceptrons.* Pattern Recognition, 2001. **34**(12): p. 2459–2466.

18. Cootes, D.C.a.T. *Facial feature detection using adaboost with shape constraints*. in *14th BMVC*. 2003. Norwich, UK.
19. L. Wiskott, J.M.F., and N. Kruger, *Face recognition by elastic bunch graph matching.* IEEE Trans. Pattern Analysis and Machine Intelligence, 1997. **19**(7): p. 775–779
20. K. Toyama, R.F., J. Gemmell, and V. Kruger. *Hierarchical wavelet networks for facial feature localization*. in *IEEE International Conference on Automatic Face and Gesture Recognition*. 2002. Washington D.C.
21. T. Cootes, G.E., and C. Taylor, *Active appearance models.* IEEE Trans. Pattern Analysis and Machine Intelligence, 2001. **23**(6): p. 681–685.
22. J. Xiao, S.B., I. Matthews, and T. Kanade. *Real-time combined 2D+3D active appearance models*. in *IEEE CVPR*. 2004.
23. Sim, T., Baker, S., and Bsat, M., *The CMU pose, illumination, and expression database.* IEEE Trans. Pattern Analysis and Machine Intelligence, 2003. **25**(12): p. 1615-1618.
24. Martinez, A.M., *Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class.* IEEE Trans. Pattern Analysis and Machine Intelligence, 2002. **24**(6): p. 748–763.
25. *3D_RMA Database, http://www.sic.rma.ac.be/~beumier/DB/3d_rma.html*.
26. *The UOY 3D Face Database,* http://www-users.cs.york.ac.uk/~nep/research/3Dface/tomh/3DFaceDatabase.html
27. A. B. Moreno, Á.S., José Fco. Vélez y Fco. Javier Díaz,. *Face recognition using 3D surface-extracted descriptors*. in *The Irish Machine Vision & Image processing Conference (IMVIP'03)*. 2003.
28. Phillips, P.J. *FRGC and ICE Workshop–NIST*. in *NRECA Conference Facility* 2006. Arlington, Virginia
29. Griffin, C.B.a.P., *Comparing and combining depth and texture cues for face recognition.* Image and Vision Computing, 2005. **23**: p. 339-352.
30. T. Nagamine, T.U., and I. Masuda, *3D facial image analysis for human identification.* International Conference on Pattern Recognition, 1992: p. 324 - 327.
31. Xiaoguang Lu, a.A.K.J. *Automatic Feature Extraction for Multiview 3D Face Recognition*. in *International Conference on Automatic Face and Gesture Recognition*. 2006. Southsampton, UK.
32. Phillips, P.J. P.G., R. J. Micheals, D. M. Blackburn, E. Tabassi and M. Bone, 4 , *FRVT 2002: Overveiw and summary*. 2002.
33. G. Medioni and R. Waupotitsch. *Face Recognition and modeling in 3D* in *IEEE International WorkShop on Analysis and Modeling of faces and Gesture Recognition*. 2003. Nice, France.
34. Vetter, V.B.a.T. *Morphable model for the synthesis of 3D faces*. in *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*. 1999.
35. T.F.Cootes, G.J.E., and C.J. Taylor. *Active Apperance Models*. in *European Conferance on Computer Vision* 1998. .
36. G. J. Edwards, T.F.C.a.C.J.T. *Face Recognition using active appearance models*. in *European Conference on Computer Vision* 1998.

37. Romdhani, V.B.a.S. *Face Identification across Different Poses and Illuminations with a 3D Morphable Model*. in *Fifth IEEE International Conference on Automatic Face and Gesture Recognition*. 2002: IEEE Computer Society.
38. Friesen, P.E.a.W.V., *Manual for the Facial Action Coding System.* Consulting Psychologists Press, 1977.
39. A. Lanitis, C.J.T.a.T.F.C., *Toward automatic simulation of aging effects on face images.* IEEE Transactions on pattern Analysis and machine Intelligence, 2002. **24**: p. 442-455.
40. P. J. Phillips, P.J.F., T. Scruggs, K. W. Bowyer, C. Jin, K. Hoffman, J. Marques, M. Jaesik, and W. Worek. *Overveiw of the face recognition grand challenge*. in *IEEE Computer Society Conferance on Computer vision and Pattern Recognition CVPR*. 2005.
41. Pentland, M.A.T.a.A.P. *Face recognition using eigenfaces*. in *Proceedings of the Conferences on Computer Vision and Pattern Recognition*. 1991.
42. A. X. Guan, H.H.S., and Z. Markowitz. *Local ICA for the most wanted face recognition*. 2000. Orlando,FL, USA.
43. A. Hyvarinen, J.K., and E. Oja,, *Independant Component Analysis*. 2001: Wiley Interscience.
44. P. N. Belhumer, J.P.H., and D. J. Kriegman, *Eigenfaces vs. Fisherfaces: recognition using class specific linear projection.* IEEE Transactions on patten Analysis and Machine Intelligence, 1997. **19**: p. 711 - 720.
45. A. Pentland, B.M., and T. Starner. *View-based and modular eigenspaces for face recognition*. in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 1994. Proceedings CVPR 94.
46. Moghaddam, B., *Principal manifolds and probabilistic subspaces for visual recognition.* IEEE Transactions on patten Analysis and Machine Intelligence, 2002. **24**: p. 780 - 788.
47. M. S. Bartlett, J.R.M., and T. J. Sejnowski, *Face recognition by independent component analysis.* IEEE Transactions on Neural Networks, 2002. **13**: p. 1450 - 1464.
48. Hyvarinen, A., *Fast and robust fixed-point algorithms for Independant Component Analysis.* iEEE Transactions on Neural Networks, 1999. **10**(3): p. 626-634.
49. Y. Jian, D.Z., A. F. Frangi, and A. J. -y. Y. Jing-yu Yang,, *Two-dimensional PCA: a new approach to appearance-based face representation and recognition.* IEEE Transactions on pattern Analysis and machine Intelligence, 2004. **26**: p. 131-137.
50. H. Kong, L.W., E. K. Teoh, J. G. A. W. J. G. Wang, and R. A. V. R. Venkateswarlu. *A framework of 2D Fisher discriminant analysis: application to face recognition with small number of training samples*. in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2005.
51. S. Bernhard, S., Alexander, and M. Klaus-Robert, *Nonlinear component analysis as a kernal eigenvaluse problem.* Neural Computation, 1998. **10**: p. 1299-1319.
52. W. Zhao, R.C. *SFS Based Veiw Synthesis for Robust Face Recognition*. in *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition* 2000: IEE Computer Society

53. Kanade, T., *Picture Processing by computer complex and recognition of human faces.* Pattern Recognition Letters, 1987. **5**: p. 183 - 187.
54. Poggio, R.B.a.T., *Face Recognition through geometrical features.* European Conference on Computer Vision, 1992. **588**: p. 792 - 800.
55. Poggio, R.B.a.T., *Face Recognition: Features versus templates.* IEEE Transactions on pattern Analysis and machine Intelligence, 1993. **15**: p. 1024 - 1052.
56. V. Blanz, S.R.a.T.V. *Face identification across different poses and illumination with a 3D morphable model*. in *IEEE International Conference on Automatic Face and Gesture Recognition*. 2002.
57. C.J.Taylor, T.F.C.a., *Statistical models of appearance for computer vision* 2004, Imaging Science and Biomedical Engineering, University of Manchester.
58. Atick, P.S.P.a.J.J., *Local features analysis: a general statistical theory for object representation.* Network: Computation in Neural Systems, 1996. **7**: p. 477 - 500.
59. B. Heisele, P.H., J. Wu, and T. Poggio, *Face recognition: component-based versus global approaches.* Computer Vision and Image Understanding, 2002. **91**: p. 6 - 21.
60. Harter, F.S.S.a.A.C. *Parameterisation of a stochastic model for human face identification*. in *Proceedings of the second IEEE Workshop* 1994.
61. Tsuhan, S.L.a.C. *A GMM parts based face representation for improved verification through relevance adaptation*. in *In proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 2004.
62. Identix, <*http://www.11id.com/*>.
63. C. Systems, <http://www.cognitec-systems.de>
64. Phillips, P.J. P.G., R. J. Micheals, D. M. Blackburn, E. Tabassi and M. Bone, 4 , *FRVT 2002: Overveiw and summary*. *http://www.frvt.org/FRVT2002/documents.htm*>. 2002.
65. R. Cross, J.S., and J. Cohn, *Quo vadis face recognition.* Third Workshop on Empirical Evaluation Methods in Computer Vision, 2001.
66. X. Chenghua, W.Y., T. Tieniu, and A. L. Q. Long Quan. *Depth vs. intensity: which is more important for face recognition*. in *17th International Conference on Pattern Recognition*. 2004.
67. K. W. Bowyer, K.C., and P. Flynn, *A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition.* Computer Vision and Image Understanding, 2006. **101**: p. 1-15.
68. K. W. Bowyer, K.C., and P. Flynn. *A survey of approaches to three-dimensional face recognition*. in *17th International Conference on Pattern Recognition*. 2004.
69. Cartoux, J., Lapreste, J., and Richetin, M., *Face authentication or recognition by profile extraction from range images.* Workshop on Interpretation of 3D Scenes, 1989: p. 194–199.
70. Gordon, G. *Face Recognition Based on Depth and Curvature Features*. in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. 1992.
71. Zhang, L., A. Razdan, G. Farin, J. Femiani, M. Bae, C. Lockwood *3D face authentication and recognition based on bilateral symmetry analysis.* The Visual Computer 2006. **22**(1): p. 43–55.
72. Lee, J.C.a.M., E. . *Matching range images of human faces*. in *IEEE International Conference on Computer Vision*. 1990. Osaka, Japan.

73. Horn, B.K.P., *Extended gaussian images.* Proceedings of the IEEE, 1984. **72**: p. 1671 - 1686.
74. Tanaka, H., Ikeda, M., and Chiaki, H. *Curvature-based surface recognition using spherical correlation principal directions for curved object recognition*. in *Third IEEE International Conference on Automatic Face and Gesture Recognition*. 1998.
75. Lee, N.F.a.A., *Correlation coefficients for random variables on a unit sphere or hypersphere.* Biometrika, 1986. **73**: p. 159 - 164.
76. G. Pan, Y.W., Y. Qi, and Z. Wu. *Finding Symmetry Plane of 3D Face Shape*. in *Proceedings of the 18th International Conference on Pattern Recognition*. 2006: IEEE Computer Society.
77. S. Changming, a.J.S., *3D symmetry detection using the extended Gaussian image.* IEEE Transaction on pattern Analysis and Machine Intelligence, 1997. **19**: p. 164.
78. S. Romdhani, V.B., T. Vetter. *Face Identification by Fitting a 3D Morphable Model using Linear Shape and Texture Error Functions*. in *The European Conf. on Computer Vision, LNCS* 2002.
79. Blanz, V.a.V., T. *A morphable model for the synthesis of 3D faces*. in *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics* 1999.
80. Huang, J., Heisele, B., and Blanz, V. *Component-based Face Recognition with 3D Morphable Models*. in *4th International Conference on Audio- and Video Based Biometric Person Authentication*. 2003.
81. W. Zhao, R.C. *3D Model Enhanced Face Recognition*. in *Int. Conf. on Image Processing*. 2000.
82. ftp://ftp.wisdom.weizmann.ac.il/pub/FaceBase/.
83. Beumier, C.a.A., M. *Automatic Face Verification from 3D And Grey Level Clues*. in *11th Portuguese Conference on Pattern Recognition*. 2000.
84. Y. Wang, C.-S.C., and Y.-K. Ho, *Facial features detection and face recognition from 2D and 3D images.* Pattern Recognition Letters, 2002. **23**: p. 1191 - 1202.
85. Besl, P.a.M., N., *A method for registration of 3D shapes.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992. **14**(2): p. 239–256.
86. Jarvis, C.-S.C.a.R., *A new representation for 3D object recognition.* International Journal of Computer Vision, 1997. **25**: p. 63 - 85.
87. Burges, J.C.a.C., *A Tutorial on Support Vector Machines for Pattern Recognition.* Data Mining and Knowledge Discovery, 1998. **2**: p. 121 - 167.
88. K. Chang, K.B., P. Flynn. *Face Recognition Using 2D and 3D Facial Data*. in *ACM Workshop on Multimodal User Authentication*. 2003. Santa Barbara, California.
89. Hesher, C., Srivastava, A., and Erlebacher, G. *A novel technique for face recognition using range imaging*. in *International Symposium on Signal Processing and Its Applications*. 2003.
90. C.Hesher, A.S., and G. Erlebacher,. *PCA of range images for facial recognition,*. in *In Proc of International Multiconference in Computer Science*. 2002. Las Vegas.
91. A. Bronstein, M.B., R. Kimmel, and A. Spira. *3D face recognition without facial surface reconstruction*. in *European Conference on Computer Vision*. 2004.
92. Yale Face Database, < http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html >

93. Chua, C., Han, F., and Ho, Y. *3D Human Face Recognition Using Point Signature*. in *4th IEEE IInternational Conference on Automatic Face and Gesture recognition*. 2000.
94. A. M. Coombes, R.R., A. Linney, V. Bruce, R. Fright. *Description and recognition of faces from 3D data*. in *SPIE The International Society for Optical Engineering*. 1992.
95. Alexander M. Bronstein, M.M.B., and Ron Kimmel, *Expression-Invariant 3D Face Recognition*, in *Audio- and Video-Based Biometric Person Authentication*, J.K.a.M.S. Nixon, Editor. 2003, Springer: Berlin / Heidelberg. p. 62-70.
96. D. P. Huttenlocher, G.A.K., and W. J. Rucklidge, *Comparing images using the Hausdorff distance.* PAMI, 1993. **15**(9): p. 850-863.
97. Bunke, B.A.a.H., *Classifying range images of human faces with Hausdorff distance.* ICPR, 2000. **2**: p. 809-813.
98. Ackermann, B.a.B., H., t. . C. *3D Face Recognition*. in *15th International Conference on Pattern Recognition*. 2000.
99. Pan, G., Wu, Z., and Pan, Y., *Automatic 3D face verification from range data.* 2003: p. 193–196.
100. Moreno, A., Sanchez, A., Velez, J., and Diaz, F. *Face recognition using 3D surface-extracted descriptors*. in *Irish Machine Vision and Image Processing Conference (IMVIP)*. 2003.
101. Hallinan, P., G. Gordon, A.L. Yuille, P. Giblin, D. Mumford *Two and three-Dimensional Pattern of the Face.* 1999.
102. Anshuman Razdan, G.F., Myung Soo-Bae and Mahesh Chaudhari *State of 3D Face Biometrics for Homeland Security Applications*, in *National Security (Handbooks on Information Systems)*, T.S.R. Hsinchun Chen, Ram Ramesh, Ajay Vinze and Daniel Zeng, Editor. 2007, Elsevier. p. 73-99
103. Myung Soo-Bae, A.R., and Gerald Farin. *Automated 3D Face Authentication and Recognition*. in *IEEE International Conference on Advanced Video and Signal based Surveillance*. 2007. London, UK.
104. Elyan, E. and Ugail, H. *Automatic 3D Face Recognition Using Fourier Descriptors*. in *International Conference on CyberWorlds*. 2009. Bradford, UK: IEEE Computer Society.
105. L. Xiaoguang, A.K.J., and D. Colbry, *Matching 2.5D face scans to 3D models.* IEEE Transactions Pattern Analysis and Machine Intelligence, , 2006. **28**: p. 31.
106. Bronstein, A., Bronstein, M., and Kimmel, R., *Three-dimensional face recognition.* International Journal of Computer Vision, 2005. **64**: p. 5–30.
107. Passalis, G., Kakadiaris, I., Theoharis, T., Toderici, G., and Murtuza, N, *Evaluation of 3D face recognition in the presence of facial expressions: an annotated deformable model approach.* 2005: p. 1022 – 1029.
108. J. Y. Cartoux, J.T.L., and M. Richetin, *Face authentification or recognition by profile extraction from range images.* IEEE Computer Society Workshop on Interpretation of 3D Scenes, 1989: p. 194 - 199.
109. Beumier, C.a.A., M., *Face verification from 3D and grey level clues.* Pattern Recognition Letters, 2001. **22**: p. 1321–1329.

110. Bronstein, A., Bronstein,M., and Kimmel, R. *Expression-invariant 3D face recognition*. in *International Conference on Audio- and Video-Based Person Authentication*. 2003.
111. K. I Chang, K.W.B., and P. J. Flynn, *Multimodal 2D and 3D biometrics for face recognition*. IEEE International Workshop on Analysis and Modeling of Face and Gestures, ed. K.W. Bowyer. 2003. 187 - 194.
112. Tsalakanidou, F., Tzocaras, D., and Strintzis, M., *Use of depth and colour eigenfaces for face recognition.* Pattern Recognition Letters, 2003. **24**: p. 1427–1435.
113. Xu, C., Wang, Y., Tan, T., and Quan, L. *Automatic 3D face recognition combining global geometric features with local shape variation information.* in *International Conference on Automated Face and Gesture Recognition*. 2004.
114. Y. Lee, J.S. *Curvature-based human face recognition using depth-weighted Hausdorff distance*. in *International Conference on Image Processing (ICIP)*. 2004.
115. Lu, X.a.J., A. *Intergrating range and texture information for 3D face recognition.* in *7th IEEE Workshop on Applications of Computer Vision*. 2005b.
116. Lee, Y., Song, H., Yang, U., Shin, H., and Sohn, K. *Local feature based 3D face recognition*. in *International Conference on Audio- and Video-based Biometric Person Authentication*. 2005.
117. Pan, G., Han, S.,Wu, Z., andWang, Y., *3D face recognition using mapped depth images.* 2005: p. 175–175.
118. Qatawnah S, I.S., Qahwaji R, and Ugail H *3D Face Recognition Based on Machine Learning*, in *IASTED International Conference on Visualization, Imaging and Image Processing (VIIP 2008)*. 2008: Palma de Mallorca, Spain.
119. Lee, J.a., E. *Matching range images of human faces*. in *International Conference on Computer Vision (ICCV*. 1990.
120. Chua, C.a.J., R., *Point signatures - a new representation for 3D object recognition.* International Journal of Computer Vision, 1997. **25**(1): p. 63–85.
121. Wong, H., Chueng, K., and Ip, H., *3D head model classification by evolutionary optimization of the extended gaussian image representation.* Pattern Recognition, 2004. **37**(12): p. 2307–2322.
122. Medioni, G.a.W., R. *Face recognition and modeling in 3D.* in *IEEE International Workshop on Analysis and Modeling of Faces and Gestures*. 2003.
123. Papatheodorou, T.a.R., D. *Evaluation of automatic 3D face recognition using surface and texture registration*. in *International Conference on Automated Face and Gesture Recognition*. 2004.
124. Nagamine, T., Uemura, T., and Masuda, I. *3D facial image analysis for human identification*. in *International Conference on Pattern Recognition*. 1992.
125. Wu, Y., Pan, G., and Wu, Z. *Face authentication based on multiple profiles extracted from range data*. in *Audio- and Video-Based Biometric Person Authentication*. 2003.
126. Mavridis, N., Tsalakanidou, F., Pantazis, D., Malasiotis, S., and Strintzis, M. *The hiscore face recognition application: Affordable desktop face recognition based on a novel 3D camera*. in *International Conference on Augmented Virtual Environments and 3D Images*. 2001.

127. Chang, K., Bowyer, K., and Flynn, P. *Face recognition using 2D and 3D facial data*. in *Multimodal User Authentication Workshop*. 2003.
128. Papatheodorou, T.a.R., D. *Evaluation of 3 D face recognition using registration and PCA*. in *Audio- and Video-based Biometric Person Authentication*. 2005.
129. Tsalakanidou, F., Malassiotis, S., and Strintzis, M. *Integration of 2D and 3D images for enhanced face authentication*. in *International Conference on Automated Face and Gesture Recognition*. 2004.
130. G¨okberk, B., Salah, A., and Akarun, L. *Rank-based decision fusion for 3D shapebased face recognition*. in *International Conference on Audio- and Video-based Biometric Person Authentication*. 2005.
131. Srivastava, A., Liu, X., and Hesher, C., *Face recognition using optimal linear components of face images.* Journal of Image and Vision Computing, 2003. **24**(3): p. 291–299.
132. Schwartz, E., Shaw, A., and Wolfson, E., *A numerical solution to the generalized mapmaker's problem: flattening nonconvex polyhedral surfaces.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 1989. **11**(9): p. 1005–1008.
133. Heisele, B., Serre, T., Pontil, M., and Poggio, T. *Component-based face detection* in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2001.
134. Ansari, A.a.A.-M., M. *3D face modelling using two orthogonal views and a generic face model*. in *International Conference on Multimedia and Expo*. 2003a.
135. Ansari, A.a.A.-M., M. *3D face modelling using two views and a generic face model with application to 3D face recognition*. in *IEEE Conference on Advanced Video and Signal Based Surveillance*. 2003b.
136. Kakadiaris, I., Passalis, G., Theoharis, T., Toderici, G., Konstantinidis, I., and Murtuza, N. *Multimodal face recognition: combination of geometry with physiological information*. 2005.
137. Yin, L.a.Y., M. *3D face recognition based on high-resolution 3D face modeling from frontal and profile views*. in *ACM Workshop on Biometric Methods and Applications*. 2003.
138. Kim, T.a.K., J., *Locally linear discriminant analysis for multimodally distributed classes for face recognition.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005. **27**(3): p. 318–327.
139. Li, Y., Gong, S., and Lidell,H. *Support vector regression and classification based multiview face detection and recognition*. in *International Conference on Face and Gesture Recognition*. 2000.
140. Cootes, T., Edwards, G., and Taylor, C. *Active appearance models*. in *European Conference of Computer Vision (ECCV)*. 1998.
141. Gross, R., Matthews, I., and Baker, S. *Eigen light-fields and face recognition across pose*. in *International Conference on Automatic Face and Gesture Recognition*. 2002.
142. Newell, J.F.B.a.M.E., *Texture and Reflection in Computer Generated Images.* CACM, 1976. **19**(10): p. 542-547.
143. Grother, P.J., *Face recognition vendor test 2002: Supplemental report*, in *Tech. Rep. NISTIR*. 2004, National Institute of Standards and Technology.

144. P.J. Phillips, P.J.G., R.J. Micheals, D.M. Blackburn, E. Tabassi, and J.M. Bone, *Face recognition vendor test 2002: Evaluation report*, in *Tech. Rep. NISTIR* 2003, National Institute of Standards and Technology.
145. *Information Access Division,National Institute of Standard and Technology, <http://face.nist.gov/frgc/>*. 2007.
146. Phillips, P.J. *Overview of the Face Recognition Grand Challenge*. in *Biometric Consortium Conference*. 2004. Crystal City, Arlington, VA
147. P. J. Phillips, P.J.F., T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, W. Worek. *Overview of the Face Recognition Grand Challenge*. in *International Computer Vision and Pattern Recognition (CVPR)* 2005.
148. Fahlmann, S.E., Lebiere, C., *Advances in Neural Information Processing System 2(NIPS-2)*, in *Touretzky, D.S.* 1989: Morgan Kaufmann, Denver. p. 524.
149. DTREG, *<http://www.dtreg.com/cascade.htm>*.
150. Shet, R.N., Lai, K.H., Edirisingh, E., Chung, P.W.H. *Pattren Recognition and Image Analysis , Lecture Notes in Computer Science*. in *Marques,J.S.,de la Pe'rez,B.N.,Pina,P*. 2005. Berlin: Springer.
151. Acir, N., and Guzelis,C. , *Automatic recognition of sleep spindles in EEG by using artificial neural networks,*. Expert.Syst.Appl. , 2004. **27**: p. 451-458.
152. Qahwwaji, R., and Colak, T. , *Automatic Short Term Solar Flare Prediction Using Machine Learning and Sunspot Associations,*. Springer, 2007. **241**: p. 195-211.
153. Qu, M., Shih, F.Y., Jing, J., and Wang, H. , *Automatic Solar Flare Detection Using MLP, RBF and SVM* Solar Phys. , 2003. **217**: p. 157-172.
154. Pal, M.a.M., P.M. , *Assessment of the effectiveness of support vector machines for hyperspectral data.* Futur.Gener.Comput.Syst., 2004. **20**: p. 1215-1225.
155. Huang, Z., Chen, H.C., Hsu, C.J, Chen,W.H., and Wu,S.S. , *Credit rating analysis with support vector machines and neural networks: a market comparative study*. Decis.Support Syst. , 2004. **37**: p. 543-558.
156. Distante, C., Ancona, N., and Siciliano, P., *Support Vector Machines for olfactory signals recognition.* Sensors and Actuators, Elsevier, 2003. **88**: p. 30-39.
157. Vapnik, V., *The Nature of Statistical Learning Theory.* Springer, NY, USA, 1999.
158. Chih-Chung Chang, a.C.-J.L. *LIBSVM : a library for support vector machines*. 2001< http://www.csie.ntu.edu.tw/~cjlin/libsvm>
159. Tom Mitchell, *Machine Learning*. 1997: McGraw Hill.
160. Skellam, J.G., *Studies in statistical ecology.* I. Spatial pattern, Biometrica, 1952. **39(**346-362).
161. P. J. Clark, F.C.E., *Distance to nearest neighbour as a measure of spatial relationships in populations.* Ecology, 1954. **35**: p. 445-453.
162. Fukunaga, K., *Introduction to Statistical Pattern Recognition*, in *Academic*. 1990: New York, USA. p. 220.
163. Allred, L.G. *JackKnife method for validating neural network models* in *IJCNN-91-Seattle International Joint Conference* 1991. Seattle, WA.
164. Fawcett, T., *An introduction to ROC analysis.* Pattern Recognition Letters, ElsevieR, 2006. **27**: p. 861-874.
165. Chernesky, M., Jang, D., Krepel, J., Sellors, J. & Mahony, J., *Impact of reference standard sensitivity on accuracy of rapid antigen detection assays and a leukocyte*

esterase dipstick for diagnosis of Chlamydia trachomatis infection in first-void urine specimens from men. Journal of Clinical Microbiology 1999. **37**: p. 2777 - 2780.
166. Watt, R.J.a.B.J.R., *Human Vision and Cognitive Science.* Cognitive Psychology Research Directions in Cognitive Science: European Perspectives 1989. **1**: p. 10-12.
167. Hill, F.S., *Computer Graphics Using Open GL*, ed. H. Prentice. 2001.
168. J. C. Hager, P.E., and W. V. Friesen, *Facial Action Coding System.UT: A Human Face*. 2002: Salt Lake City.
169. K. I. Chang, K.W.B., and P. J. Flynn., *Multiple nose region matching for 3d face recognition under varying facial expression.* IEEE Transaction on pattern Analysis and Machine Intelligence, 1992. **14**(2): p. 239-256.
170. Andrew, C.G., *Detection of Linear and Cubic Interpolation in JPEG Compressed Images.* Eastman Kodak Company.
171. Hutton, T., *Dense Surface Models of the Human Face*. 2004, University College London: London.
172. Rueckert, D., Frangi, A. F., and Schnabel, J. A., *Automatic construction of 3-D statistical deformation models of the brain using nonrigid registration.* IEEE Transactions on Medical Imaging, 2003. **22**(8): p. 1014–1025.
173. Lin, H.-T. *piaip's Using (lib)SVM Tutorial*. 2003 [cited; Available from: <http://www.csie.ntu.edu.tw/~r91034/svm/svm_tutorial.html >
174. E. Trucco, a.A.V., *Introductory Techniques for 3D Computer Vision*, ed. H.P. Prentice. 1998.
175. T.Vetter, V.B.a., *Face recognition based on fitting a 3D morphable model.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 2003. **25**:1063-1074.
176. C. BenAbdelkader, a.P.A.G., *Comparing and combining depth and texture cues for face recognition.* Image and Vision Computing, 2005. **23**: p. 339-352.
177. Ajmal Mian, M.B., Robyn Owens. *Automatic 3D Face Detection, Normalization and Recognition*. in *Third International Symposium on 3D Data Processing, Visualization and Transmission (3DPVT)*. 2006.
178. K.Hattori, S.M., and Y.Sato,, *Estimating pose of human face based on symmetry plane using range and intensity images*, in *Pattern Recognition*. 1998.
179. D. Colbry, a.G.S., *Canonical Face Depth Map: A Robust 3D Representation for Face Verification*, in *IEEE Conference on Computer Vision and Pattern Recognition CVPR*. 2007.
180. A. V. Tuzikov, O.C., and I. Bloch, *Brain Symmetry Plane Computation in MR images using inertia axes and optimization*, in *ICPR*. 2002: Quebec, Canada, .
181. M. Benz, X.L., T. Maier, E. Nkenke, S. Seeger, F.W. Neukam, G.Husler,, *The Symmetry of Faces.* VMV, 2002: p. 43-50.
182. H. Zabrodsky, S.P., and D. Avnir,. *Hierarchical Symmetry*. in *ICPR-92,The Hague*. 1992.
183. M.Kazhdan, B.C., D. Dobkin, A. Finkelstein, T.Funkhouser,. *A reflective symmetry descriptor*. in *7th European Conference on Computer Vision*. 2002.
184. *Object Files, <http://sebastien.mavromatis.free.fr/dl/I_TP5_spec_obj.html>*.
185. Gavin Bell, A.P., Mark Pesce, *The Virtual Reality Modeling Language*. 2005.

186. F. Hurtado, a.M.N., *Triangulations, visibility graph and reflex vertices of a simple polygon* Computational Geometry, ScienceDirect, 1996. **6**(6): p. 355-369.
187. J. Davis, S.R.M., M. Garr, M. Levoy, *Filling holes in complex surfaces using volumetric diffusion*, in *3D Data Processing Visualization and Transmission*. 2002. p. 428–861.
188. Meijering, E., *A chronology of interpolation: from ancient astronomy to modern signal and image processing.* Proceedings of the IEEE, 2002. **90**(3): p. 319–342.
189. Wu, G.P.a.Z., *3D face recignition from range data.* International journal of image and graphics, 2005. **5**: p. 573 - 593.
190. A. Howard, a.R.C., *Elementary Linear Algebra with Applications*. 9th ed, ed. Wiley. 2005.
191. William H. Press , S.A.T., William T. Vetterling , Brian P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*. 1992: Cambridge University Press. 408-413.
192. J. Piper, a.E.G., *Computing distance transformations in convex and non-convex domains.* Pattern Recognition, 1987. **20**(6): p. 599-615.
193. Weisstein, E.W. *Great Circle* From MathWorld -- A Wolfram Web Resource [cited; Available from: http://mathworld.wolfram.com/GreatCircle.html.
194. C. Li, A.B., J. Zhai, C. Chin,, *Exploring face recognition by combining 3D profiles and contours.* IEEE SoutheastCon, 2005: p. 576–579.
195. B. Gökberk, M.O.I., L. Akarun, *3D shape-based face representation and feature extraction for face recognition.* IVC 2006. **24**(8): p. 857–869.
196. S. Berretti, A.D.B., P. Pala, F. Silva Mata, *Face recognition by matching 2D and 3D geodesic distances*, in *MCAM07*. 2007. p. 444–453.
197. Kyong I., C., Kevin W. Bowyer, and Patrick J. Flynn, , *An evaluation of multi-modal 2d+3d face biometrics.* Trans.PAMI, 2005. **27**: p. 619–624.
198. Husken, M.B., M.; Gehlen, S.; Vonder Malsburg, C., *Strategies and Benefits of Fusion of 2D and 3D Face Recognition* 2005. **3**: p. 174-179.
199. Beymer, D.a.P., T., *Image representations for visual learning.* Science, 1996. **272**: p. 1905–1909.
200. Kirby, M.a.S., L., *Application of the Karhunen-Lo´eve procedure for the characterization of human faces.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 1990. **12**(1): p. 103–108.
201. Frangi, A., Rueckert, D., Schnabel, J., and Niessen, W., *Automatic construction of multiple-object three-dimensional statistical shape models: Application to cardiac modeling.* IEEE Transactions on Medical Imaging, 2002. **21**(9): p. 1151–1166.
202. V. Blanz, a.T.V. *A Morphable model for the synthesis of 3D faces*. in *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*. 1999.
203. Davies, R., *Learning Shape: Optimal Models for Analysing Natural Variability*. 2002, University of Manchester: Manchester.
204. Wang, Y., Peterson, B., and Staib, L. *Shape-based 3D surface correspondence using geodesics and local geometry*. in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2000.

205. Brett, A.a.T., C. *A method of automatic landmark generation for automated 3D PDMconstruction*. in *9th BritishMachine Vision Conference Proceedings*. 1998: Springer.
206. Brett, A., Hill, A., and Taylor, C., *A method of automatic landmark generation for automated 3D PDM construction.* Image and Vision Computing, 2000. **18**: p. 739–748.
207. Information Access Division's Face Recognition Grand Challenge <*http://face.nist.gov/frgc/*> , 2007.

Appendices

Appendix A: Construction of 3D statistical approach

The application of statistical models of 3D faces have shown promising results in face recognition [89, 126-128] and also outside face recognition [79]. The basic premise of statistical face models is that given the structural regularity of the faces, one can exploit the redundancy in order to describe a face with fewer parameters. To exploit this redundancy, dimensionality reduction techniques such as PCA can be used. A fundamental problem when building statistical models is the fact that they require the determination of point correspondences between the different shapes. The manual identification of such correspondences is a tedious and time consuming task. This is particularly true in 3D where the number of landmarks required to describe the shape accurately increases dramatically compared to 2D applications.

The correspondence problem

The key challenge of the correspondence problem in the context of face recognition is to find points on the facial surface that correspond, anatomically speaking, to the same surface points on other faces [138]. It is worth noting that early statistical approaches for describing faces did not address the correspondence problem explicitly [3, 139].

The gold standard to establish correspondence is by using manually placed landmarks to mark anatomically distinct points on a surface. As this can be a painstaking and error-prone process, several authors have proposed to automate this by using a template with annotated landmarks. This template can be then registered to other shapes and the landmarks can be propagated to these other shapes [140, 141]. Similarly, techniques such as optical flow algorithm that computes correspondence between two faces without the need of a morphable model can be used for registration [142]. Also, correspondences between 3D facial surfaces can be estimated by using optical flow on 2D textures to match anatomical features to each other [79]. Some work has been done on combining registration techniques with a semi-automatic statistical technique, such as active shape models, in order to take advantage of the strengths of each [143]. Yet another approach defines an objective function based on minimum description length (MDL) and thus treats the problem of correspondence estimation as an optimization problem [144]. Another way of establishing correspondence between points on two surfaces is by analyzing their shape. For example, curvature information can be used to find similar areas on a surface in order to construct 3D shape models [145]. Alternatively, the surfaces can be decimated in a way that eliminates points from areas of low curvature. High curvatures areas can then assumed to correspond to each other and are thus aligned [146, 147].

Appendix B: FRGC Data Set

Data for the FRGC was collected at the University of Notre Dame. The FRGC data set is part of an ongoing multi-modal biometric data collection. A subject session is the set of all images of a person taken each time a person's biometric data is collected. It consists of (four controlled still images, two uncontrolled still images and one 3D image) [148]. The controlled images were taken in a studio setting, are full-frontal facial images taken under two lighting conditions (two or three studio lights) and with two facial expressions (smiling and neutral). The uncontrolled images were taken in varying illumination conditions; e.g., hallways, atria, or outdoors [147]. Each set of uncontrolled images contains two expressions, smiling and neutral. The 3D images were taken under controlled illumination conditions appropriate for the Vivid 900/910 sensor, which are not the same as the conditions for the controlled still images. In the FRGC, a 3D image set includes both range and texture channels [144].

Table 1 includes a summary of the sizes of the faces for the uncontrolled, controlled, and 3D image categories. For comparison, the average distance between the centres of the eyes in the FERET database is 68 pixels with a standard deviation of 8.7 pixels [147]. Size is measured in pixels between the centres of the eyes. Reported are mean, median, and standard deviation. And Figure 1 shows details of the data used in the FRGC experiments.

Table 1 Size of faces in the validation set imagery according to category.

	Mean	Median	Std. Dev.
Controlled	261	260	19
Uncontrolled	144	143	14
3D	160	162	15

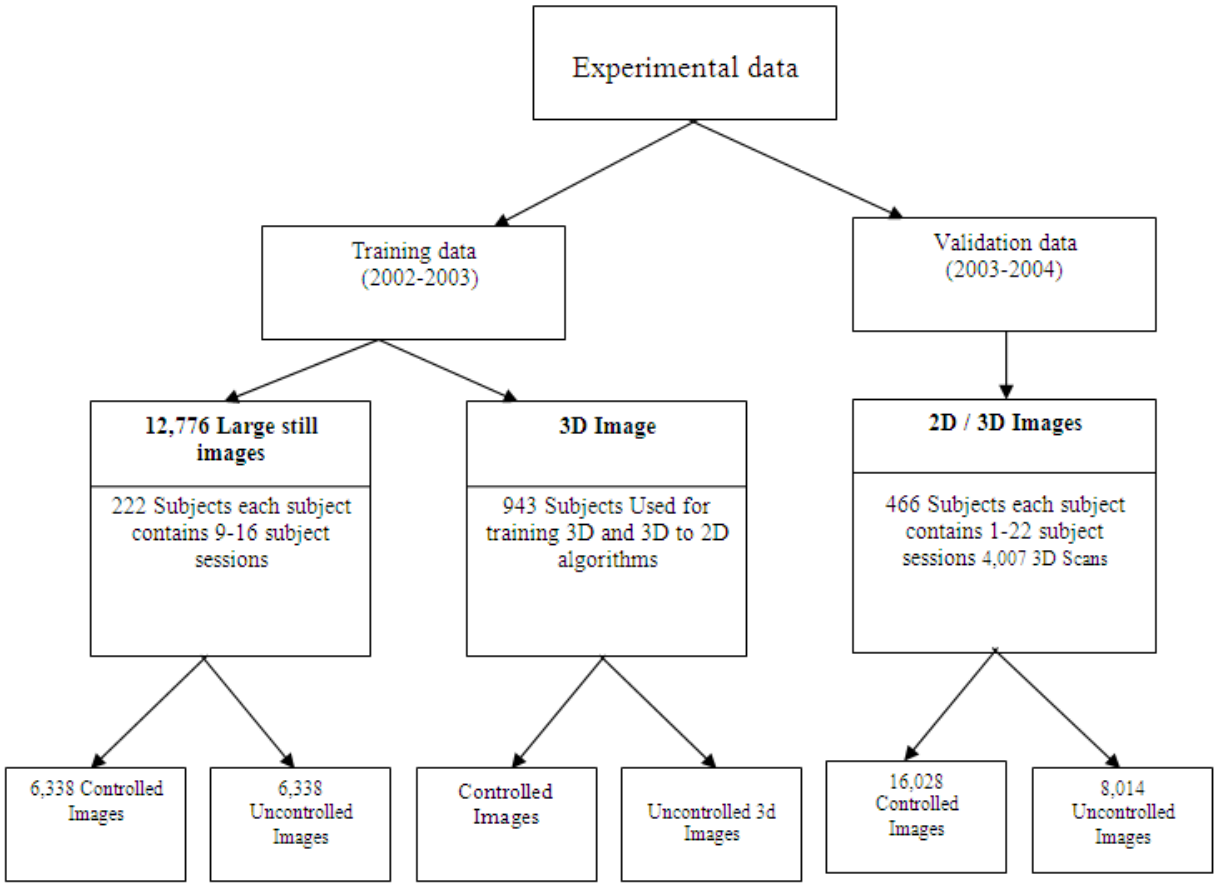


Figure 1 The FRGC dataset.

And some of FRGC installation difficulties are mentioned in Section 3.7.

Appendix C: Triangulation Algorithm

In order to have a clear face representation, the cloud points must be joined together to form the mesh, thus, a better face observation is obtained. This can be done using the *triangulation algorithm*.

The triangulation was devised by Boris Delaunay in 1934 [35]; every three points generate a triangle (providing that they are not on a straight line). The circumcircle (the circle that contains all the vertices of the triangle) must not contain any other vertex; otherwise the Delaunay triangulation condition will not be met. Figure 1 [35], shows a Delaunay triangulation in a space with the circumcircles.

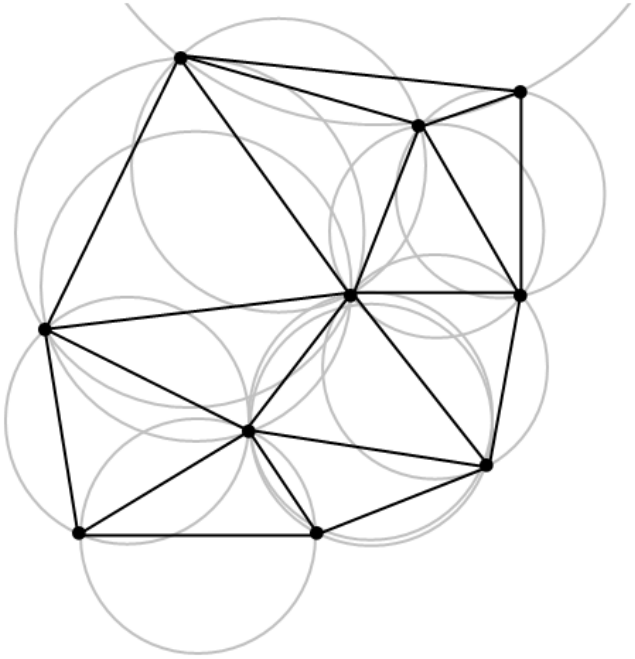


Figure 1 – Delaunay triangulation with circumcircles

Moreover, triangulation can be applied on every simple polygon with a number of vertices *n* [36]. The possibility of applying triangulation can be justified by the fact that every

polygon has a diagonal if the polygon contains at least one convex vertex [36]. A vertex is called a convex, if the internal angle of the same vertex $v^o \leq \pi$ [37]. There are three facts about polygons:

- Every polygon has to have at least one strictly convex vertex [36] [37].
- Every polygon has diagonal, if $n \geq 4$ [36].
- Every polygon (that can have at least one diagonal) can be divided into triangles by placing those diagonals [36].

There are many ways for triangulating a polygon but all of them share the fact that the number of the generated diagonals ND = n – 3, and the number of the resultant triangles NT = n – 2 [36]. Figure 2 proves this theory:

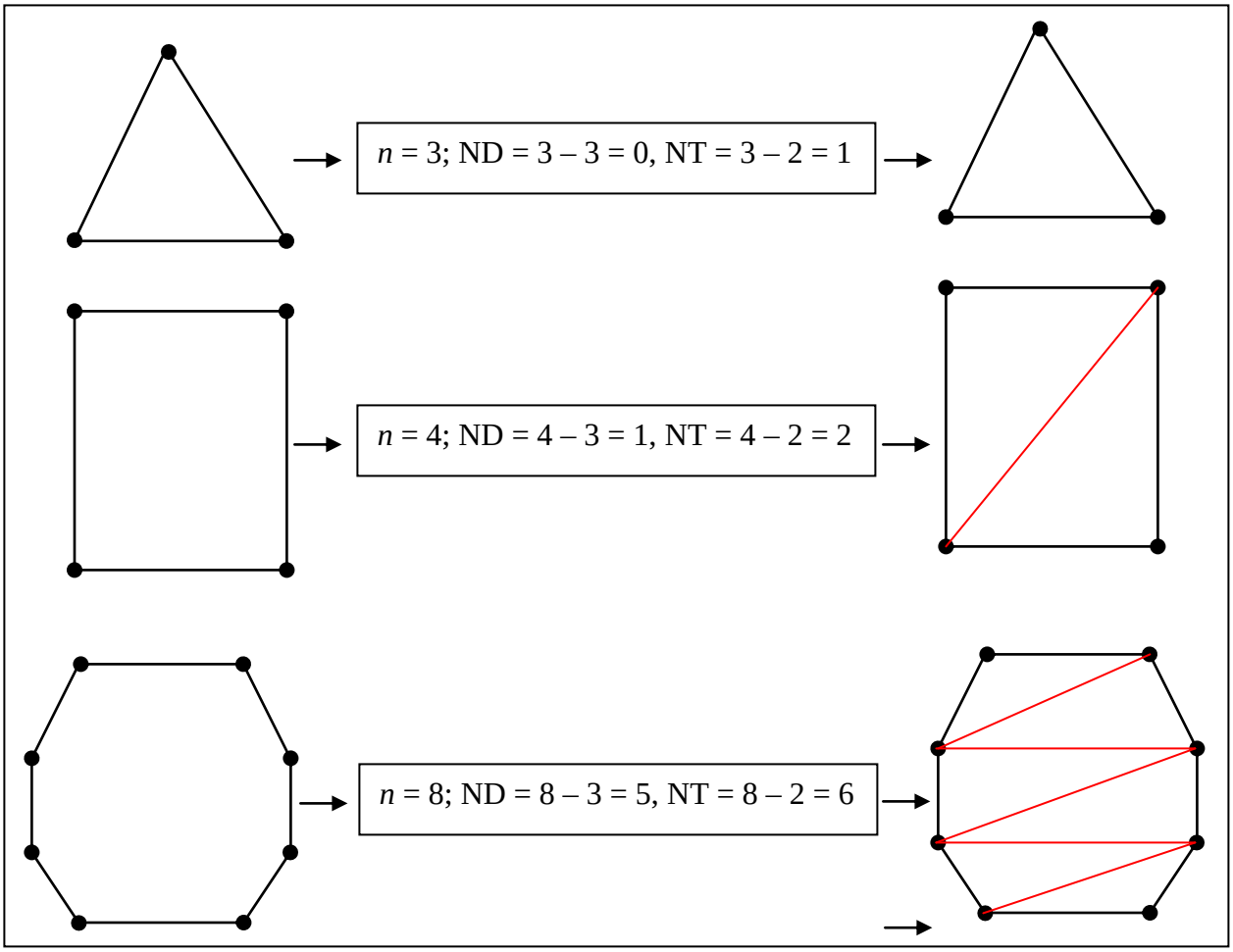
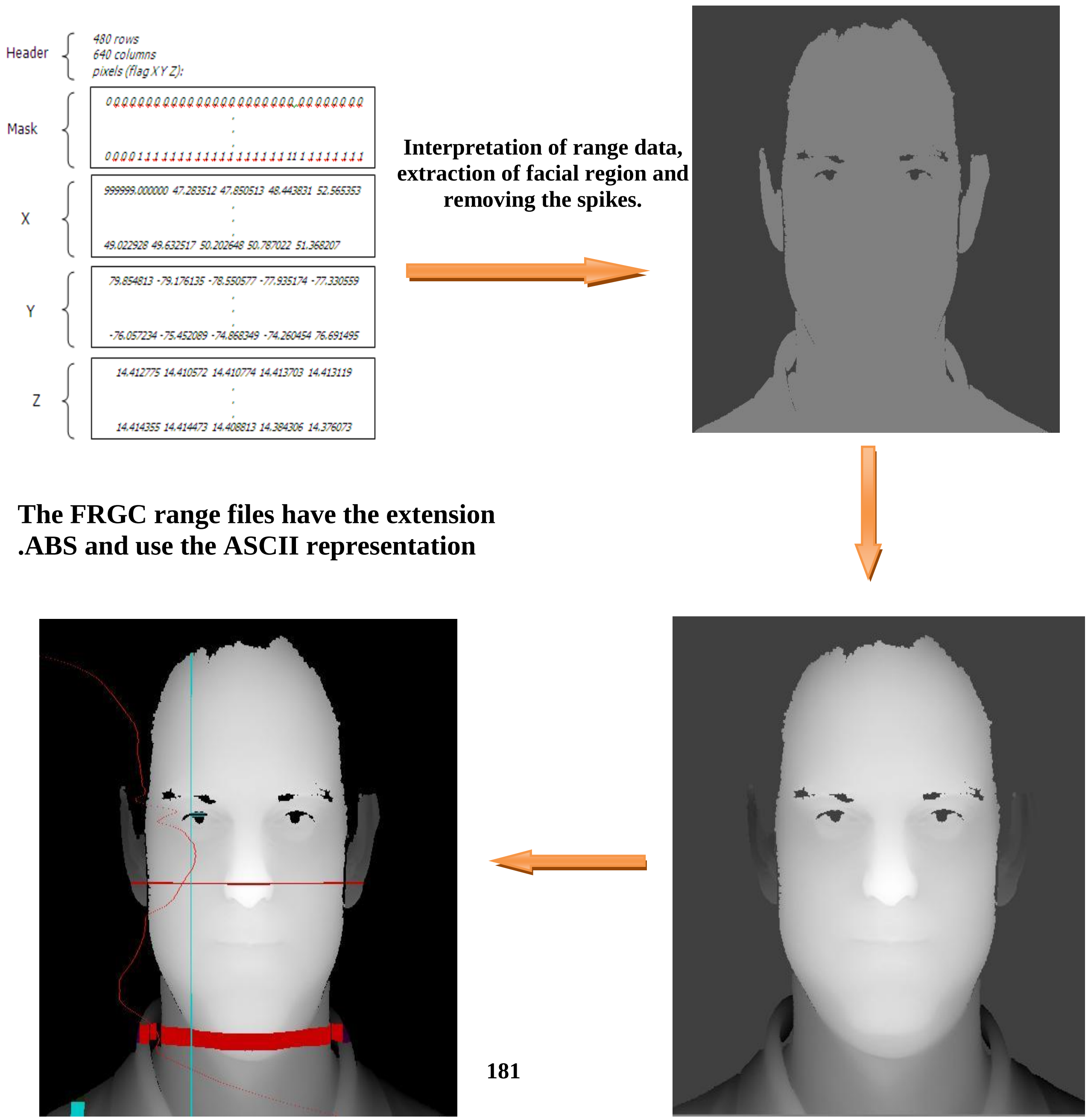
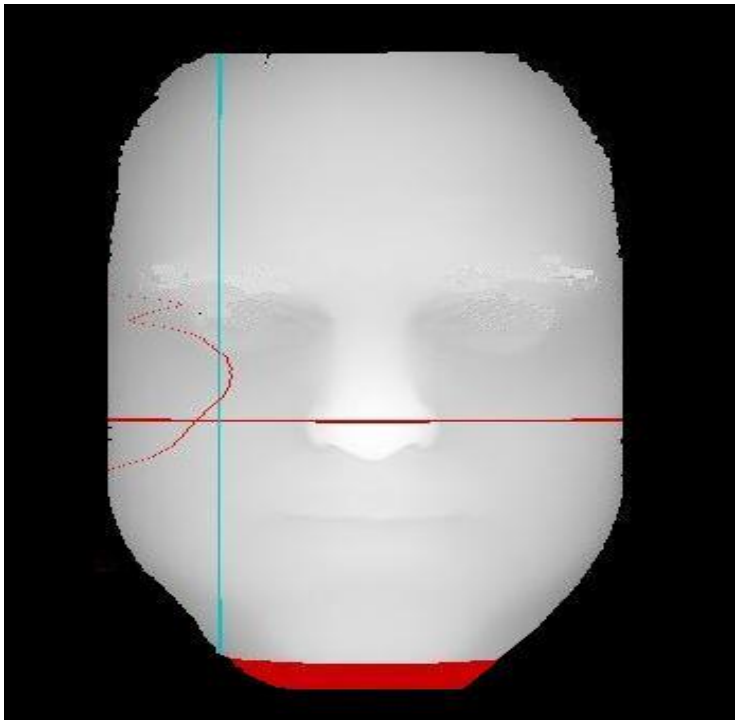


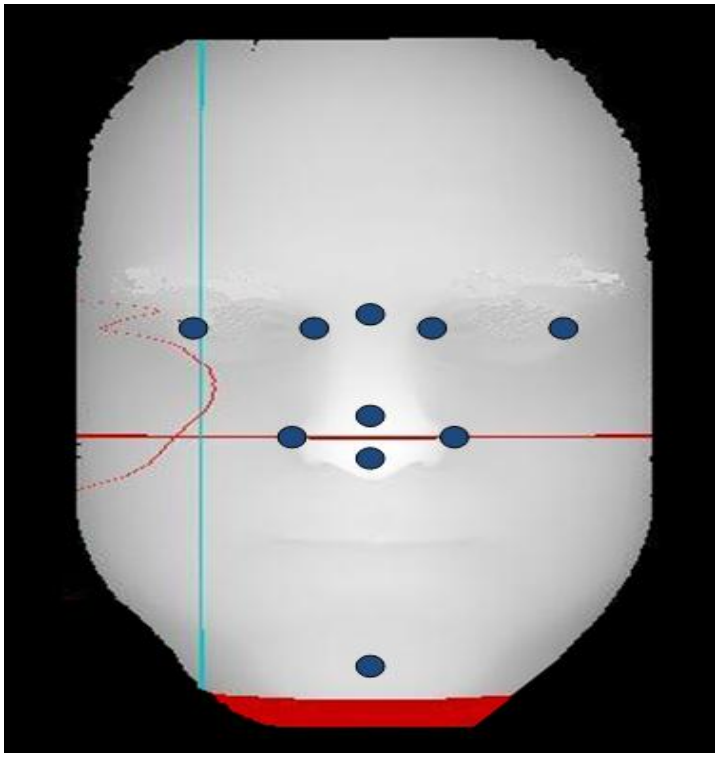
Figure 2 – Polygons with their diagonals

Appendix D: 3D Facial Feature Extraction Example

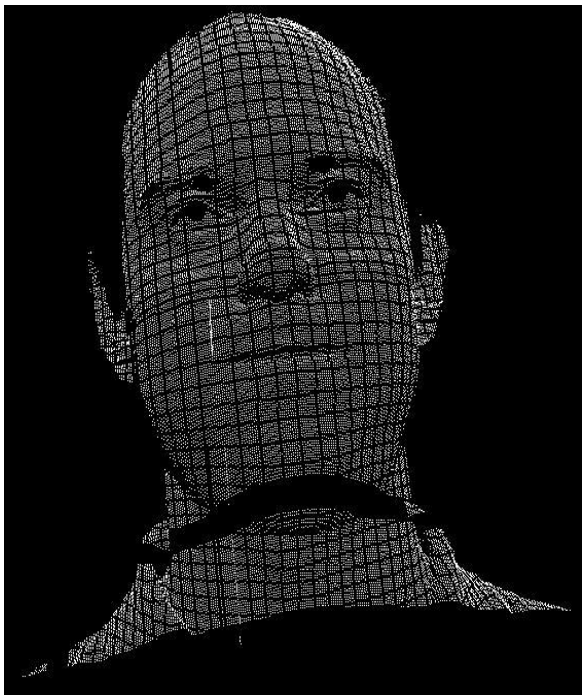
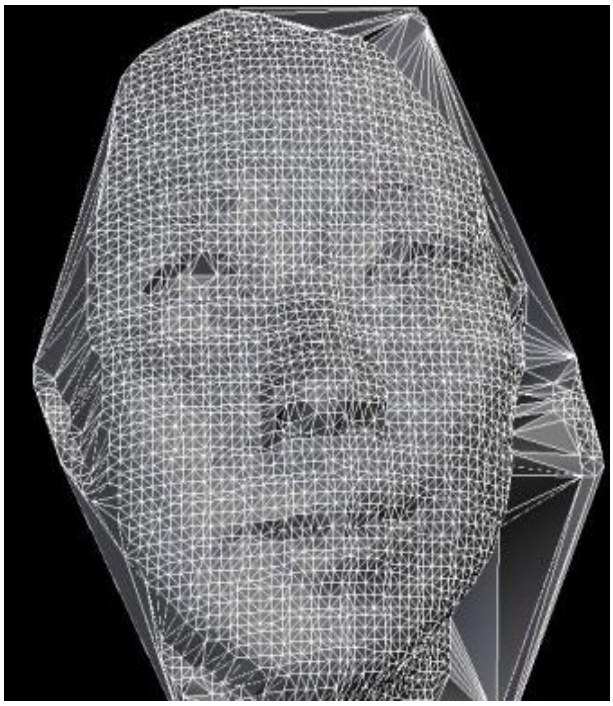
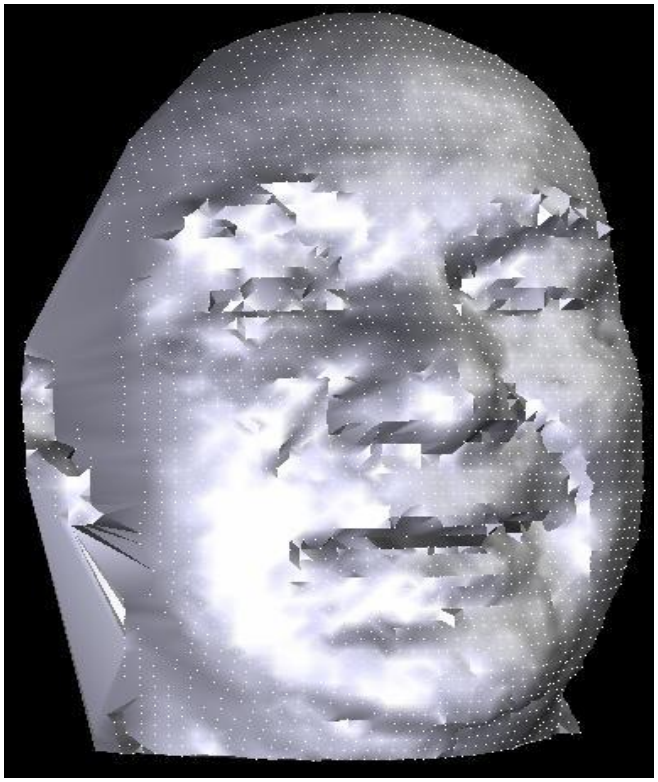
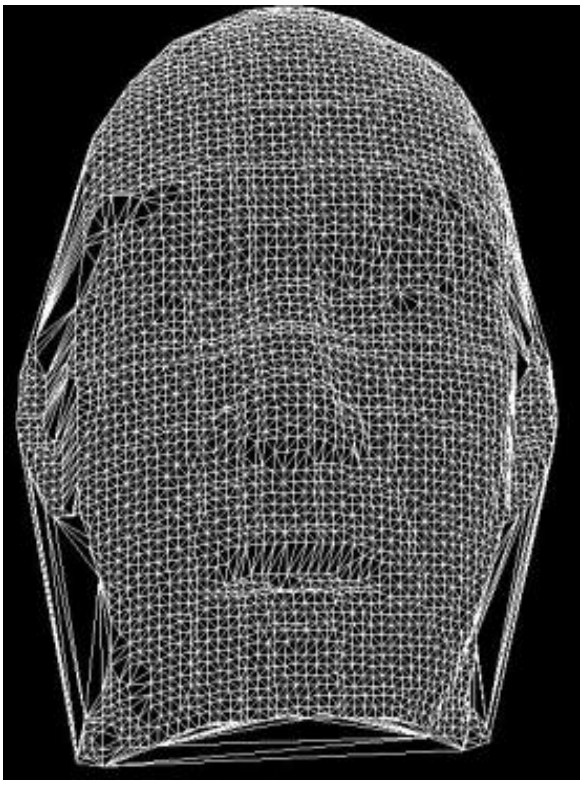




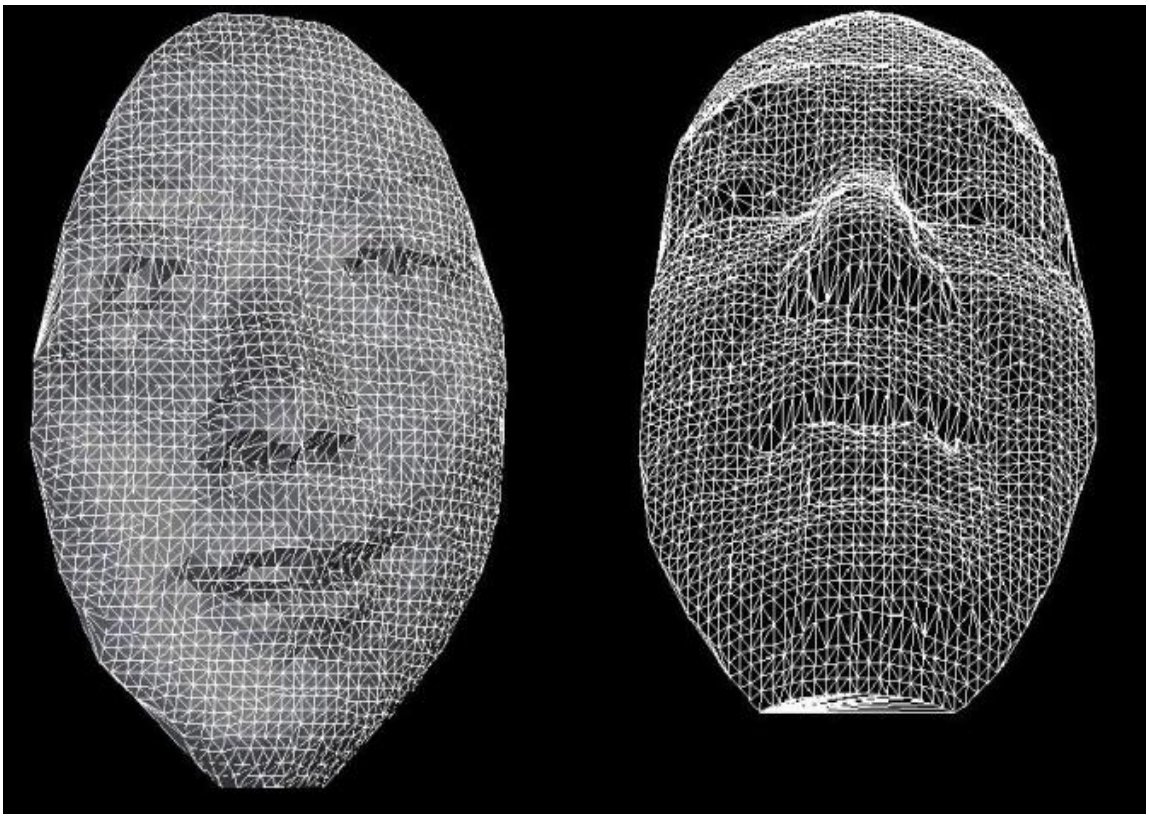
Cropping the Z data, filling the holes and feature extraction

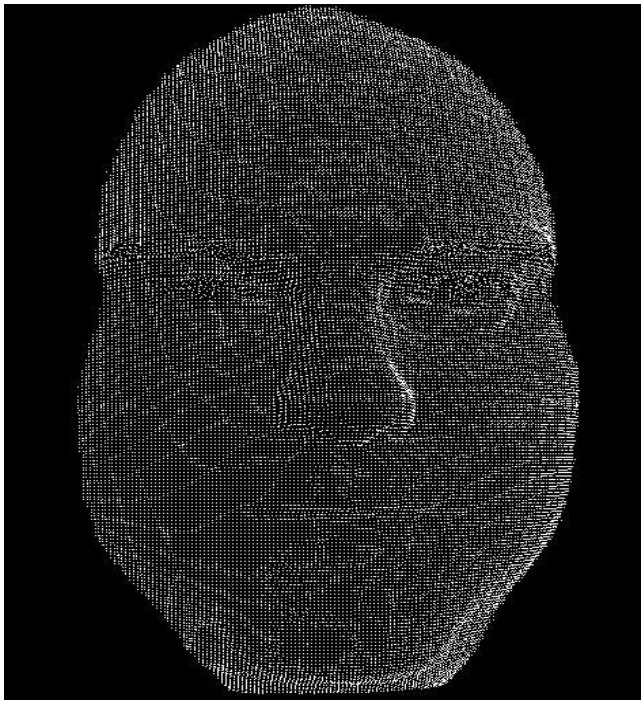


The 10 manually selected features chosen because of their anatomical distinctiveness

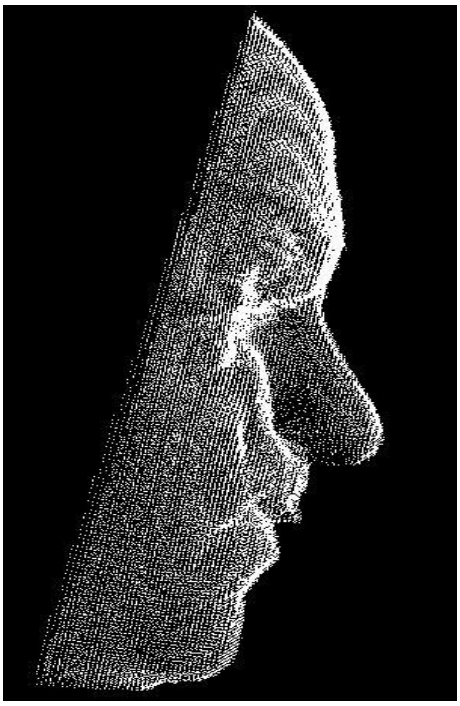
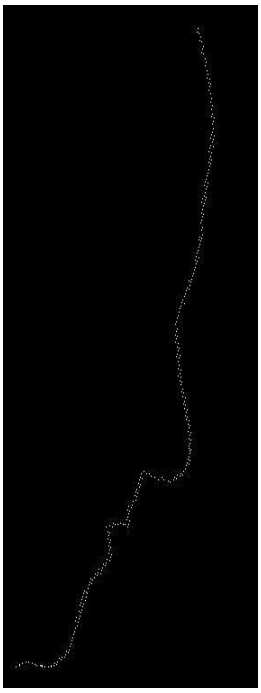
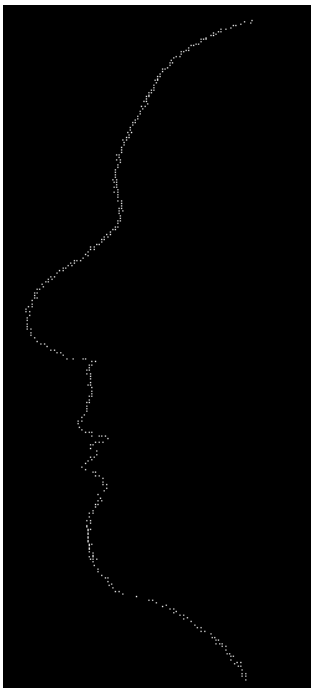
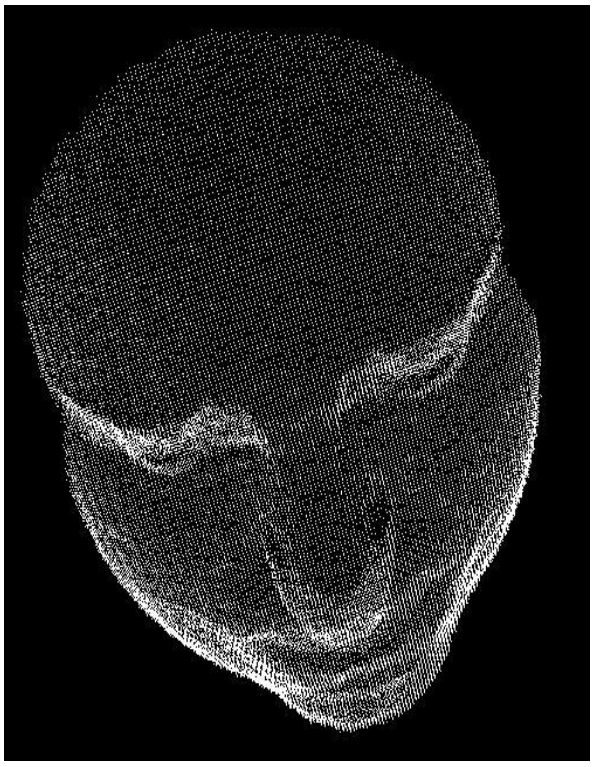
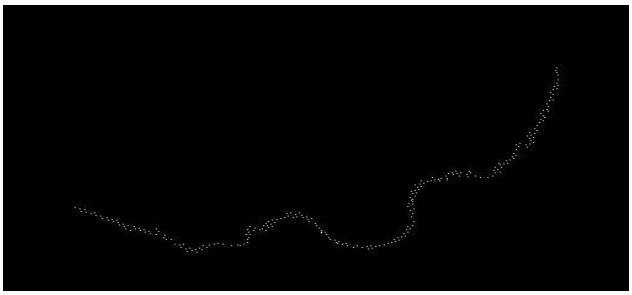
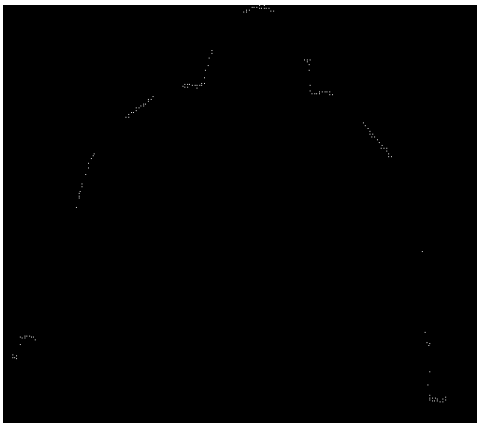





Create the OBJ and VRML files

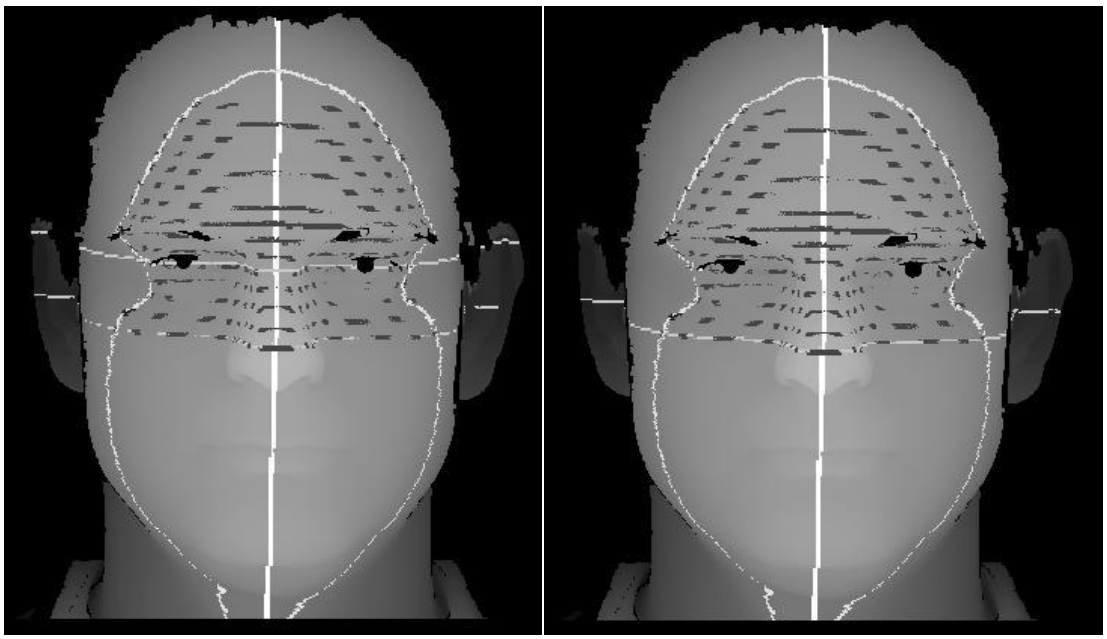




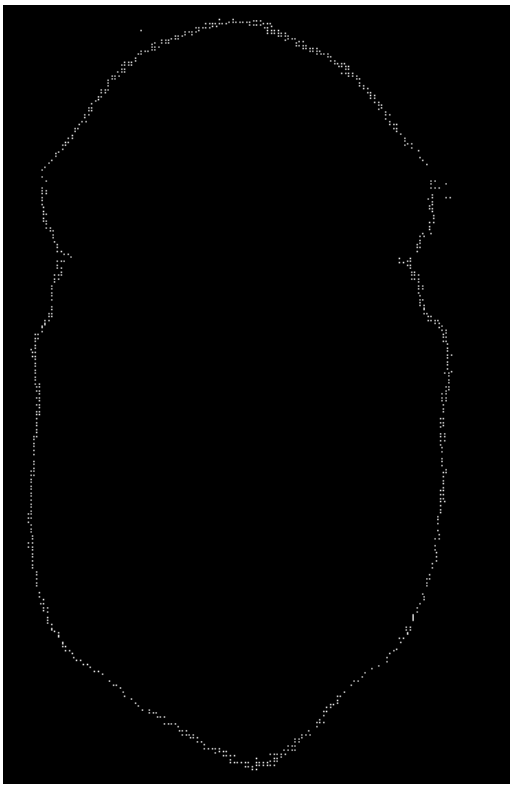
Removing the sharp spikes, fill the gaps, smooth the surface and extract a facial region

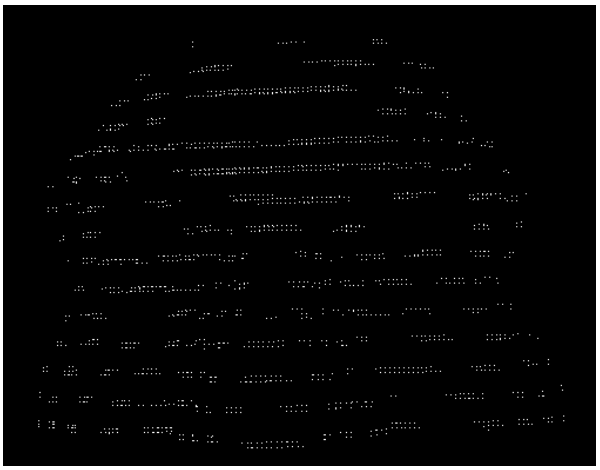
Compute the symmetry plane for the purpose of finding the tip of nose, nose-bridge and the bottom of nose after that extracting the eye profile and the inner



Symmetry plane and Eye profile identifications for 3D Facial Feature Extraction



Frontal Profile



Symmetry points within a square tube included in calculating the reflection measure