THREE DIMENSIONAL FACE RECOGNITION

by

Berk Gökberk

B.S, in CmpE., Boğaziçi University, 1999

M.S, in CmpE., Boğaziçi University, 2001

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

Graduate Program in Computer Engineering

Boğaziçi University

2006

ACKNOWLEDGEMENTS

This thesis you are about to read is just a mere summary of a great deal of effort carried out over the last few years. During this period, I have benefited from the support of many people to whom I should express my gratitude. First and foremost, I would like to thank Prof. Lale Akarun for her invaluable guidance, support and patience. I am indebted to Prof. Bülent Sankur and Prof. Ethem Alpaydın for their helpful comments and feedbacks that have greatly improved the work. Technically, without the contributions of Okan İrfanoğlu and Helin Dutağacı, some parts of the thesis would not be complete. I thank them all for their scientific contributions. I am also grateful to Pınar Santemiz for her careful reading of the manuscript, humble suggestions, and corrections. This thesis could never have been written without the wonderful environment of my department. This is greatly due to Prof. Cem Ersoy and my colleagues. Specially, I would like to thank Prof. Cem Ersoy for his perfect philosophy of the balance between scientific research and fun where the latter significantly improved the quality of the former. My deep thanks go to my friends: Oya Aran, Onur Dikmen, Burak Gürdağ, Itır Karaç, Rabun Koşar, Atay Özgövde, Albert Ali Salah, Pınar Santemiz, Burak Turhan, Aydın Ulas and many others which make my life in the department enjoyable. Without their spiritual assistance, this thesis could never see the light of day. I also wish to thank my psyhoacoustician Osman Kaytazoğlu who helped me to explore new acoustical worlds that ultimately had a side effect of improving my academic motivation. Here, my iPod and its fantastic playlist deserve a special thank too.

Finally, I am most grateful to my parents and Gamze Esen who provided endless love and support. I would like to mention that without my mother's insistence, I might not be pursuing my Ph.D. career. Many thanks to her for showing me the truth. Words are simply not enough to express my gratitude to Gamze Esen who always believed in me, and recovered me when I find myself in the gloom. This thesis is dedicated to my parents and Gamze Esen.

ABSTRACT

THREE DIMENSIONAL FACE RECOGNITION

In this thesis, we attack the problem of identifying humans from their three dimensional facial characteristics. For this purpose, a complete 3D face recognition system is developed. We divide the whole system into sub-processes. These sub-processes can be categorized as follows: 1) registration, 2) representation of faces, 3) extraction of discriminative features, and 4) fusion of matchers. For each module, we evaluate the state-of-the art methods, and also propose novel ones. For the registration task, we propose to use a generic face model which speeds up the correspondence establishment process. We compare the benefits of rigid and non-rigid registration schemes using a generic face model. In terms of face representation schemes, we implement a diverse range of approaches such as point clouds, curvature-based descriptors, and range images. In relation to these, various feature extraction methods are used to determine the discriminative facial features. We also propose to use local region-based representation schemes which may be advantageous in terms of both dimensionality reduction and for determining invariant regions under several facial variations. Finally, with the realization of diverse 3D face experts, we perform an in-depth analysis of decision-level fusion algorithms. In addition to the evaluation of baseline fusion methods, we propose to use two novel fusion schemes where the first one employs a confidence-aided combination approach, and the second one implements a two-level serial integration method. Recognition simulations performed on the 3DRMA and the FRGC databases show that: 1) generic face template-based rigid registration of faces is better than the non-rigid variant, 2) principal curvature directions and surface normals have better discriminative power, 3) representing faces using local patch descriptors can both reduce the feature dimensionality and improve the identification rate, and 4) confidence-assisted fusion rules and serial two-stage fusion schemes have a potential to improve the accuracy when compared to other decision-level fusion rules.

ÖZET

ÜÇ BOYUTLU YÜZ TANIMA

Bu tezde, üç boyutlu (3B) bir yüz tanıma sistemi geliştirilmiştir. Önerilen tanıma sistemi 1) çakıştırma, 2) betimleme, 3) öznitelik çıkarma, ve 4) karar tümleştirme kısımlarından oluşmaktadır. Yaptığımız çalışmada bu kısımların herbiri incelenmiş, ve bu alt problemler için yeni çözümler sunulmuştur. Önerilen yöntemlerin her biri standart algoritmalar ile karşılaştırılmıştır. 3B yüzlerin karşılaştırılması ve benzerlik derecelerinin bulunması için kayıtlama safhası önemli bir yere sahiptir. Yaptığımız çalışmada, yüzlerin ortalama bir yüz modeli kullanılarak çakıştırılması önerilmiştir. Ortalama yüz modelinin kullanımı çakıştırma safhasının zamansal karmaşıklığını oldukça azaltmaktadır. Hareketli bir yapıya sahip yüz yüzeylerinin çakıştırılması için katı ve katı olmayan varsayımlara sahip iki farklı çakıştırma yöntemi önerilmiştir. Yaptığımız tanıma ve doğrulama deneylerinde katı yüzey varsayımına dayalı Döngülü Yakın Nokta (DYN) yönteminin daha iyi sonuç verdiği görülmüştür. Yüzlerin betimlenmesi için nokta kümeleri, yüzey kıvrımları, ve derinlik imgeleri gibi çeşitli yöntemler denenmiştir. Her betimleme yöntemiyle uyumlu farklı öznitelik çıkarımları yapılmıştır. 3D RMA ve FRGC yüz kütüphanerinde yapmış olduğumuz deneylerde, yüzey normallerinin ve kıvrım doğrultularının daha iyi tanıma başarımına sahip oldukları gösterilmiştir. Tezde ayrıca, birden fazla tanıma algoritmasının kullanıldığı durumlarda, bu tanıyıcıların karar seviyesinde birleştirilmesinin yararlı olduğu gösterilmiştir. Standart tümleştirme algoritmalarına ek olarak, güvenilirliğe dayalı ve iki-seviyeli tümleştirme yöntemleri önerilmiştir. Öğrenme kümesinin az olduğu durumlarda güvenilirlik tabanlı yöntemin, diğer durumda ise iki-seviyeli tümleştirme yönteminin diğer yöntemlerden iyi tanıma başarımı gösterdiği gösterilmiştir. Tezde son olarak, yüzlerin yerel bölgelere ayrılarak betimlenmesi ile ilgili çalışmalar yapılmıştır. Yerel betimleme yöntemlerinin hem öznitelik boyutlarında azalmayı sağladığı hem de yüz yüzeylerindeki yerel değişimlere karşı daha dayanıklı olduğu ve böylece tanıma başarımını arttırdıkları gösterilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS/ABBREVIATIONS

H	Mean curvature
K	Gaussian curvature
n_k^i	3D unit surface normal
p^i	(x, y, z) coordinates of i^{th} 3D point
S	Shape index
κ_1	Minimum curvature
κ_2	Maximum curvature
Λ_i	3D face of the i^{th} individual
Φ	Feature subset
Ψ	Patch-based representation of a face
ρ_i	Principal curvature direction
BC	Borda count
BIF	Best-individual selection algorithm
CCR	Correct classification rate
CMC	Cumulative match characteristics
EER	Equal error rate
FAR	False acceptance rate
FRR	False rejection rate
ICA	Independent component analysis
ICP	Iterative closest point algorithm
LDA	Linear discriminant analysis
NMF	Nonnegative matrix factorization
PCA	Principal component analysis
ROC	Receiver operating characteristics
SFBS	Sequential floating backward search

1. Introduction

Recognizing humans from their biological and behavioral characteristics has always been a crucial need for secure applications. A measurable characteristic of a human is commonly called as a biometric, and several special biometrics are suitable for differentiating persons from others. Biometrics such as faces, iris patterns, hand geometry, fingerprints, speech, retina, dynamic signature, and gait are among the most frequently used modalities. Depending on the needs of the application, one can choose an appropriate biometric. For instance, high-security applications use iris patterns or fingerprints because these modalities offer better discriminatory information among humans. Automatic recognition of humans from their facial characteristics has a special importance due to several reasons: 1) no contact with the subject is required, 2) faces can be sensed easily with available cameras. On the other hand, automatic identification systems that use facial biometrics have to deal with important difficulties. These difficulties stem from the following reasons: 1) facial images may have lighting, expression and pose variations, 2) external factors such as glasses, hats, makeup and hair change the appearance of a face drastically, 3) faces change over time. The negative effect of these factors is generally significant if we consider the non-cooperative nature of the face acquisition phase.

Traditional approaches employed in face recognition systems generally use 2D static intensity images. However, due to the previously mentioned difficulties, many of these systems may not provide acceptable accuracies under more realistic operating conditions. Especially in adverse situations, intra-class facial variations, i.e., differences between the two images of the same person, are larger than the inter-class facial variations, i.e., differences between the images of different persons. This intra-class and inter-class variation problem makes the face recognition task as one of the most challenging pattern recognition problems. In terms of pattern recognition viewpoint, face recognition problem has several unique properties which further complicate the task: 1) feature dimensionality is so high that without the use of efficient feature selection and feature extraction algorithms, it is almost impossible to design a successful classi-

fication system, 2) sample size per class is scarce which makes it harder for a learning function to generalize to unknown examples. In terms of computer vision point of view, representation of faces also encounters significant obstacles such as 1) the detection and normalization faces, 2) extraction of discriminative features, and 3) correction of illumination differences. Therefore, in the classical pattern classification chain, the feature extraction phase is the most studied part in face recognition systems, and most of the time state-of-the-art pattern classifiers are used at the recognition phase. However, we see the dominance of instance-based classification algorithms, especially the k-nn algorithm, in face recognition systems.

In the biometrics literature, *recognition* is a general term which includes *identification* and *verification*. However, recognition mostly refers to identification in many papers, and *authentication* is used instead of verification. In identification, the task is to find the ID (class) of the person from the *template database*. Template database is the collection of biometric features of the previously enrolled users, and may be called as training set or *gallery set*. In *closed set* identification, we assume that the gallery set contains the biometric templates of all of the users. On the other hand, in an *open set* identification scenario, the template database may not contain the biometric samples of the unknown person. Therefore, the system should also provide an output indicating that the ID of the unknown person is not in the database. The open set identification scenario is useful in *watchlist* type of applications, where the aim is to detect whether the person is in the wanted list or not. In identification, the biometric template of *the client* (test person, or probe) should be matched with all of the templates in the *gallery set* (training set). The *matcher* simply calculates the similarity between two biometric features. The ID of the gallery template whose similarity is greatest to the client is given as the output ID. If the ID of the client and the ID of the found gallery template is the same, then it is said to reach a correct identification; otherwise misclassification occurs. In open set identification, an extra decision should also be produced indicating that the ID of the client is not present in the template database. In verification or authentication, the client provides both his biometric template and ID to the system. Then, the task of the authenticator is to accept or reject the claimed identity. In an authentication scenario, there is no need to match the client's template with all of the

templates in the gallery set. It suffices just to match with the gallery templates of the claimed ID. However, authentication systems should learn a general or a subject-specific threshold in order to output an accept or a reject decision. In general, the performance characteristics of identification systems are presented via rank-1 or rank-N statistics. If the ID of the client is found at the first N similar gallery templates, then it is said to have a correct classification. Cumulative statistics obtained from rank-1 to rank-N experiments are usually summarized in *cumulative match characteristics (CMC)* curves. In verification systems, performance characteristics are summarized by two measures: *false accept rate (FAR)* and *false reject rate (FRR)*. FAR denotes the proportion of imposters accepted, and FRR denotes the proportion of genuine users rejected. Changing the *threshold parameter* in the verification system, one can obtain different FAR and FRR. It is therefore common to summarize the performance of a verification system using different thresholds and plot the FAR/FRR values in *receiver operating characteristics* (ROC) curves. Another frequently used technique is to report the *equal error rate (EER)* where FAR equals FRR.

Identification and verification systems that use facial characteristics have shown great promise under controlled environments. However, under more realistic conditions they can not guarantee acceptable accuracy. In order to improve the recognition performance of the face recognition systems, the search for other modalities is indispensable. For this purpose, researchers now try to incorporate additional information coming from videos, infra-red sensors, and 3D acquisition devices. With the availability of 3D sensors, a great deal of effort is devoted to developing recognition systems that use 3D shape of a human facial surface [1]. In the last five years, a rapid increase for the need to design 3D face recognition algorithms has taken place both in academy and industry. However, it is clearly visible that the 3D face recognition technology is at the beginning steps. The motivation to use 3D technology was to overcome the disadvantages of 2D face recognition systems that arise especially from significant pose, expression and illumination differences. However, with the exception of few recent works, most of the 3D systems generally study controlled frontal face recognition. With the construction of bigger 3D face databases that contain enough samples for different illumination, pose, and expression variations, it is expected to develop more realistic 3D face recognition

systems.

3D face recognition concept usually contains three main categories: 1) 3D-to-3D, 2) 3D-to-2D or 2D-to-3D, and 3) 2D-to-2D via 3D. With the exception of the 3D-to-3D case, the other two categories are usually referred to as 3D-assisted face recognition since they specifically involve 2D images. In this thesis, we refer to the 3D-to-3D case as 3D face recognition, and will not mention other approaches in the other two categories. It is just sufficient to note that these approaches benefit from 3D head or face models in the matching process of 2D images. In Chapter 2, we provide a detailed literature survey of 3D face recognition algorithms according to their use of face representation methods.

1.1. Contributions of the Thesis

In this thesis, we concentrate on the fundamental problems of 3D face recognition algorithms, provide comparative analysis of several approaches, and offer novel solutions for each of the problems analyzed. The contributions can be grouped according to the following categories:

- *Evaluation of facial registration techniques:* Efficient registration of facial surfaces is crucial for any 3D face recognition system. Without the use of registration and alignment phases, it is very hard to define similarity between faces. The nature of the face recognition problem, the similarity of faces as 3D objects, makes the efficient registration part as an important requirement as opposed to more general 3D object classification tasks. However, facial surfaces have a free-form characteristic which complicates the task. In addition, typical deformations due to expression variations drastically change the structure of the facial surface. We propose two different facial registration systems where the first one assumes only rigid transformations, and the second one allows non-rigid deformations also [2, 3]. These two registration methods are presented in Chapter 3.
- *Evaluation of representation and feature extraction techniques:* 3D raw facial data coming from acquisition devices can be represented in various forms. Addi-

tionally, depending on the employed representation method, a number of feature extractors can be utilized to extract distinctive facial features. The combination of representation and feature extraction methods constitute the core of the 3D recognition algorithm. We evaluate the benefits of different recognition algorithms according to their identification power, and also propose novel ones [2, 4]. Chapter 4 is devoted to the explanation of these algorithms.

- *Evaluation of feature selection techniques for face recognition:* Local feature-based approaches offer viable alternatives to holistic approaches. Their power stems from their ability to robustly represent faces locally. In order to harness this property, we formulate the face representation task through the use of local features, and apply this methodology to both 2D and 3D face recognition problem [5, 6, 7, 8, 9]. Employed methodologies and experimental results are presented in Chapter 5.
- *Evaluation of fusion methods:* It is widely believed in the biometrics community that through the use of different modalities, or matchers, it is possible to improve the performances of the recognition systems. To accomplish the integration of multiple algorithms or modalities, information fusion principles are frequently exploited. In this thesis, we utilize several decision-level fusion algorithms and also propose new ones to boost the performance of 3D face recognition algorithms [10, 11, 2, 4]. The explanation of fusion schemes is provided in Chapter 6.

This thesis is organized as follows: Chapter 2 presents the review of state-of-the-art 3D face recognition systems. Facial registration algorithms are explained in Chapter 3. Representation and feature extraction approaches are provided in Chapter 4. Chapter 5 is devoted to the explanation of local region-based face representation schemes. The algorithms that are used in the fusion processes are provided in Chapter 6. Experimental results of the proposed approaches are presented in Chapter 6.

2. Literature Survey

The 3D shape of a human face can be represented as a non-rigid free-form surface. Currently, most of the static 3D sensors acquire data from only the visible part of the human facial surface from the viewpoint of the camera lenses. It is also possible to acquire full 3D model of a human head from multi-view stereo systems or using rotating tables during the scanning process. However, multi-view or rotating sensors are not practical for identification scenarios. Therefore, static 3D sensors such as laser scanners or structured light based stereo systems produce the so-called 2.5D surface data. 2.5D surface is usually defined as having at most one z-depth measurement from a given (x,y) coordinate. It is possible to generate full 3D face models by combining several 2.5D images. In the 3D face recognition literature, the term 3D is commonly used to denote 2.5D data. If equipped with standard 2D cameras, 3D sensors provide registered shape and texture information: for each (x,y,z) coordinate, the corresponding RGB texture information is provided.

Initial works which propose identification algorithms for 3D faces use only the facial shape information. However, current systems generally take into account texture information as well as shape. Systems which use texture and shape information together are referred to as multi-modal systems. In order to comply with the existing literature, we classify the 3D face recognition systems according to the shape representations they use. According to the shape representations utilized to represent 3D face data, algorithms can be broadly grouped as:

- *Point Cloud-based Approaches:* Human facial surface is represented by a 3D point cloud. In this category, only (x,y,z) coordinates of the sampled points from the facial surface are used.
- *Depth Image-based Approaches:* This is an appearance-based approach where the 2.5D data is projected to an image where pixel intensities denote z-depths.
- *Curve-based Approaches:* Approaches in this category extract vertical, horizontal, or contour curves from the facial surface, and represent the face using features

extracted from these curves.

- *Differential Geometry-based Approaches:* Algorithms in this category use differential geometry-based descriptors such as surface normals or curvature-based features.
- *Facial Feature-based Geometrical Approaches:* The fundamental aim of this type of approach is first to locate several facial features such as nose tip, eye corners, and mouth and then extract several features from them such as lengths, angles, or geometrical invariants.
- *Shape Descriptor-based Approaches:* Inspired by 3D free-form object representation methods, these approaches consider the facial surfaces as free-form surfaces, and try to describe them using local or global shape descriptors such as point signatures or spin images. See Figure 2.1 for the specific methods used in each of these categories.

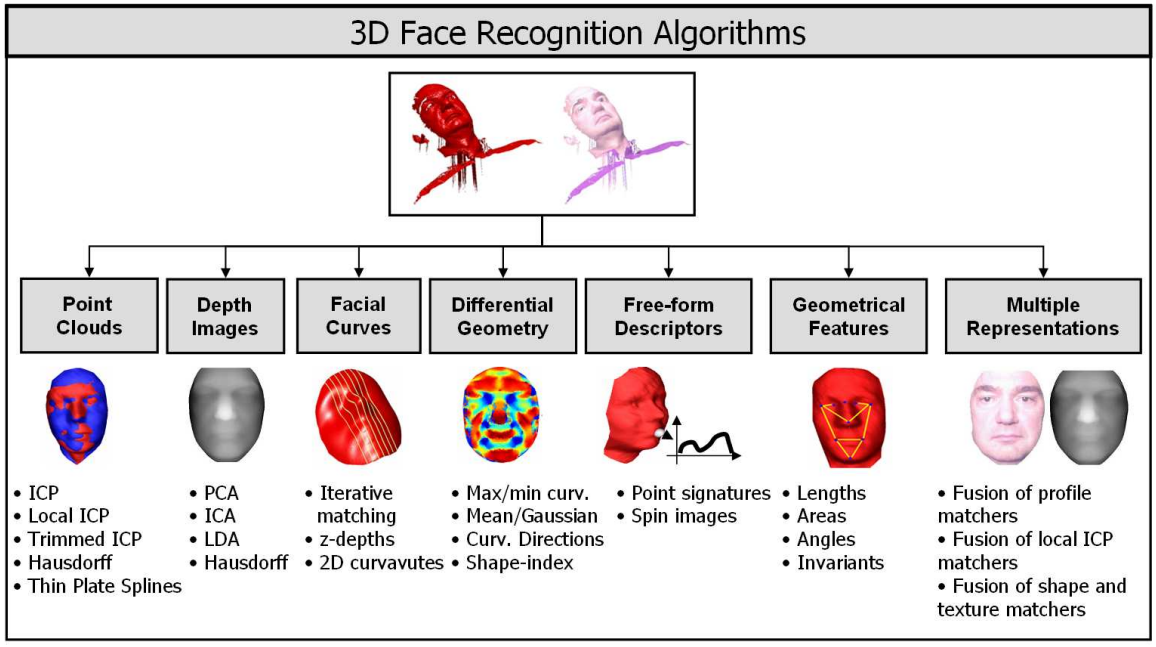


Figure 2.1. Taxonomy of 3D face recognition systems.

2.1. Point Cloud-based Approaches

Many 3D acquisition systems provide 3D point clouds as raw data, possibly coupled with 2D texture information. Thus, for many 3D face recognition systems, point cloud or point set data is the default input data representation [12, 13, 14]. The popu-

larity of the point cloud representation scheme is due to i) its generality: almost every 3D acquisition device produces (x,y,z) coordinates without any higher-level information such as connectivity, and ii) its simplicity: point coordinates, if sampled with good accuracy, are simple and sufficient to represent a complex surface. On the other hand, point cloud-based representation has several drawbacks, most notably: i) since there is no connectivity information, search for nearest points may be cumbersome, and efficient search algorithms usually require advanced data structures such as kd-trees, ii) storage requirements are high.

Generally all of the proposed approaches for 3D face recognition systems and specifically, point cloud-based algorithms require alignment and registration of facial surfaces before the matching module. The reason for the necessity of accurate alignment and registration is due to the similarity of different facial surfaces as opposed to more general 3D object classes. Thus, in contrast to the most of the 3D object identification systems, face identification algorithms put a great deal of emphasis on the correct alignment of different facial surfaces. To this end, algorithms which operate on point clouds make use of registration algorithms such as iterative closest point algorithm (ICP). Given two models, i.e., two point clouds, the ICP algorithm tries to find the best rotation and translation parameters to align one model to the other one iteratively. The limitation of of the ICP algorithm is that: it can not handle non-rigid deformations. The quality of the alignment found by the ICP algorithm can also be considered as a dissimilarity between two input models. Therefore, most of the 3D face recognition systems that use point sets utilize the ICP dissimilarity as a matching metric. The usage of the ICP-based matching algorithms is so common that it is now considered as a baseline 3D face matcher [15]. For example, the systems presented in [16, 17] use the ICP method as a core shape matching algorithm. The basic principle is to align a given test image, i.e., its point cloud data, to all of the images in the training set, and select the ID of the training image which produces the lowest ICP alignment error. Although many of the ICP-based algorithms work using this scheme, it is worthwhile to note that aligning each training image with the probe image at the identification phase is computationally very expensive.

In order to handle the computational complexity of the ICP-based matchers, several systems use template faces or average face models to align and register faces [18, 3]. Russ et al. [18] utilize a reference face to establish dense point-to-point correspondence between faces. A given face is registered to the reference face using the ICP algorithm. However, after the ICP transformation, nearest points are found using the surface normal directions which is different from using Euclidean distance. The aim of the overall scheme is to get an ordering of facial points which is necessary for the PCA. Here, PCA is directly applied to the point coordinates, as opposed to range images. Experiments are performed on two databases: FRGC v1.0 (198 subjects, 745 probe images) and FRGC v2.0 (242 subjects, 1287 probe images). In all experiments, single gallery image is used. Proposed approach attains 91 and 97 per cent rank-1 classification rates for v1.0 and v2.0 databases, respectively, which are superior to depth image-based PCA approach.

A notable extension to the standard ICP-based matching algorithms can be referred to as local ICP methods. Two important reasons for the use of local ICP methods are: 1) it is rarely the case that all of the facial surfaces exhibits rigid deformations, and 2) matching local regions is faster. A typical local ICP-based scheme is presented in [19]. Koudelka et al. [19] first locate several facial landmarks such as nose tip, sellion, inner eye corners and mouth center automatically, and then sample 150 random points in their neighborhood. The matching of two facial surfaces is then accomplished via a mixture of ICP and Hausdorff algorithms. The use of the Hausdorff measure is beneficial if there is incomplete or missing data in one the facial surfaces. The authors show the feasibility of their method on FRGC v1.0 database. Achermann and Bunke [20] also use an extension of Hausdorff distance for matching point clouds.

Another local ICP-based method is presented in [21] where Mian et al. propose the fusion of four different face classifiers. The first two classifiers use a local region based ICP algorithm where the regions are selected from nose and forehead parts. Each gallery face is segmented into three regions (nose, forehead/eyes, cheeks) by manually locating six points around nose and eye regions. During the identification phase, nose parts of the gallery faces are registered to the probe images. Initialization

of the ICP matching is performed by automatically locating nose ridge line, thus by finding the nose in the probe image. After locating the nose, forehead/eyes region-based ICP matcher is initialized automatically. In addition to ICP-based matchers, authors use two holistic PCA-based matchers that use texture and depth images. The scores obtained from four matchers are normalized by min-max rule, and fused by product rule in order to calculate the final fused score. The proposed system achieves 100 per cent verification rate at 0.0006 FAR on the FRGC v.10 database (275 subjects, 668 probe scans). Forehead/eyes region-based ICP algorithm is found to be the best matcher among the four matchers.

Wang et al. [22] use a different ICP-based scheme to deal with expression variations. During the similarity calculation at each iteration of the ICP algorithm, a fixed percentage of closest point pairs that has a large point-to-point distance are removed dynamically. The assumption is that these points correspond to the regions where deformation is present due to expression changes. The recognition experiments on ZJU-3DFED database (40 subject, 9 scans per subject) which contains expression variations show that their partial-ICP approach obtains 96.88 per cent rank-1 identification rate, whereas standard ICP approach attains 89.69 per cent accuracy. They denote the percentage of the used points as the p-rate in the partial ICP iterations. p-rates are empirically chosen to optimize performance for the given test set.

As noted previously, one shortcoming of the ICP algorithm is that it can only handle rigid transformations, and human faces generally exhibit non-rigid deformations under expression variations. Therefore, non-rigid registration algorithms could be beneficial in establishing the correspondence between facial surfaces. A representative idea is proposed in [23] where a generic face model is fitted to a given face, and the related displacement information forms a separate deformation image. Finally, the biometric signature is obtained from the wavelet analysis of this deformation image.

Another system which is designed specifically to deal with local non-rigid deformations is given in [24] where Lu and Jain propose a deformable model-based matching scheme. Using manually labeled fiducial landmarks i.e., the nose tip, eye corners, and

mouth corners, along with the mouth contour, detailed dense landmark points are sampled at the geodesic paths between several feature points. Surface deformation is then learned using a set of training images having expression variations. Then, this knowledge is used to synthesize new deformed faces from a given neutral face. These deformable models are fitted to the probe images with the help of an optimization technique in order to find the similarity. Using deformable models, it is possible to account for expression variations present in the probe images. Experimental results on two different databases (DB1: 10 subjects, 21 scans per subject, DB2: 90 subjects, 6 scans per subject, from FRGC v2.0) demonstrate that with deformation modeling, classification accuracy improves from 87.6 per cent to 92.1 per cent.

It is also possible to use other means to establish registration between two facial models apart from the ICP algorithm. However, even these type of systems use ICP at the early stages of the alignment phase. The ultimate aim is to construct point-to-point, or vertex-to-vertex correspondence. Using the correspondence information, an ordering of 3D points can be established. For instance, Papatheodorou and Rueckert [25] present a system where facial surfaces are modeled by B-splines. First, 19 landmarks are selected manually, and the central facial region is cropped. Cropped faces are registered to a template face using the ICP algorithm. Spherical B-spline-based model is then fitted to the facial surface. After fitting, 3D facial points are sampled according to an average face model. This procedure produces faces with equal number of registered points. PCA is applied to the 3D coordinates, and extracted coefficients are used as features having a dimensionality of 83. Stereo camera-based Vision RT VRT3D system is used to collect a 3D face database which contains 83 subjects. In the recognition experiments, authors report 100 per cent and 90 per cent accuracies for shape and texture channels, respectively.

Another approach to establish point correspondences is given in [26]. Hong et al. [26] present a 3D deformable model-based face recognition system. Correspondences between face vertices among all facial surfaces are established by a pixel-to-vertex map method. The pixel-to-vertex map method produces a sparse vertex representation around 5000 vertices. After finding corresponded vertices in the training set, PCA-

based synthesis algorithm is used to construct a deformable 3D facial model for both shape and texture information. Given a probe image, this deformable model is fitted to the probe image using inverse compositional image alignment algorithm. Texture and shape coefficients of the fitted model are then used as features. Recognition experiments performed on a 3D Korean face database (110 subjects, totally 218 3D face scans) demonstrate that using only texture coefficients for frontal test images, 90.4 per cent accuracy is obtained. With both shape and texture coefficients, accuracy is found to be the same. If test images contain pose variations, accuracies drop to 71.4 per cent for both texture and shape-based matchers. It is also worthwhile to note that fitting-based identification is computationally very expensive, since it is reported that identification takes 11.2 seconds on the average.

A novel registration and representation scheme which is based on point clouds is presented in [27]. Bellon et al. [27] propose a new metric, called the Surface Interpenetration Measure (SIM) to define similarity between two registered surfaces. SIM can be considered as an alternative to the commonly used Root Mean Square Error (RMSE). SIM basically quantifies the amount of interpenetration around the regions crossing over each other for the overlapping surfaces. Authors also propose a genetic algorithm (GA)-based registration algorithm and compare it with the ICP algorithm. Though no recognition experiments are reported, authors show that the use of SIM may be a better alternative to common metrics such as point differences.

A completely different idea was proposed in [28] where the distances between 3D facial points are approximated by geodesic distances. In their work, authors apply multidimensional scaling algorithm to the geodesic distance matrix to obtain a canonical face representation. In their later work [29, 30], they have extended their approach using surface gradients field. Their experimental results confirms that canonical form matching is robust to expression variations and outperforms 2D image-based eigenfaces [30].

2.2. Depth Image-based Approaches

As discussed previously, currently used 3D sensors generally produce 2.5D information. 2.5D data can easily be projected to a 2D image plane. Therefore, it is very popular to convert 2.5D facial data to a depth image, also called the range image. Each pixel in the depth image represents the distance of the corresponding 3D facial point to the camera. Although some sensors are capable of producing range images directly, point cloud data acquired from sensors is usually converted to yield depth images. During the conversion, some information may be lost. Most importantly, two sources of information loss should be mentioned: 1) In the surface areas whose normals are almost perpendicular to the camera view, such as the lower nose regions, a significant portion of the depth measurements is generally under-represented in the depth images, and 2) 8-bit standard gray-level quantization may lose accuracy information. Another important concern in depth image construction is the conversion of irregularly sampled 3D points to a regular (x,y) grid. To accomplish this task, interpolation methods are generally used.

Once the depth images are formed, one can treat the 3D face recognition problem as simply a 2D image-matching problem. Therefore, depth image solutions employ well-known subspace techniques borrowed from intensity images, such as PCA, LDA, and ICA. As in the 2D face recognition literature, PCA-based eigenface approach is considered as a baseline 3D face algorithm [31]. It is of great interest to researchers to compare which subspace methods offer best accuracies in the depth image domain. For example, in [32], Srivastava et al. compare the optimal component analysis with PCA and ICA methods, and show the superior performance of the optimal component analysis. Similarly, Hesher et al. [33] show the superiority of ICA to the PCA for depth image representation. Zhong et al. [34] show that extracting depth image features using the Discriminant Common Vectors (DCV) method is superior to PCA and LDA methods. It is also possible to generate several binary depth images by intersecting the original depth images by planes at different altitudes. Lee et al. [35] present a system where several depth image levels are used as face representation after locating the nose tip.

Depth image construction should be preceded with a pose normalization module in order to transform faces to a frontal configuration. ICP algorithm can be used for this purpose. However, once several facial features are located, a coarse alignment can be sufficient as well. In [36], Pan et al. design a pose-invariant recognition system by projecting the 3D point cloud data to a plane parallel to the face plane. The projection flattens out the facial surface, which is the point where this algorithm differs from other depth image-based techniques. Their PCA-based identification method outperforms other depth image-based approaches on the FRGC v1.0 database.

An approach for matching range images, using original measured data and not its subspace projection, is discussed in [37]. In this work, Russ et al. apply partial shape Hausdorff distance metric to range images. The motivation behind using Hausdorff distance is its partial invariance to inconsistencies such as noise, holes, and occlusions in the 3D facial data. Proposed approach enables a reduction in Hausdorff distance computation from $O(N^2)$ to $O(N)$ in range images. Their classification experiments conducted on the FRGC v1.0 database show the superiority of the proposed scheme to the standard PCA-based matching algorithm. Another system which generates local depth image features is presented in [38] where Cook et al. extract Log-Gabor features from depth and texture images. Filter coefficients are compressed by the PCA algorithm. Scores obtained from shape and texture-based matchers are fused with the sum rule.

2.3. Curve-based Approaches

Early studies for 3D face recognition emphasize the use of 2D curves extracted from facial surface such as the facial profiles. Once these curves are extracted, 2D shape analysis techniques for curves can be used for identification purpose. In this category, a seminal work is presented in [39]. In [39], central and a number of lateral profiles derived from 3D facial surfaces are used for recognition. Matching of the profiles of is carried out using Iterative Conditional Mode (ICM) optimization. Curvature values computed along the profile curves are used as features. In [40], authors extend their system where gray level information is fused with shape features.

Zhang et al. [41] also present a profile-based face matcher. Authors propose a system which automatically finds the vertical symmetry profile curve, and three points on this curve (nose bridge, nose tip, and lower nose point). After finding vertical profile curve, two horizontal profile curves that pass through forehead and cheek regions are computed. Each of these three profile curves are matched separately, and their similarity scores are fused by a weighted sum rule. Weights are determined by the LDA algorithm. It is found that the most discriminative profile curve is the symmetry profile. However, the fusion of three matchers significantly improves the identification/verificiaton accuracy. Experiments done on a 3D face database constructed via 3Q stereo system (32 subjects) illustrates 0.8 per cent EER / 96.9 per cent rank-1 identification rates for neutral-to-neutral case, and 10.8 per cent EER / 87.5 per cent rank-1 identification rates for non-neutral case.

Feng et al. [42] extracts 35 horizontal and 35 vertical facial curves from the facial surface. Facial curves are represented by integral invariants which are robust to several transformations such as translation, rotation, and scale. After describing curves using invariants, 12 curves are selected according to discriminant analysis and Jensen-Shannon divergence analysis. It is found that 10 of the selected curves are vertical and extracted from the nose and eye regions. PCA-based dimensionality reduction is applied to produce a more compact representation. Recognition experiments on a subset of UND 3D face database (35 subjects) shows that it is possible to obtain 92.57 per cent rank-1 classification accuracy using feature vectors of dimensionality 96.

2.4. Differential Geometry-based Approaches

The use of differential geometry-based surface descriptors which are invariant to typical transformations such as rotation and translation is a very common technique for face representation. The most prominent approach in this category is to use curvature-based surface descriptors. Due to their attractive characteristics, curvature-based systems are frequently used to locate facial landmarks, and to classify local surface types [43, 44]. Moreno et al. [45] segment a facial surface into seven regions using curvature and extract several features such as region areas, area relations, and

curvature means. In [46], maximum and minimum principal directions are represented by two Enhanced Gaussian Images (EGIs) and similarity between faces is computed by Fisher's spherical correlation method. Lee et al. [47] propose a curvature-based face recognition system. Proposed approach uses PCA to extract features. Classification is performed by a cascade architecture of fuzzy neural networks (CAFNN). Experiments conducted on a very small database (46 subject, 2 scans per subject) shows that CAFFN-based classification scheme is better than the k-nn method.

Abate et al. [48] generate normal maps, which store three-variate mesh normals as RGB components. The difference between the normal maps of two images is calculated in terms of three difference angle histograms. In their later work, Abate et al. [49] propose a system where each face vertex is mapped to a sphere. Sphere mesh is constructed regularly from an icosahedron by generating triangles recursively. Each spherical triangle has three neighboring triangles. Each triangle on the sphere is then represented by the difference of surface normal directions according to its neighbors. Three surface normal differences are considered as RGB values after a quantization step. As features, Fourier descriptors computed from these RGB images are used. Experimental results on a 120 subject (10 scans per subject) face database confirm that when there are significant pose variations, the proposed approach attains better identification rate than PCA and surface normal-based approaches.

2.5. Facial Feature-based Geometrical Approaches

As in the early stages of 2D face recognition systems, facial feature-based systems are also applied in 3D face recognition algorithms. Riccio and Dugelay [50] present a geometric invariant-based face recognition system. 19 control points are first manually located on the 2D intensity images. Using the mapping between 2D and 3D images, 3D locations of these landmarks are found. A number of cross ratios computed from the 2D landmark locations are used to produce a candidate class list. Among these candidate classes, the final decision is made according to the voting of several 3D geometrical invariant-based face classifiers. The proposed system relies only on the 19 control points which makes the system sensitive to their localization performance.

Recognition experiments conducted on the small database of EURECOM (50 subject, 3 samples per subject) reveals that their approach may obtain 80 per cent rank-5 identification rate when the random noise added to the control point coordinates is moderate.

Lee et al. [51] propose two feature based 3D face recognition systems. The first one uses depth coordinates of four vertical and two horizontal curves which are extracted from the central facial region. As a matcher, the dynamic programming (DP) algorithm is used. The second recognizer uses facial landmark coordinates, their distances, and angles computed from them as features. A Support Vector Machine (SVM) classifier is then used to classify faces according to these geometrical features. Identification experiments demonstrate that DP-based and SVM-based classifiers can obtain 95 per cent and 96 per cent accuracies, respectively. For the DP-based system, authors use their own face database that contains 20 subject, and for the SVM-based system, Biometrics Engineering Research Center (BERC) face database (100 subjects) is used.

2.6. Shape Descriptor-based Approaches

Inspired by the 3D free-form object recognition systems, a number of free-form object representation techniques are applied to extract either local or global surface features. *Point signatures* are among these popular 3D descriptors for face recognition. In [52], point signatures are used for both coarse registration and for rigid facial region detection which provide expression invariance. In their later work [53], authors include texture into their systems by using 2D Gabor wavelets. Another 3D shape descriptor similar to point signatures was used for face recognition in [54] where authors proposed *local shape maps* to extract 2D histograms from 3D feature points. Their approach does not require registration, and the similarity between two faces is calculated by a voting algorithm as in [52].

Xu et al. [55] fit a regular mesh to a 3D point cloud data, and then extract local shape descriptors at each vertex from the mesh. Gaussian-Hermite moments of

these shape features are then computed and used as features. Recognition experiments performed on 3DRMA (120 subjects, leave-one-out protocol) and 3DPEF (30 subjects) databases implies the advantage of the proposed approach when compared with other local shape descriptors such as point signatures and surface curvature. In their later work [56], authors try to select most discriminative features using the AdaBoost algorithm. Their original features comprise: z-depths of regular mesh vertices (545 dimensions), cosine signatures (2814 dimensions), and associative features computed from 0th, 1st, and 2nd order Gaussian-Hermite moments (2814×3). Thus, original input dimensionality is 11,801. AdaBoost algorithm is used to learn a cascade of classifiers by converting the original multi-class face recognition task into a binary classification problem. This conversion is performed by constructing inter-personal and intra-personal features. Authors perform several recognition experiments on a 3D face database that contains 123 subjects, with each subject having 37 (without glasses) or 38 (with glasses) scans with expression, pose, and illumination variation. AdaBoost algorithm is trained on a separate training set in the experiments. They report that AdaBoost-based fusion scheme is slightly better than fusing three different classifiers (classifiers that are based on: z-depth, cosine signatures, and associative features) by a weighted sum rule.

Wang et al. [57] propose a new 3D free-form representation scheme called Sphere-Spin-Images (SSI). The difference between SSI and spin images is that, shape histograms are calculated with the help of a local sphere centered on the point to be represented, as opposed to a plane passing thorough all of the object. Since it is infeasible to match two facial models using the local features extracted from all 3D points, authors select a subset of points around the center of the face where minimum principal curvature is below a certain threshold. Matching models is carried out by a voting mechanism similar to that of spin images. However, authentication experiments are performed on a very small 3D face database (SAMPL, 31 models of six subjects).

2.7. Multiple Representation-based Approaches

Up to now, we have discussed a number of 3D face recognition approaches according to their shape representations. However it is possible to combine different matchers with the aim of increased classification rate. For this purpose, a number of systems propose to fuse different shape- or texture-based individual matchers. The most typical example in this category is to design two classifiers, one for shape and the other for the texture modality, and fuse their opinions at the decision level. For instance, Tsalakanidou et al. [58, 59] propose a classic approach where shape and texture images are coded using PCA and their scores are fused at the decision level. Their experimental findings confirms that using both of the modalities is better than using shape or texture only. In their later work, Malassiotis and Strintzis [60] present a pose correction and illumination correction scheme for 3D face recognition. Proposed algorithm starts with locating the facial region by a statistical modeling of the head and torso points using a mixture of Gaussians assumption [61]. After detecting the head region, an automatic algorithm is used to locate the nose tip and the nose ridge line. Using the coordinates of these features, pose correction is carried out. After pose correction, illumination compensation is done by rendering a novel image illuminated from a frontal direction. Given normalized shape and texture images, an embedded hidden Markov model-based (EHMM) classifier produces similarity scores and these scores are fused by a weighted sum rule. Experiments carried out on two databases (each has 20 subjects) demonstrate that correction of pose and illumination increases the correct identification rates.

Similar approach which uses PCA is given in [31], Chang et al. [31] use PCA-based matchers for shape (depth image) and texture modalities. The outputs of these matchers are fused by a weighted sum rule. The experimental results obtained on a database containing 198 subjects reveal that fusing modalities achieves 97 per cent identification rate whereas individual 2D and 3D modalities have 96 and 91 per cent identification rates, respectively.

Subspace-based representations are frequently used for the fusion. BenAbdelka-

der and Griffin [62] use Local Feature Analysis (LFA) technique instead of the classical PCA to extract features from both shape and texture modalities. This classifier combines texture and shape information with the sum rule. Another interesting variant in this work is the data-level fusion. The depth image pixels are concatenated to the texture image pixels to form a single vector. Linear discriminant analysis (LDA) is then applied to the concatenated feature vectors to extract features. Authors report 100 and 98.58 per cent accuracies for the LFA-based and LDA-based fusion methods, respectively, for a face database of 185 persons. These accuracies improve the best single modality (texture) rates by 0.24 and 1.36 per cent for the LFA and LDA methods, respectively.

A prominent example of shape and texture feature fusion is presented in [63]. Wang and Chua [63] select 2D Gabor wavelet features as local descriptors for the texture modality, and use point signatures as local 3D shape descriptors. These feature-based representations are matched separately using structural Hausdorff distance, and then their similarity scores are fused at the score-level by using a weighted sum rule. The authors had previously used 3D Gabor features instead of point signatures as local shape descriptors in [64] in the same setting.

Maurer et al. [65] use the ICP algorithm to align two facial surfaces. After alignment, a difference map is produced. Each pixel in the difference map represents the distance between registered point clouds. Average pixel intensities are calculated as the final dissimilarity between facial surfaces. A texture-based matcher is adopted from a commercial product. Shape and texture scores are fused by the weighted sum rule. However, authors simply discard the results of the texture matcher if the score of shape-based classifier is very high. Verification experiments performed on the FRGC v2.0 database reveal that by fusing shape and texture matchers, verification error can be reduced by a factor of 2-2.5.

Pan and Wu [13] present a 3D face recognition system, which combines profile and surface matchers. The three profile experts use one vertical and two horizontal profile measurements. The surface expert is based on a weighted ICP-based surface

matcher. The similarity scores from these four matchers are combined by the sum rule. Obviously, their system is based on shape information only. Their recognition performance on the 3DRMA database having 120 persons show that the surface matcher which obtains 8.79 per cent error rate is better than profile matchers, and the fusion of four experts reduces the error rate to 7.93 per cent.

Another type of shape-based expert fusion is proposed in [66]. This approach is essentially a multi-region approach where different matchers responsible for the different facial regions are combined at the decision level by the product rule. Local experts compute the surface similarities of three overlapping regions around the nose by using the ICP algorithm. The distance scores produced by three ICP-based surface matchers are then combined. The local experts choose regions around the nose to obtain expression invariance. The recognition experiments conducted on the FRGC v2.0 database show that the proposed multi-region approach obtains 91.9 per cent classification rate in multiple probe experiments which is better holistic PCA (70.7 per cent) and ICP (78.1 per cent) algorithms.

Although most of the studies that use decision level fusion of different matchers that are trained on different modalities, it is also possible to combine the modalities before the decision phase. A typical example is given in [14], where shape and texture information is merged at the point cloud level thus producing 4D point features. A variant of the ICP method is then employed to determine the combined similarity of textured 3D facial shapes.

A two-level sequential combination idea was also used in [12] for 2D texture images, where the ICP-based surface matcher eliminates the unlikely classes at the first round, and at the second round, LDA analysis is performed on the texture information to finalize the identification at the second round.

An interesting algorithm that uses feature fusion via hierarchical graph matching (HGM) is presented in [67]. HGM method has the role of an elastic graph, which stores local features at its nodes, and structural information in its edges. HGM is fitted to

both the texture image and shape features, since the shape image is registered to the texture image. The scores produced from texture and shape HGM's are then fused by a weighted sum rule. Experimental results obtained on the FRGC v2.0 database show that although texture modality significantly outperforms shape modality, the integration of scores outperforms the texture modality.

Li et. al. [68] present a system which learns discriminative 2D and 3D features using AdaBoost algorithm. Local Binary Pattern (LBP) features are first extracted from 2D texture images and 3D depth images. LBP features are local descriptors and their contribution to the identification task is learned automatically by the AdaBoost algorithm. Using AdaBoost, a number of weak learners are produced from 2D and 3D modalities. Each weak learner is responsible for a local region in the images. In the first part of their experimental results, authors demonstrate that LBP-based features are superior to a PCA-based baseline algorithm. In the second part of the proposed algorithm, authors use the AdaBoost algorithm to fuse combined 2D and 3D features at the feature level. Experimental results on a 3D face database which contains 2,305 images shows that AdaBoost-based learned fusion scheme obtains better identification rate than sum rule-based fusion of PCA matchers.

Kakadiaris et al. [69] present a multimodal identification system which fuses shape, texture and Infrared (IR) imagery. 3D shape-based identification algorithm fits an annotated deformable model to a face, and computes the deformation image. The deformation image is then coded using Haar wavelets. For the thermal image modality, first a segmentation is carried out to locate skin pixels. After segmentation, a binary image indicating the presence of vessels is computed around the forehead region. This binary image thus represents the facial vasculature. The matching scores produced from shape, texture, and thermal modalities are first normalized, and then fused using product rule. 3D-shape based classifier obtains 99.3 per cent rank-1 identification rate on the FRGC v1.0 database (gallery set: 152 images, probe set 608 images). On a second database (University of Houston face database, 88 subjects, 1-5 scans per subject, totally 356 scans, with expression variations, gallery set: 62, probe set: 223), the fused system obtains 98.22 per cent rank-1 identification rate.

[70] et al. uses three nose region-based surface matchers. Their approach automatically locates nose tip, nose bridge and inner eye corners using curvature information. Once these points are located, three overlapping regions around nose are distinguished, and matched by the ICP algorithm. ICP scores are then combined by product rule. Experimental results on FRGC v2.0 database show that using only nose region is significantly better in terms of recognition accuracy even in neutral-to-neutral comparisons when compared to depth image-based PCA approach.

Table 2.1 and Table 2.2 summarize the 3D face recognition algorithms according to their representation techniques, and their coarse alignment methodologies. Many systems use several fiducial landmark coordinates as input to their coarse alignment algorithms. For each system, the details of the 3D face database used is given (column DB). Each system is analyzed according to: 1) the experimental protocol used (column P): recognition (R), or authentication (A), 2) whether they handle expression or not (column E), 3) modalities used (column M): shape (S) or texture (T), 4) whether they use fusion or not (column F), and 5) the year of publication (column Year).

Table 2.1. 3D face recognition algorithms that were published in 2005.

Ref.	Year	Representation	Database	P	E	M	F
[13]	2005	Point cloud, profiles	3DRMA	A	-	S	Y
[37]	2005	Depth image, Hausdorff	FRGC v1.0, single probe	RA	-	S	-
[32]	2005	Depth images, NN/SVM	67 subjects, FSU	R	-	S	-
[67]	2005	Depth and texture image	FRGC v2.0	A	-	ST	Y
[64]	2005	2D/3D Gabor wavelets	30×12, METRICOR-3D	R	-	ST	Y
[63]	2005	2D Gabor and point sign.	80×12, METRICOR-3D	R	Y	ST	Y
[62]	2005	Texture and depth image	185 subjects, Rainbow250	RA	-	ST	Y
[66]	2005	Point cloud, local ICP	FRGC v2.0	R	Y	S	Y
[25]	2005	B-splines, point clouds	83 subjects, VRT3D	RA	-	ST	-
[65]	2005	Point cloud, ICP	FRGC v2.0	A	Y	ST	Y
[48]	2005	Surface normal diff. map	Synth. images, 102 subjects	R	Y	S	-
[31]	2005	Depth image, PCA	FRGC v1.0, 198 subjects	RA	-	ST	Y
[50]	2005	2D/3D invariants	50×3, Geometrix	R	-	S	Y
[27]	2005	Surface interpenetration	UND, OSU SAMPL	-	-	S	-
[30]	2005	MDS-based surface rep.	30 subjects, 220 scans	RA	Y	ST	Y
[36]	2005	Depth images, PCA	FRGC v1.0	RA	-	S	-
[19]	2005	Point clouds + Local ICP, Hausdorff	FRGC v1.0	R	-	S	-
[23]	2005	Point cloud, deformation image	FRGC v2.0	A	Y	S	-
[68]	2005	Depth and texture im., LBP features	2305 scans, Minolta	R	-	ST	Y
[51]	2005	Facial curves, SVM	100 subjects, BERC db.	R	-	S	-
[69]	2005	Shape deformation, texture, IR image	FRGC v1.0, University of Houston DB, 356 scans	RA	-	STI	Y
[38]	2005	Log-Gabor features	UND database	A	-	ST	Y
[60]	2005	Depth and texture, EHMM classifier	Two databases (20 subjects)	R	-	ST	Y

Table 2.2. 3D face recognition algorithms that were published in 2006.

Ref.	Year	Representation	Database	P	E	M	F
[18]	2006	Point clouds + ICP	FRGC v1.0 and v2.0	RA	Y	S	N
[34]	2006	Depth Images + DCV	123 × 10, CASIA	R	Y	S	-
[42]	2006	Facial curves + Integral invariants	UND (35 subjects)	R	-	S	-
[21]	2006	Segmented ICP, PCA(2D/3D)	FRGC v1.0	RA	-	ST	Y
[22]	2006	Point clouds + ICP	40 × 9, ZJU-3DFED, Inspeck Mega Capturor	R	Y	S	-
[41]	2006	Facial profiles	32 subjects, 3Q stereo system	RA	Y	S	Y
[12]	2006	Point cloud	100 subjects MSU DB + 100 subjects USF DB	RA	-	ST	-
[24]	2006	Point cloud + deformable model	DB1: 10 × 21, DB2 (from FRGC v2.0): 90 × 6	R	Y	S	-
[26]	2006	Deformable model (shape and texture)	110 subjects, Geometrix FaceVision.	R	-	ST	-
[56]	2006	Regular mesh, cosine signatures, G-H moments	110×38 Minolta VIVID 910	R	Y	S	Y
[70]	2006	Local ICP	FRGC v2.0	RA	Y	S	Y
[47]	2006	Depth images, curvatures, PCA	46 × 2	R	-	S	-
[49]	2006	Fourier descriptors.	120 × 10	R	-	S	-
[17]	2006	Point Cloud + ICP	50 3D full faces, 400 2.5D faces, (8 scans per subject)	A	Y	S	-

3. 3D Face Preprocessing, Alignment and Registration

The 3D face recognition problem can be considered as a special case of a more general 3D object recognition problem. An important distinction of human faces when compared to other 3D objects is that they all share common attributes and are similar to each other. In 3D object retrieval, one aims to categorize 3D objects that have large dissimilarities, such as different animals, archeological objects, or items of furniture. As opposed to this, face recognition presents a different challenge: All surfaces are similar in that there is a nose, two eyes, a mouth, and a chin. It is needed to characterize more subtle differences in order to differentiate between human faces. Due to this important distinction, the registration problem becomes a crucial step in 3D face recognition systems. Although there are methods which do not need any registration in the object recognition systems, the similarity of human face shapes makes it a necessity to first align and register facial surfaces before proceeding to feature extraction and recognition steps.

In this chapter, we present our registration methods in detail. Registration can be considered as a two step procedure: 1) alignment and 2) dense point-to-point correspondence establishment. Our approach is based on a generic face template. Most of the previous studies register the probe image to all of the gallery images in the gallery set. However, this approach may become infeasible in actual systems since the registration process is computationally very intensive. Therefore, we first construct an average face model and register all of the training images prior to the identification phase. At the identification phase, registering the client image with the average face model is sufficient to establish correspondence with gallery images. We also propose two different registration schemes using this methodology. The first one uses iterative closest point approach to find optimal translation and rotation parameters, and the second one additionally employs warping in order to handle non-rigid local deformations.

In this chapter, we first explain preprocessing modules which are responsible for noise removal, smoothing, and hole filling. Then, we provide our alignment and

registration algorithms starting with average face model construction algorithm. Comparative analysis of registration methods will be given in Chapter 7.

3.1. Noise removal, Smoothing, and Cropping

3D acquisition devices generally output raw 3D point measurements that are sampled from the surface of the scanned object. According to the type of the 3D sensor, these point measurements may contain noise, or, in some cases, the device may not be able to obtain a measurement. Currently available 3D face sensors generally produce small to large surface protrusions, and may not measure 3D points on the areas that have low reflectance, such as eyebrows. In order to deal with noisy protrusions over the facial surface, a two phase filtering methodology is applied. In the first phase, median filtering is applied to remove impulse-like protrusions. Input 3D data is provided in a 2D matrix format where each entry in the matrix contains the z-depth measurement of a point. Each row and column of the matrix corresponds to specific x and y coordinates, respectively. In this form, input data can be considered as a range image. This input data representation makes it easy to apply 2D filters. This representation makes it also easy to interpolate the depth coordinates of missing points to some extent. We apply median filtering in order to fill small holes: If a point in the 2D matrix does not have z-depth measurement, we interpolate it by using the median value of its neighbors' z-depth values. The neighborhood is determined by the size of the median filter mask. In summary, the first preprocessing phase eliminates the large protrusions and fills small holes with the aid of median filtering. In the second phase, we apply mean filtering in order to smooth the facial surface. Figure 3.1 shows sample outputs of the median and mean filtering operations.

In a typical scanning scenario, the acquired face data contains non-face regions such as shoulders, neck, or background clutter. It is therefore needed to isolate the central facial region. For this purpose, a 3D face detector can be used. However, since our main concern is not to detect the facial region, a simple method is used to crop the central facial region with the help of manually labeled nose tip coordinates. Cropping is performed by discarding any 3D point outside a spherical volume which is centered

Figure 3.1. Original noisy face, median filtered version, smoother version.

on the nose tip. Figure 3.2 shows a sample cropped face.

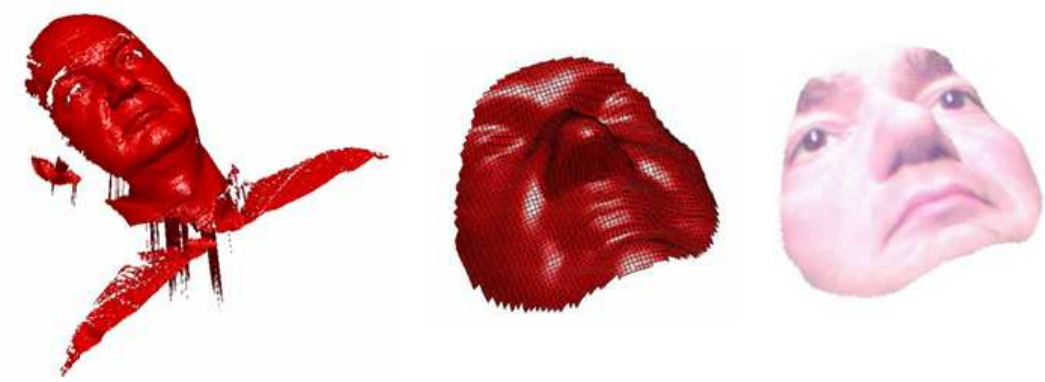


Figure 3.2. Original face, cropped shape and texture images.

3.2. Alignment

Careful alignment and registration of facial surfaces are crucial to the performance of a 3D face recognizer. In this thesis, alignment refers to the transformation of a given facial surface to a common coordinate system such that one can define similarity between any two facial surfaces. Our approach is based on aligning each facial surface to a common face model. It is possible to compare two facial surfaces which are aligned to a common face model. Most of the previous approaches [16, 19, 21, 22, 66] align a given probe face to each gallery image directly and compute the similarities. However this approach is computationally expensive since it performs N registration operations if there are N gallery images in the training set. If all of the gallery images are previously registered to a common face, then it is sufficient to register the given probe image to the common face once. Therefore, using only *one* registration operation at the

identification phase, it is possible to compare the probe image to all of the previously registered gallery images. Most of the current systems now start to employ a similar scheme of using a generic face for registration [18, 23].

We will define an average face model (AFM) and construct it off-line from a given set of training faces. In order to construct the AFM, several manually located facial landmark coordinates of the training faces are needed. We have labeled seven facial landmarks (inner and outer eye corners, nose tip, and mouth corners) on each training face. Given a set of training images and their landmark coordinates, the AFM can be constructed as follows:

1. *Computation of average landmark locations:* If all of the training faces were perfectly frontal and had a same scale with a gaze direction parallel to the z-axis, then it would be enough to employ a simple averaging to compute the average landmark locations. However, in practice, some of the facial images may have slight rotation and scale variations, which may lead to incorrect average landmark coordinates. Therefore, it is useful to first transform faces into a canonical position. In order to compute average landmarks, we employ a two phase procedure. In the first phase, we average the individual landmark coordinates in order to compute a rough estimation of the final average landmarks. This phase assumes that majority of faces have the same scale and orientation parameters. In the second phase, each set of landmark coordinates in the training set is transformed to the estimated average locations. This accomplished by Procrustes analysis [71]. Procrustes analysis finds the best translation, rotation and scale parameters in order to transform one set of measurements to another set of measurements in the least squares sense. Once all of the transformed landmark coordinates are found, the final average landmark locations are found by averaging them.
2. *Surface fitting and average facial surface computation:* Once the locations of the average face landmarks are found as explained in the previous stage, each training face is transformed to these landmarks with the help of Procrustes analysis. After this transformation, faces can be considered as fully frontal with the same scale, with a gaze direction parallel to the z-axis. After transformation, faces are

resampled at regular (x,y) intervals by using linear interpolation. After resampling, the average facial surface can be computed by averaging the z-depth values of the training faces at regular (x,y) positions. Figure 3.3 shows the average face model together with its landmark positions.

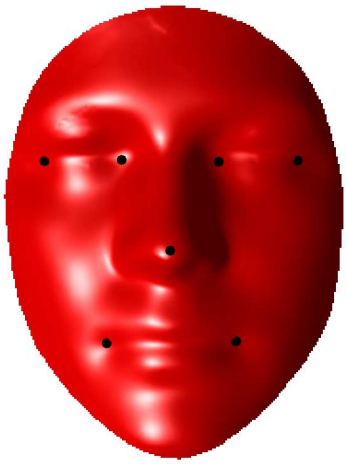


Figure 3.3. AFM and its seven landmarks.

Once the AFM is computed, we can proceed to the alignment procedure. The alignment procedure consists of two phases: coarse alignment and fine alignment. Coarse alignment is needed for the correct convergence of the fine alignment phase. Our aim in the alignment step is to align a given face image to the AFM by finding the transformation parameters such as translation and rotation matrices. If the locations of several facial landmarks of a given face is known, Procrustes analysis can be used to align the face to the AFM. If only the nose tip coordinate is known beforehand, then simply translating the face to the AFM model such that nose tips coincide may suffice. Either method can be used in the coarse alignment phase. After coarse alignment which operates on landmark coordinates only, a more detailed alignment step is necessary. In this fine alignment step, all of the 3D points over the facial surface is taken into account to better align the face to the AFM. For this purpose, Iterative Closest Point (ICP) method is used. ICP algorithm iteratively finds the best rotation and translation parameters to align a given face to the AFM [72].

3.3. Dense Correspondence Establishment

At each iteration, the ICP algorithm finds pointwise correspondences between the given face and the AFM. Correspondences are established by searching the nearest point in the given face to a point in the AFM. Once the ICP algorithm converges, the final correspondences established for each AFM point can be considered as the final dense point-to-point correspondence between a face and the AFM. Thus, ICP algorithm essentially finds a mapping for each AFM point to its corresponding nearest point in the given face. Given any two faces, their correspondence can be found via their correspondences with the AFM. For instance, let p^i_{AFM} be the i^{th} point in the AFM, and $p^i_{F_1}$ and $p^i_{F_2}$ be the corresponding points (i.e., nearest points) to the p^i_{AFM} in faces F_1 and F_2, respectively. Then, one can say that points $p^i_{F_1}$ and $p^i_{F_2}$ are the corresponding points. Using this method, if there are M points in the AFM, then a dense point-to-point correspondence between subsets of M points in any two faces can be established.

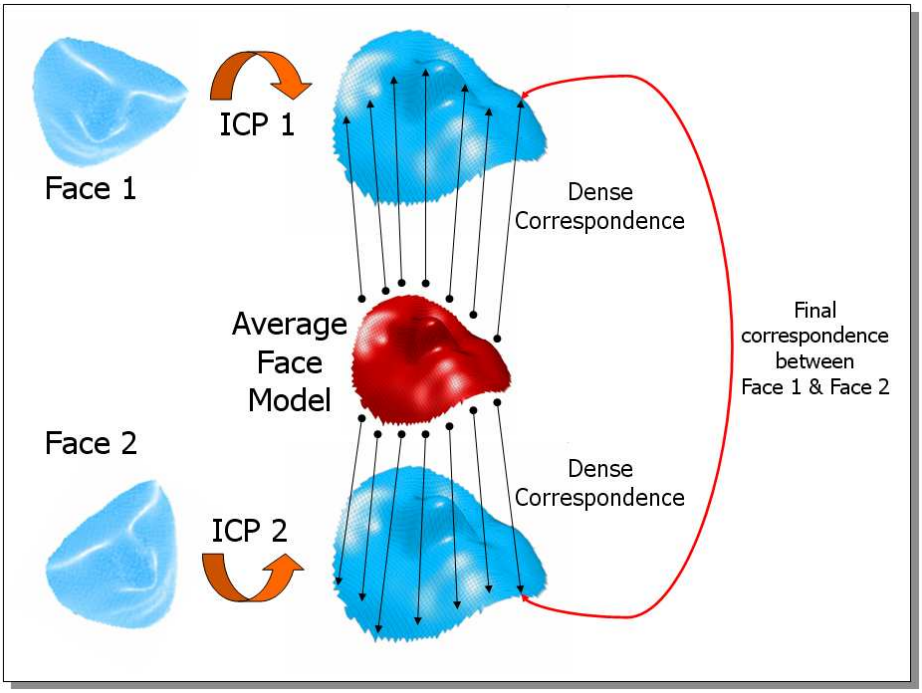
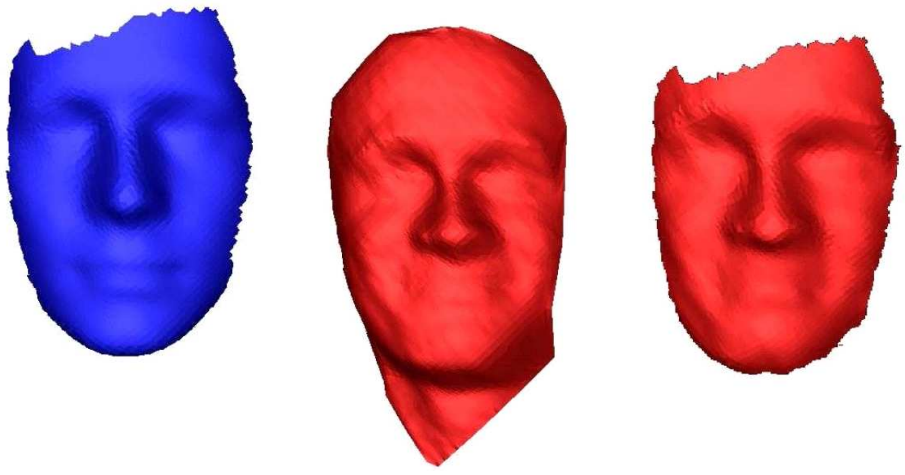


Figure 3.4. AFM-based ICP registration.

ICP-based registration handles only rigid transformations. However, human faces may exhibit local non-rigid deformations especially around the mouth region. There-

fore, if such variations are present in the input face, the use of the ICP-based registration method may obtain sub-optimal registration performance. In order to deal with such problems, warping-based registration can be utilized. This warping-based alternative registration method may deform a facial surface to find dense pointwise correspondences between faces. For this purpose, Thin Plate Spline (TPS) warping method can be used. TPS non-linearly warps facial surfaces such that their facial landmarks coincide exactly, and the rest of the surface points are transformed as a thin plate. In order to apply TPS-based registration, several facial landmarks should be known beforehand. The general outline of the TPS-based registration algorithm can be stated as follows:

1. *Compute warping parameters:* Given the AFM and a facial image together with their landmark locations, find the warping parameters using the TPS method. Finding warping parameters depends only on the landmark coordinates.
2. *Warp the face to the AFM:* Once the warping parameters are known, apply the warping function to all of the points in the given face to warp it to the AFM.
3. *Dense point-to-point correspondence establishment:* Given the warped input face and the AFM, find the nearest points in the input face to every point in the AFM. Figure 3.5 shows the effect of TPS warping on a sample face.



(a) (b) (c)

Figure 3.5. (a) *AFM*, (b) the original face, and (c) the warped/cropped version of the original face.

4. Representation and Feature Extraction

The 3D face data obtained from the sensors originally contains only 3D point measurements. Representing facial data using 3D points is just an example among various possible representation schemes. In this chapter, we present different representation schemes including point clouds, surface normals, surface curvatures, facial profiles, 3D voxels, depth images, and 2D intensity images.

According to the representation schemes used, several feature extraction methods can be applied to construct feature spaces. For example, statistical dimensionality reduction techniques such as PCA or LDA can be applied to depth images, or Gabor wavelets can be used to compute feature coefficients from 2D intensity images. Additionally, (x,y,z) coordinates can also be viewed as features for point cloud representation, or transformations can be applied to extract more compact and descriptive features.

Therefore, face patterns that will be recognized are constructed using a two-level scheme: 1) by choosing the representation method, and 2) by choosing the feature extraction method. As stated previously, chosen feature extraction method depends on the representation technique used.

The use of point clouds [66, 12, 70], facial profiles [39, 40, 41], or depth images [31, 33, 34] as face representation methods are very common in 3D face recognition systems. Prior to our work, surface normals were not used to describe registered faces. Recently, their use is also presented in [48, 49]. Similarly, surface curvatures are heavily used in segmentation and facial feature localization, but their usage for the representation and identification of human faces is very rare [46]. Here, we attempt to define similarities between faces using their mean, Gaussian curvatures and their principal directions. The voxel-based volumetric face representation scheme is also novel, and first studied in [73]. Together with the use of different feature extraction techniques, we form a large number of 3D face recognizers. Therefore, a thorough comparative analysis of various

3D face recognition algorithms is made. In addition, we also use them as individual face experts in a fusion setting.

4.1. Point Cloud Representation

Let Λ_i be a 3D face of the i^{th} individual. In the point cloud representation method, Λ_i is represented by the set of 3D point coordinates: $\Lambda_i^P = \{p_1^i, p_2^i, ...p_m^i\}$, where p^is are the (x, y, z) coordinates of each 3D point and m is the number of points in the face (See Figure 4.1 for sample point clouds). After the registration phase, we know that each 3D point has a corresponding point. So, for example, if the k^{th} point p_k^i is the nose tip in face Λ_i, then the point p_k^j in face Λ_j is also the nose tip.

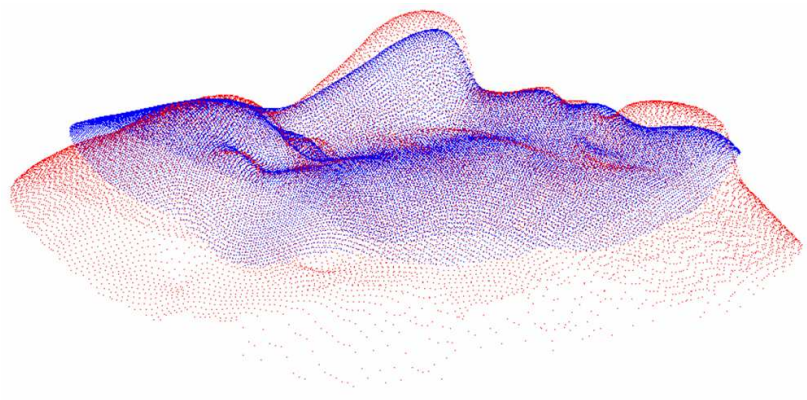


Figure 4.1. Two point cloud samples.

4.1.1. (x,y,z) Coordinate Features

The natural feature that can be extracted from point cloud representation is the set of (x,y,z) coordinates of the ordered 3D points. We define the distance between two faces Λ_i and Λ_j as: $D(\Lambda_i^P, \Lambda_j^P) = \sum_{k=1}^{m} ||p_k^i - p_k^j||$, where $||.||$ denotes Euclidean norm. This distance function can be viewed as a discrete approximation of the volumetric difference between two facial surfaces.

4.1.2. ICA of Point Clouds

Independent Component Analysis (ICA) is a powerful unsupervised statistical method which is frequently used in 2D face recognition systems [74]. ICA features have been tested for 3D face recognition in [33, 32, 73, 75]. The sensitivity of ICA to the high-order relationships among pixels makes it an attractive alternative to other methods such as PCA. There are two ICA variants: *ICA Architecture I* and *ICA Architecture II* [74]. In our work, we use the second architecture, and refer to it as ICA. Basically, given a training image set X, where each column in X represents different faces, ICA computes the matrices W^{-1} and U such that $X = W^{-1}U$. Here, columns of W^{-1} are the basis images (also called the *mixing matrix*), and columns of U are the coefficients of the basis images in W^{-1}. ICA attempts to make the outputs, U, as independent as possible. In [4], FastICA method [76] has been used to compute the W^{-1} matrix. Once the basis images (W^{-1}) are computed, the representational code for test images is obtained by: $U_{test} = WX_{test}$. The columns of X_{test} and U_{test} contain test images and found ICA coefficients, respectively. In practice, instead of applying ICA directly on the high-dimensional image features such as pixels, PCA analysis is first performed to reduce the dimensionality the columns in matrix X.

For the point cloud representation, all (x,y,z) coordinates of a face are concatenated to a single vector. Its dimensionality is then reduced by applying PCA to the training set of point-cloud vectors. Each face is then represented by the first K PCA coefficients. The columns of the data matrix X for the ICA analysis are constituted of PCA coefficient vectors. Then, the FastICA algorithm described by [76] is applied to obtain the basis and the independent coefficients.

4.1.3. NMF of Point Clouds

Nonnegative Matrix Factorization [77] is a matrix factorization technique which factorizes a given data matrix V into two matrices W and H such that every coefficient in matrices W and H are non-negative. Formally, given an $n \times m$ data matrix V that contains n-dimensional data vectors in m columns, the NMF technique produces W

and H matrices such that $V = WH$, where W is an $n \times r$ *basis* matrix, and H is an $r \times m$ *coefficient* matrix. Non-negativity constraint on W and H matrices leads to a part-based representation of the input images.

W and H matrices are computed iteratively and in [4], we use the multiplicative update rules presented in [77]. Parallel to the preprocessing stage of ICA decomposition, we first apply PCA to reduce the dimensionality of the raw data (point cloud coordinates) and place the first M PCA coefficients of each face into the columns of the data matrix. We add a constant to the PCA coefficients to obtain a nonnegative data matrix.

4.2. Surface Normal Representation

Surface normals are features inspired by differential geometry of surfaces and they actually encode the rate of change of the surface over local patches. Surface normals can be used as 3D features. For each 3D point on the facial surface, surface normals are computed with the help of Delaunay triangulation. For each triangular polygon, we compute the polygon's surface normal using its corner points. For each vertex in the triangulated face data, we can compute the surface normal by averaging the neighboring polygon's surface normals.

4.2.1. Raw Surface Normal Features

At each 3D point on the facial surface, we encode the points using their unit surface normal vectors: $\Lambda_i^N = \{n_1^i, n_2^i, ..., n_m^i\}$ where n_k^is are 3D unit normals: $n_k^i = \{n_x, n_y, n_z\}$. The distance between two registered facial surfaces is then described by: $D(\Lambda_i^N, \Lambda_j^N) = \sum_{k=1}^{m} ||n_k^i - n_k^j||$. Figure 4.2 shows all of the surface normals calculated for a given facial surface.

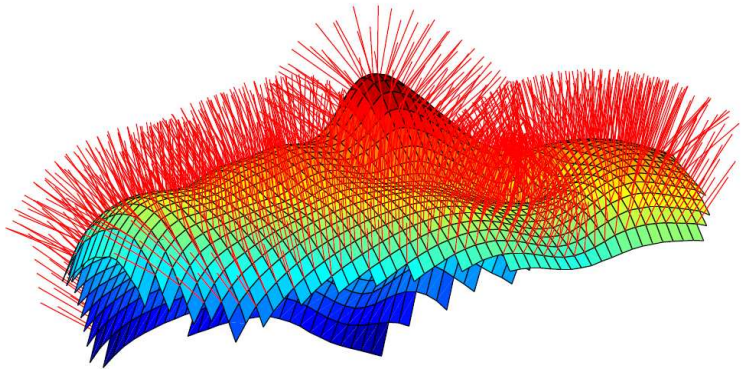


Figure 4.2. Surface normals of a sample face surface.

4.2.2. LDA of Surface Normals

Once the surface normals of all of the registered face vertices are computed, concatenation of them produces a single feature vector. Linear discriminant analysis can be applied to these feature vectors to obtain a discriminative feature subspace. Then LDA coefficients in the reduced subspace can be used as features. In order to deal with the large dimensionality of the input feature vectors, PCA is first applied, and then LDA is used to extract final coefficients. The distance between faces is calculated by the Euclidean norm.

4.3. Facial Profile Set Representation

Facial profiles are defined as 2D curves extracted from the facial surface [39, 40, 41]. Figure 4.3 shows seven vertical profiles of a sample face. We locate the central profile of the *AFM* using the nose region, and use dense registration information to locate central profiles for every test face. The direction of the central profile on the AFM is found by the principal directions of the (x, y) coordinates of the points over the nose region (See Figure 4.4.a). Once the central profile is found, it is straightforward to locate left/right lateral profiles. The algorithm to locate central and six lateral profile curves (three left and three right) is presented in Figure 4.5.

Before matching profile curves, we need a registration between them. Registration

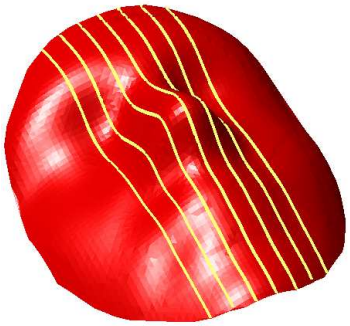
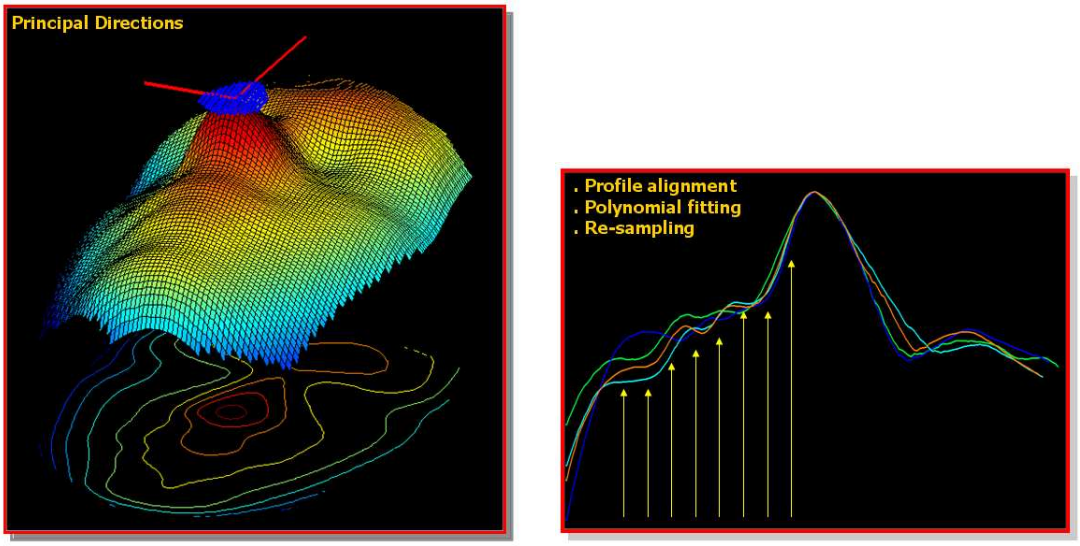


Figure 4.3. Seven equally spaced vertical profiles.

of profile contours is performed by translating profile curves using the nose tip coordinate. A spline is fitted to the profile curve, and it is regularly sampled in order to be able to compute Euclidean distances between two profiles. Figure 4.4.b illustrates the alignment and regular sampling operations. In this work, we use seven equally spaced vertical profiles. The distance between faces Λ_i and Λ_j is defined as the sum of the distances between each corresponding profile curve.



(a) (b)

Figure 4.4. (a) Finding central profile for the AFM, (b) Aligning profile curves

4.4. Curvature-based Representation

Curvature of a surface in 3D measures the amount of local bending. Curvature-related descriptors are attractive since they are invariant to rotations, and therefore,

ALGORITHM: Facial Profile Curve Set Extraction

FUNCTION: Locates central and lateral profile curves on the AFM

INPUT:

Average face model: a set of points $\Omega = \{p_1, \ldots, p_m\}$,

Nose tip location: $n = \{n_x, n_y, n_z\}$

Lateral profile spacing: l

OUTPUT:

Central and three left and right lateral profile curves: $C_1, \ldots, C_7$

where each $C_i = \{c_1, \ldots, c_k\}$ contains 3D points

1 FINDPROFILECURVES(Ω,n,l)

2 Find topmost nose points, $\Omega_n \in \Omega$, whose z-depths are in the range: $[n_z, n_z - t]$, where t is depth range value.

3 Apply PCA to the (x,y) coordinates of points in Ω_n.
Compute the maximum (d_{max}) and minimum (d_{min}) principal directions.

4 Select a subset of points, C_1, whose projections to the xy-plane are nearest to the d_{max}. C_1 is the central profile curve.

5 Obtain the xy- projections of three left and three right lateral profiles curves using l and d_{max}. Repeat Step.4 for all lateral profile curves.

6 RETURN: $C_1, \ldots, C_7$.

Figure 4.5. Pseudocode of the profile set finding algorithm.

they are frequently used in segmenting 3D surfaces [78]. There are different forms of curvature-based descriptors such as minimum/maximum curvatures, their principal directions, mean/Gaussian curvatures, and shape-index values. These descriptors can be used to represent facial surfaces, and are suitable as discriminative features.

4.4.1. Principal Directions

Given a point on a surface, there are many curves passing through that point, and each of them has a curvature. Among these curves, two extremal curves have a

special importance: the one that has the minimum curvature, and the one that has the maximum curvature. Therefore, each point on a surface can be characterized by its minimum (κ_1) and maximum curvature (κ_2) values, and their directions. These directions are called principal directions (ρ_1, ρ_2), and are expressed as vectors in 3-space. Given two registered facial surfaces, we compute the distance using minimum principal directions as: $D(\Lambda_i^\rho, \Lambda_j^\rho) = \sum_{k=1}^{m} ||\rho_k^i - \rho_k^j||$ where ρ may represent either the minimum principal direction or the maximum principal direction. The final distance is computed by the summation of these two distances.

4.4.2. Mean/Gaussian Curvatures

Mean and Gaussian curvature values are commonly used surface descriptors in the computer vision community and they are related to the minimum and maximum curvatures [43]. Let κ_1 and κ_2 be the maximum and minimum curvatures, respectively. Then mean (H) and Gaussian (K) curvatures are defined as:

$$H = \frac{1}{2}(\kappa_1 + \kappa_2) \tag{4.1}$$

$$K = \kappa_1 \kappa_2 \tag{4.2}$$

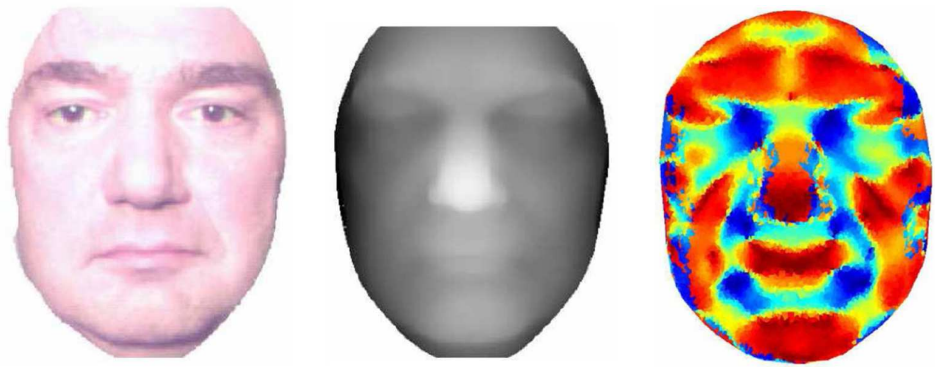
The distance between any two registered facial surfaces can be computed as: $D(\Lambda_i^H, \Lambda_j^H) = \sum_{k=1}^{m} ||H_k^i - H_k^j||$ for the mean curvature. Similarly, Gaussian curvature-based distance can also ce computed.

4.4.3. Shape-index

A popular method to characterize the surface patches is the shape index. Shape index is calculated by:

$$S = \frac{1}{2} - \frac{1}{\pi} \arctan(\frac{\kappa_1 + \kappa_2}{\kappa_1 - \kappa_2}) \tag{4.3}$$

where $\kappa_1 > \kappa_2$. κ_1 and κ_2 are the *principal curvatures*. S has a range of $[0, 1]$. Note that for planar surfaces where $\kappa_1 = \kappa_2$, S is undefined. Figure 4.6.c illustrates the shape-index values of a sample face. 3D faces can be represented by the shape index values at each vertex. For instance, a 3D face image can be represented by: $\Lambda_i^S = \{s_1^i, s_2^i, ..., s_m^i\}$ where s^is are shape index values. The distance between two faces in the shape-index representation is calculated by $D(\Lambda_i^S, \Lambda_j^S) = \sum_{k=1}^{m} ||s_k^i - s_k^j||$. In the distance computation, the vertices having undefined s_j^i's are simply discarded.



(a) Texture (b) Depth (c) Shape-index

Figure 4.6. Sample (a) texture, (b) depth and (c) shape-index images.

4.5. Depth Images

The facial data provided by 3D acquisition devices usually covers the surface which is directly visible by the camera of the sensor. Due to this principle of operation, facial data can be said to have 2.5D property: each (x,y) coordinate pair has at most one z-depth measurement. This means that we usually have the depth data of the visible region from the camera's point of view. Therefore, it is natural to project the 2.5D data to an arbitrary image plane without significant loss of information. This procedure produces the so called range image or the depth image where the depth data is formatted similar to 2D intensity images. The only difference is that, in depth images, pixel intensities denote the z-depth of the object scanned. Figure 4.6.b shows a sample depth image formed in this way. In practice, z-depth measurements may not be available for all of the pixels in the depth image. In this case, linear interpolation may be used to obtain measurements at these regions. Once the 2.5D point clouds are converted to 2D depth images, many of the feature extraction techniques such as DCT,

DFT, and ICA can be applied to range images.

4.5.1. DFT/DCT of Depth Images

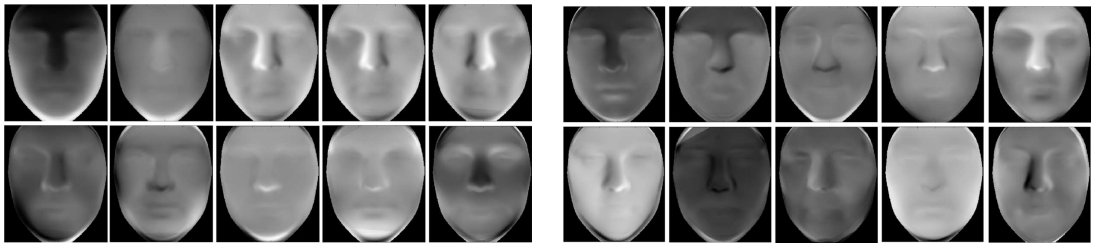
Discrete Fourier Transform (DFT) and Discrete Cosine Transform (DCT) are data independent methods which are frequently used in data representation and classification studies. Low-frequency coefficients extracted from DFT and DCT algorithms are especially important in representing highly-correlated data. Given a a depth image $I(x, y)$, we calculate its DFT and extract low-frequency coefficients to form a feature vector by concatenating the real and imaginary parts of the coefficients. Likewise, we compute the global DCT and obtain a feature vector of real DCT coefficients. Details on DFT/DCT-based face representation scheme can be found in [73, 75].

4.5.2. ICA of Depth Images

The ICA analysis for depth images follows a similar procedure as in the application of ICA to point cloud representation. The columns of a depth image are concatenated to form a single one-dimensional vector, one for each face. This data is subjected to PCA reduction and ICA decomposition. Figure 4.7.a shows the first 10 basis functions derived from principal component analysis, whereas Figure 4.7.b shows 10 independent face components. PCA only captures the second order variations due to the general face geometry, while ICA faces represent individual faces within the database fairly well. One can observe more face-like structures from the ICA basis images.

4.6. 3D Voxel Representation

Point cloud data can be converted to a voxel representation by defining a 3D grid of size $N \times N \times N$. The grid is positioned such that the center cell coincides with the center of the point cloud data. Voxel function $V(x, y, z)$ takes the value of 1 if there is at least one 3D point in the corresponding cell, otherwise it is set to 0. For facial point cloud data, binary voxel function attains 1 at the surface region.



(a) PCA bases (b) ICA bases

Figure 4.7. (a) First 10 basis faces obtained from PCA applied on depth images, and (b) Basis faces from ICA of depth images (Taken from [4]).

Given a binary voxel structure defined by the function $V(x, y, z)$, its continuous version can be obtained by the distance transformation. Based on the Manhattan distance between voxels, 3D distance transformation is applied to propagate the information present in the facial surface to the neighboring cells. Distance function is zero at the surface, and increases as we move away from the surface. It is possible to visualize the obtained continuous 3D voxel function $V_d(x, y, z)$ through slices, as depicted in Figure 4.8.

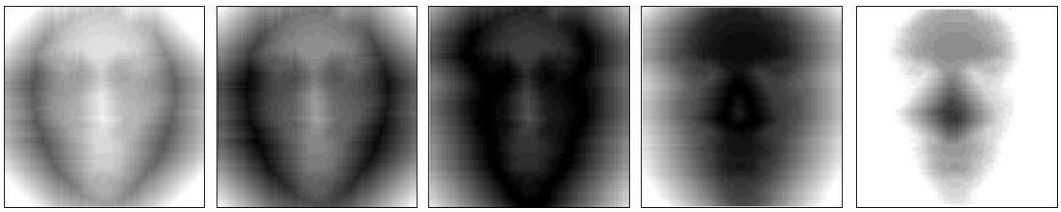


Figure 4.8. Slices from the voxel representation based on the distance transform (Taken from [4]).

4.6.1. DFT of Voxel

3D DFT is applied to the continuous voxel function $V_d(x, y, z)$ which was obtained by the distance transform. The low-pass real and imaginary DFT coefficients are used as features. Details of the DFT-based representation of voxels can be found in [73, 75].

4.7. 2D Intensity Images

For each 3D face, we have its 2D color texture information, which is also densely registered to its shape image. In other words, for each 3D point we have the corresponding RGB intensity values. We make use solely of the gray-level information in the algorithms. Therefore, we first convert the color image to gray-scale, and then apply histogram equalization to remove some of the global illumination effects.

4.7.1. Pixel Features

It is well-known in the face recognition community that if faces do not contain significant illumination differences, it is possible to use directly the pixel information as features. However, in this case, dimensionality reduction should be carried out to decrease the dimensionality. This approach provides a baseline recognition performance.

4.7.2. PCA of 2D Intensity Images

Eigenface method [79] is one of the most popular 2D intensity-based feature extraction algorithm in the face recognition community. Therefore, we choose to extract and use PCA coefficients for intensity images.

4.7.3. 2D Gabor Wavelet Features

A biologically motivated representation of face images is to code them using convolutions with 2D Gabor-like filters. In order to represent face images using Gabor filters, we have placed a square grid over the face region in the image. Since the intensity images are aligned in the registration phase, we do not need to employ a time-consuming facial feature localization algorithm. At each grid point on the image we have convolved the image with Gabor kernels. The set of convolution coefficients for kernels of different orientations and frequencies at one image pixel is called a *jet* [80]. A *jet* contains responses of convolutions in an image, $I(\vec{x})$ around a given pixel $\vec{x} = (x, y)$. It is based on a wavelet transform, defined as a convolution with a family

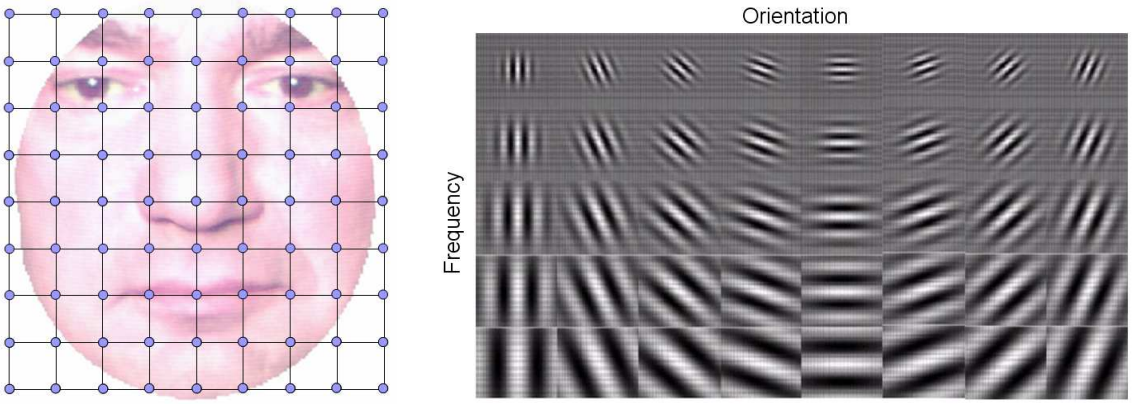
of Gabor kernels

$$\psi_j(\vec{x}) = \frac{k_j^2}{\sigma^2} e^{-\frac{k_j^2 x^2}{2\sigma^2}} [e^{i\vec{k_j}\vec{x}} - e^{-\frac{\sigma^2}{2}}] \tag{4.4}$$

in the shape of plane waves with wave vector $\vec{k_j}$, restricted by a Gaussian envelope function. We employ a discrete set of 5 different frequencies, with $v = 0, \ldots, 4$, and 8 orientations, with $w = 0, \ldots, 7$,

$$\begin{pmatrix} k_{jx} \\ k_{jy} \end{pmatrix} = \begin{pmatrix} k_v \cos\varphi_\mu \\ k_v \sin\varphi_\mu \end{pmatrix}, k_v = 2^{-\frac{v+2}{2}}\pi, \varphi_\mu = \mu\frac{\pi}{8}, \tag{4.5}$$

with index $j = \mu + 8v$. The width σ/k of the Gaussian is controlled by the parameter $\sigma = 2\pi$. Therefore, in Gabor-based representation scheme we obtain a feature vector of dimensionality $M \times N \times 40$ where $M \times N$ is the rectangular lattice resolution.



(a) (b)

Figure 4.9. (a) Rectangular grid points, and (b) 2D Gabor kernels for five different frequencies and eight different orientations.

5. Selection of Local Features/Descriptors for Face Recognition

Face recognition algorithms can be categorized according to their use of local or global features. Local and global approaches are distinguished by the size of the region of interest. To clarify this statement, consider the PCA approach. Although PCA-based methods are frequently cited as global (holistic) approaches, it is possible to use PCA to design a local feature-based algorithm by applying PCA to local regions of interest, i.e, by simply dividing the whole facial image into rectangular patches. Therefore, the size of the region that is used to extract features determines whether the algorithm should be considered as global or local.

At the early stages of the face recognition research, global approaches were preferred since they usually provide more compact features. Later on, when the researchers tried to overcome the problems caused by pose, expression, and illumination changes, local approaches received more attention. A prominent example can be given by the frequent use of 2D Gabor wavelets for facial feature extraction. Usually, 2D Gabor wavelet features are extracted from salient facial points. Alternatively, several systems use full pixelwise convolutions. However, it is crucial to determine which facial regions provide most discriminative information in order to: i) increase the identification accuracy, ii) reduce the computational load of the feature extraction phase, and iii) reduce the sensitivity of the system to various variations such as facial expressions.

For this purpose, we aim to select the most useful local features from both 2D and 3D facial data using the formalism of *feature selection*. In the next section (Section 5.1), we propose a novel, local feature-based face representation method based on two-stage subset selection where the first stage finds the informative regions and the second stage finds the discriminative features in those locations. The key motivation is to learn the most discriminative regions of a human face and the features in there for person identification, instead of assuming a priori any regions of saliency. We use the subset

selection-based formulation and compare three variants of feature selection and genetic algorithms for this purpose. In Section 5.2, we extend the fundamental idea presented in Section 5.1 to the learning of 3D surface features for the 3D face recognition task.

5.1. Feature Selection for 2D Intensity-based Face Recognition

The main idea in a feature-based face representation scheme is the extraction and analysis of local facial features. Salient facial features are first found and then used to code a face. Coding is generally carried out by extracting local image descriptions. 2D Gabor-like filters are found to be very suitable as local descriptors because of their robustness against translation and rotation [81, 82, 83]. It is essential to analyze the contribution of each feature component to the recognition performance. Important parameters of 2D Gabor wavelets are: 1) spatial location of the kernel in the image, 2) kernel orientation, and 3) spatial kernel frequency.

Several studies have concentrated on examining the importance of the Gabor kernel parameters for face analysis. These include: the weighting of Gabor kernel-based features using the simplex algorithm [84], the extraction of facial subgraph for head pose estimation [85], the analysis of Gabor kernels using univariate statistical techniques for discriminative region finding [80], the weighting of elastic graph nodes using quadratic optimization [86], the use of Principal Component Analysis (PCA) to determine the importance of Gabor features [87], boosting Gabor features [88] and Gabor frequency/orientation selection using genetic algorithms [89]. In almost all previous studies, we see two fundamental assumptions: First, the contribution of each feature dimension is analyzed independently of others (*independence assumption*); and second, Gabor kernel placement over the face region is strongly affected by prior knowledge (*saliency assumption*). Placing the kernel at visually salient facial points, e.g., eyes, mouth, etc. is one of the frequently used methods. The first assumption of independence of features is not valid, and one should incorporate more complex methodologies to analyze the relationship between the features. Moreover, the effectiveness of the fiducial points should also be studied systematically, and a better solution would be to learn these locations from given training data for a given task. In our previous work,

we have analyzed topographically important facial locations for both pose estimation and identity recognition [7], and used feature selection methods to extract optimal local image descriptor parameters for frontal face recognition [6]. We have also used such features to calculate bottom-up saliency in a selective attention-based face recognizer [90].

5.1.1. Proposed Approach: Learning the Best Features

Our aim is to relax the *independence* and *saliency* assumptions for face recognition by reformulating the optimal Gabor basis extraction problem as a feature subset selection problem. Doing this, we allow our approach to detect more complex relationships and correlations between feature dimensions, thus extracting a near-optimal Gabor basis. For this purpose, we have devised a two-stage subset selection mechanism [5]. In the first stage, a genetic algorithm is used to find the most informative facial locations. Depending on the locations of these image descriptors, useful frequencies and orientations should be found since specific parts of a face contain high frequency information (e.g., eyes) and some other parts contain low frequency information (e.g., cheeks). Orientation selectivity also depends on the location of the Gabor kernels. Therefore, in the second stage, a floating search method is used to learn the individual parameters, that is, frequency and orientation, of Gabor wavelet-based local descriptors. The overall diagram of the proposed approach is shown in Figure 5.1.

In feature selection, the aim is to select a subset from a given set such that the classification accuracy of the selected subset is maximized [91]. We use sub-optimal sequential and parallel subset selection algorithms in our system. As sequential selection algorithms, best-individual selection algorithm (BIF), sequential forward selection (SFS), and sequential floating forward search algorithm (SFFS) are used [91]. BIF approach simply selects the best k features and performs well only if each local descriptor contributes independently to the discrimination performance. In SFS, at each step, we add the most significant feature with respect to the previously selected subset. SFFS algorithm takes this idea one step further by backtracking to remove the least useful features from an existing feature subset to overcome the nesting effect. As a

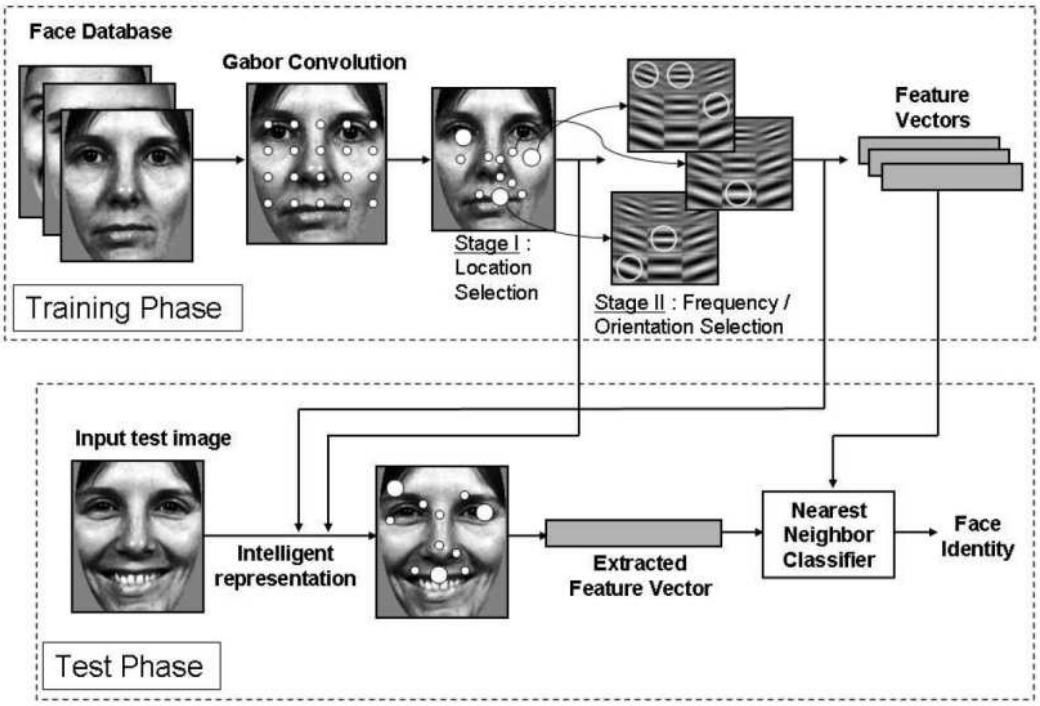
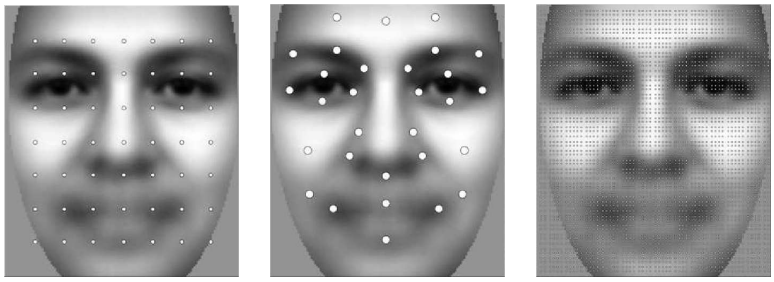


Figure 5.1. Overall diagram of our approach.

parallel subset selection method, we use a genetic algorithm where a chromosome represents a subset and a chromosome's fitness is calculated according to the classification performance of its subset.

We have designed three different methods to learn the important facial locations: *lattice-based sampling*, *landmark-based sampling*, and *dense sampling*. In *lattice-based* sampling (Figure 5.2a), we place a rectangular lattice of size $N \times N$ over the central part of the face region. At each point in the lattice, M different Gabor kernel convolutions are carried out composed of v different frequencies and u different orientations with $M = u \times v$. The concatenation of the magnitudes of the complex outputs of Gabor convolutions forms a feature vector for the whole face. In *landmark-based sampling*, we have identified $S = 30$ salient locations over the face region commonly used by researchers as seen in Figure 5.2b. The aim of constructing such a sampling scheme is to test our prior information as to whether these points are really discriminative and to determine whether these points are really important for recognition. With *lattice-based* and *landmark-based sampling*, in order to determine the important locations among these points, we perform BIF, SFS, and SFFS-based subset selection. We consider each



(a) (b) (c)

Figure 5.2. Different sampling types shown over the mean image: (a) lattice-based sampling, (b) landmark-based sampling, and (c) dense sampling.

feature vector of the i^{th} face location as a single dimension. As a stopping condition, we have defined the cardinality of the resulting subset to a value $d = 15$.

Dense sampling uses full convolutions at each pixel as shown in Figure 5.2c. This dramatically enlarges the cardinality of the feature set. SFS- and SFFS-based algorithms become infeasible for this search space. In order to cope with this problem, we have employed a GA-based subset selection algorithm. In our GA formulation, each gene in a chromosome represents the position of a Gabor kernel. We define the dimensionality of the selected subset as $d = 15$; so, each chromosome consists of d genes. The fitness function depends on the classification accuracy of the selected subset.

Once we find the *locations* of features, we determine the most useful orientations and frequencies of the Gabor kernels at the selected locations, using SFFS. The first stage returns a subset X_{loc} of dimensionality $d \times M$. In the second stage of frequency and orientation selection, we search for a subset X_{fo} of X_{loc} where $|X_{fo}| << |X_{loc}|$. Note that each dimension corresponds to a specific frequency and orientation pair of the outputs of a previously selected Gabor kernel at some specific location. Again, the feature selection criterion in SFFS is the supervised classification accuracy of the selected subset.

5.1.2. Feature Selection Results

In our experiments, we have used a subset of the FERET face database [92] which contains 146 subjects having four frontal images. Faces contain facial expression and illumination variations. Each experimental session contains two training, one validation, and one test image, and we present the average identification results of four sessions. After training with two images per person, the validation set is used to determine when to stop training (that is, adding features) and the test set is used to report the final accuracy. The classifier is the nearest neighbor classifier. We use paired t-test to compare the accuracies for statistically significant difference.

5.1.2.1. Kernel Location Selection. First experiments on kernel location selection were carried out using the lattice-based sampling method. A 7×7 lattice is positioned over the face. Gabor kernels are 15×15 pixels wide, and contain five frequencies and eight orientations [81]. At each lattice point i, we have extracted the local feature vector, υ_i of dimensionality $|\upsilon_i| = 40$. Combining all local feature vectors, we obtain a global feature vector, $\Phi = \{\upsilon_1, \upsilon_2, ..., \upsilon_k\}$ where $k = 49$ for lattice-based sampling. The cardinality of Φ is $|\Phi| = 49 \times 40$. Let Φ_{LOC} be the selected subset of dimensionality d, $\Phi_{LOC} = \{\upsilon_i : i \in 1, ..., k\}$, where d is set to 15 in our experiments. Notice that we treat each local feature vector υ_i as a single feature dimension in the subset selection formalism. Figure 5.3 shows the selected kernel locations in the subset Φ_{LOC} graphically for the experiment S_1 using BIF,SFS, and SFFS methods.

Looking at the BIF results, we see that most of the kernels are located at the upper part of the face, and are highly symmetric. These results comply with the findings of previous works and are expected. Eyes, eyebrows, and forehead seem to have more discriminating information. The symmetry property is not present in SFS and SFFS, since they evaluate the importance of a new candidate feature with respect to the existing subset, and take feature dependencies into account. This is an advantage of SFS and SFFS over BIF: They avoid redundant, symmetric features. Classification accuracies of lattice-based sampling approach for each experimental session are shown

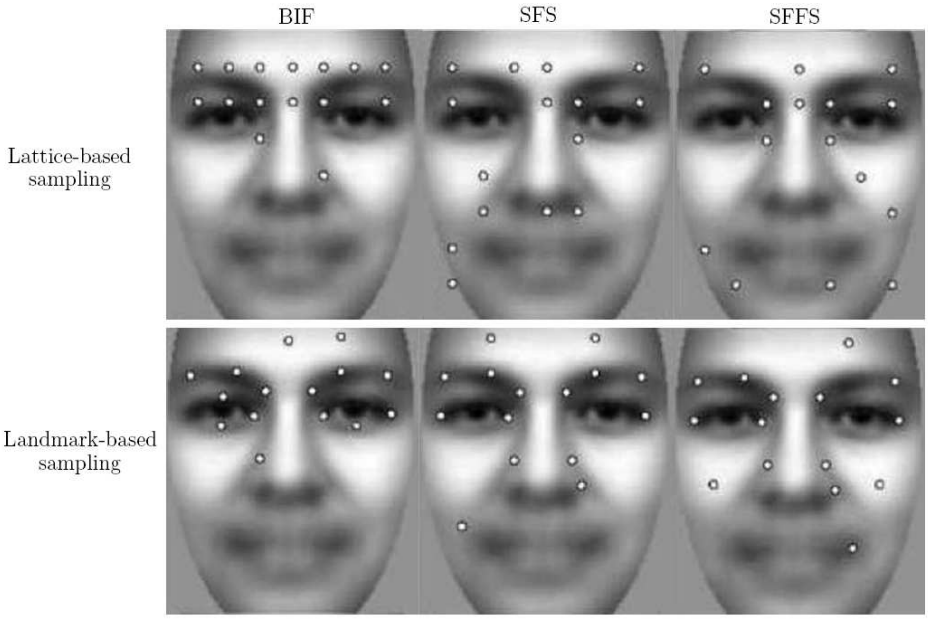


Figure 5.3. Selected Gabor kernel locations for: lattice-based sampling, and landmark-based sampling. Locations found by BIF,SFS, and SFFS are shown at the left, middle, and right columns, respectively.

Table 5.1. Average classification accuracies of lattice, landmark, and dense sampling methods.

	Lattice-based			Landmark-based			Dense Sampling
	BIF	SFS	SFFS	BIF	SFS	SFFS	GA
Mean	84.22	86.96	86.97	82.16	84.39	84.22	88.85
STD	1.65	2.40	1.09	2.93	4.23	2.42	1.97

in Table 5.1. Remember that subset selection is based on classifier accuracy and selection criterion function is calculated on the validation set. Using the 6-fold paired t-test, SFS and SFFS methods are statistically significantly more accurate than BIF, while SFS and SFFS are statistically equivalent (with 95 per cent confidence) again proving wrong the independence assumption.

The same set of experiments were carried out for landmark-based sampling. Figure 5.3 shows the locations of selected kernels in the set Φ_{LOC} for landmark-based sampling. As in the lattice case, BIF approach favors the upper face region by selecting symmetric locations around eyes, eyebrows and forehead. We see that the lower part

of the nose also contributes to the subset. With SFS and SFFS, although the contribution of the nose region and cheeks are more visible, forehead, eyes, and eyebrows are generally found to be informative. The classification performance of landmark-based sampling is shown in Table 5.1. Again, we see that SFS and SFFS are significantly more accurate than BIF and that SFS and SFFS are statistically equivalent. An important observation is that lattice-based sampling is more accurate than landmark-based sampling. This indicates that our prior beliefs in saliency regions is not always correct and that it is better to extract salient locations from data.

In dense sampling, parallel search for a subset Φ_{LOC} is done via constructing genetic chromosomes of size $|\Phi_{LOC}|$ where each gene points to a location in the face image. As in previous experimental settings, $|\Phi_{LOC}|$ is set to 15. As the fitness function, we have used the recognition performance of the subset on the validation set. The single-point crossover operator was implemented to produce new individuals. Since we have the (x, y) coordinates in genes, the mutation operator is implemented as a displacement vector, where the gene to be mutated is displaced by a vector $\eta = \{\eta_x, \eta_y\}$. In both operators, we require that the coordinates of face points in a single chromosome do not overlap by more than a specified amount in order to extract independent local information and this distance is selected to be 20 pixels. The probability of crossover and mutation are selected to be $P_c = 0.5$ and $P_m = 0.05$, respectively. The selection of a new population is based on the probability distribution of fitness values. For quick convergence, elitism is employed, where the elitism ratio is 0.05. The initial population size is 1600. GA terminates when there is no improvement on the accuracy of the best individuals for a specified time interval.

In Figure 5.4, the 15 feature points found by the best individuals of GAs are shown. From the figures, it is clear that the outline of the face, the outline of the nose region, eyes and eyebrows contribute to the most discriminative subset Φ_{LOC}. Almost in all configurations, cheeks, mouth region and the center area of the forehead are absent. In S_1 there is a feature point outside the face area. This may happen because of two reasons: i) the sub-optimal convergence of the GA algorithm, ii) the selected point does not positively or negatively contribute to the recognition performance (i.e.,

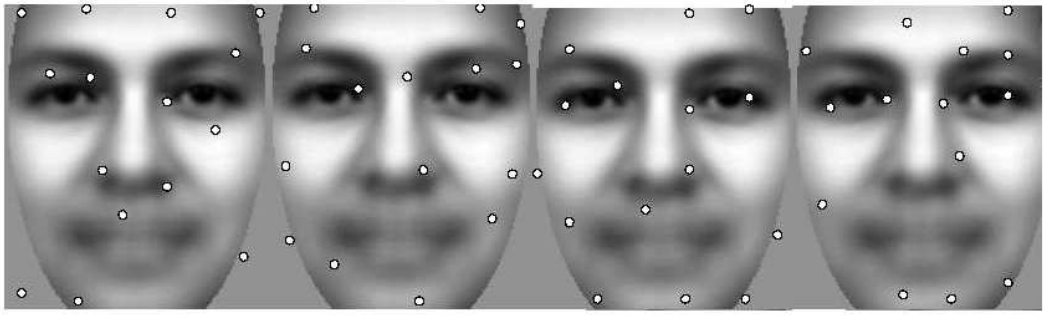
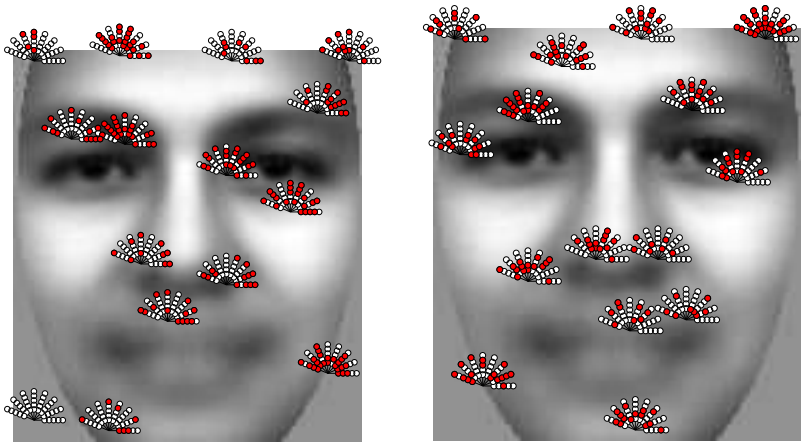


Figure 5.4. Selected kernel positions found by the GA.

effectively there are 14 useful points). The average recognition performance of GA-based location selection is shown in Table 5.1.

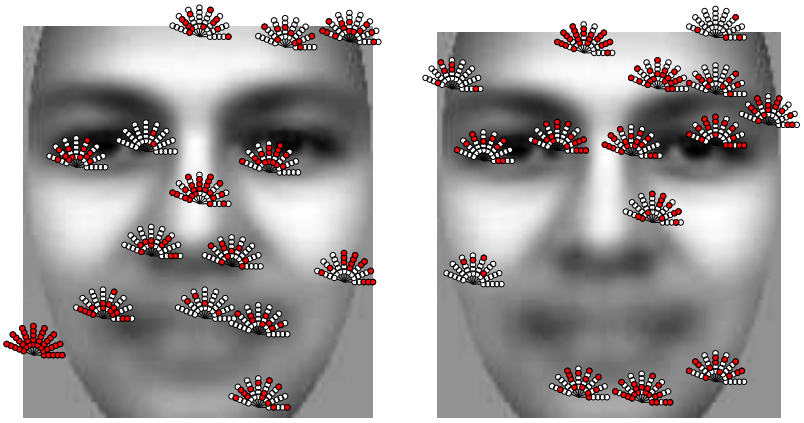
The comparison of classification accuracies of lattice, landmark and dense sampling methods shows that dense sampling using GA performs the best. Lattice-based sampling is found to be statistically more accurate than landmark-based sampling, and between lattice-based and landmark-based sampling, SFS or SFFS on lattice-based sampling is the most accurate. These results indicate that our prior beliefs as to the saliency of certain regions for discrimination (as in landmark-based sampling) are not true and that it is better to allow a general sampling from a grid (as in lattice-based sampling) and it is even better to allow a more general sampling from the whole image (as in dense sampling).

5.1.2.2. Kernel Frequency and Orientation Selection. Now, our aim is to select the useful frequency and orientation pairs from Φ_{LOC} to construct the subset Φ_{FO}, where $\Phi_{FO} \subset \Phi_{LOC}$. Since *dense sampling method* is the top performer in the previous part, we will continue our experiments using its output as our input set in this section. Recall that Φ_{LOC} consists of local feature vectors υ_i, each υ_i contains magnitudes from Gabor kernel convolutions and $|\Phi_{LOC}|$ is $15 \times 40 = 600$. Frequency and orientation (F/O) selection is carried out using the SFFS algorithm since our experiments have shown that it has the best trade-off between complexity and accuracy. The termination condition is determined empirically by observing the behavior of the classification rate on the validation set, and the dimensionality of the subset Φ_{FO} is set to a value where



(a) S_1 (b) S_2

(c) S_3 (d) S_4



(e) S_5 (f) S_6

Figure 5.5. Selected frequency and orientation pairs at the selected kernel locations. Filled circles on each oriented line represent the selected kernel frequency where innermost circles are for low frequencies and outermost frequencies are for high frequencies. Oriented lines represent the kernel orientations.

the classification performance does not improve significantly for a specified time. Our experiments have shown that the target dimensionality of 200 is sufficient for best accuracy on the validation set, which implies a decrease of complexity to one-third Figure 5.5 shows the selected F/O pairs at their specific facial locations for each of the six sessions. In general, we see that the selected kernel orientations are correlated with the underlying characteristics of facial texture. This is more obvious in locations where non-complex local facial directions are present, i.e., at the outline of faces. In terms of frequencies, some positions favor low frequencies, some high frequencies, and in some places, both of them are used together. The average recognition performance of Φ_{FO} is found to be 87.31 per cent (with STD=1.73). This indicates that the dimensionality can be decreased from 600 to 200 without losing from accuracy.

5.2. Local Representations for 3D Face Recognition

Representing 3D faces locally is advantageous because holistic approaches may suffer from local deformations. Another important advantage is that it is possible to learn informative regions for the identification task. For this purpose, we choose to represent facial surfaces locally. We distinguish two different types of local representations: 1) biologically inspired division of facial surfaces (*semantic division* or *part-based division*), and 2) a regular *patch-based division*. Part-based local representation scheme segments the facial surface according to meaningful facial parts, such as forehead region, eye region, and mouth region. On the other hand, the regular patch-based representation scheme considers the facial surface as a more general free-form surface and employs regular surface primitives such as rectangles or circular disks to form local regions. In the rest of this thesis, we specifically refer to the semantic division process as the part-based scheme, and refer to the regular division scheme as the patch-based scheme. Figure 5.6 shows a sample division of a facial surface according to meaningful regions. Automatic segmentation of a facial surface into meaningful parts is a complex problem and needs an accurate detection of several fiducial landmarks. In our work, we obtain the facial parts manually on the average face model. Thus, fine registration of faces to the average face model results in the segmentation into parts.

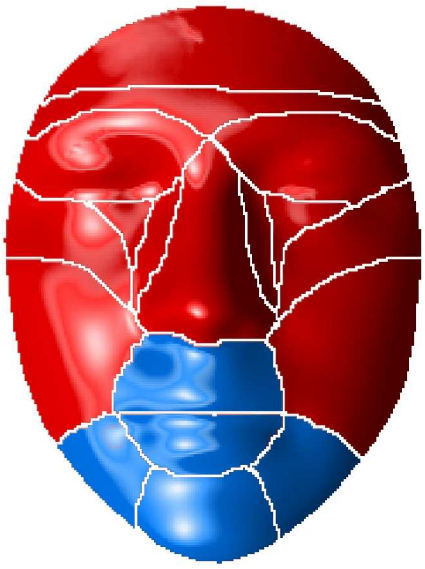


Figure 5.6. Local facial regions used in the part-based representation scheme. These regions are determined manually on the average face model.

In the patch-based segmentation scheme, we use rectangular windows in order to obtain regular regions. Since 3D faces may be considered as 2.5D surfaces, the determination of rectangular windows can be accomplished on a 2D planar surface, and then, orthographic projection can be applied to construct 3D rectangular patches. Figure 5.12 shows several examples of rectangular patch-based segmentations with different window sizes.

5.3. Features for Local Representations

Surface features that are used to represent local regions are dependent on the division scheme employed. However, low-level features such as point coordinates, surface normals and curvature values are common to all division schemes. Therefore, in both part-based and patch-based division schemes, we make use of these low-level features. However, their usage may differ according to the segmentation methods used. The details of the features used in these segmentation methods can be explained as:

- *Part-based Representation Scheme*: Let Φ be a part-based representation of a facial surface that consists of k local parts: $\Phi = \cup_{i=1}^{k} \Phi_i$. Here, each Φ_i corresponds

to regions such as eyes, nose, mouth, cheeks, etc. Assume that we use point coordinates, $p = \{p_x, p_y, p_z\}$, as low-level features. Then, each part is represented as $\Phi_i = \cup_{j=1}^{m} p_j$ where m is the number of points in part Φ_i. This representation approach uses all of the low-level features present in the surface parts to represent that part. It is possible to replace point coordinates with surface normals or curvature directions in this scheme. Thus we can obtain different part-based features. Since all of the low-level features are used to denote patches in the part-based representation scheme, this type of feature usage methodology can be called as *dense feature scheme*.

- *Patch-based Representation Scheme*: Using the similar methodology as in the part-based approach, dense feature scheme can be applied to patches as well. Let Ψ be the patch-based representation of a face containing n patches: $\Psi = \cup_{i=1}^{n} \Psi_i$. Each patch Ψ_i is represented as a collection of low-level features in the dense feature scheme: $\Psi_i = \cup_{j=1}^{m} p_j$. As in the part-based approach, surface normals or curvature directions can be used as low-level features as well. Since patch-based division approach employs regular windows and may cover smaller regions than the part-based scheme, it is possible to use local descriptors as features. We refer to the descriptor-based feature extraction scheme as *patch descriptor scheme* in the rest of this thesis. In the *patch descriptor-based* feature extraction scheme, not all of the low-level features are used to represent a local patch. Instead, a more compact descriptor is computed and used as a feature to represent the patch. In our work, we use the statistical mean operator to compute descriptors. Therefore, in patch descriptor-based scheme, each local patch Ψ_i is represented by $\Psi_i = \{d(p_j), j \in 1 \dots m\}$, where $d(.)$ is the mean operator that computes the descriptor of m point coordinates. It is possible to compute descriptors of surface normals or curvature directions also. Figure 5.7 illustrates the dense feature scheme and patch descriptor approaches for a sample face. As can be seen in Figure 5.7, the dense scheme stores all point cloud coordinates, or surface normal directions over the patch surface, whereas the patch-descriptor approach stores only a statistic computed from all of the low-level features.

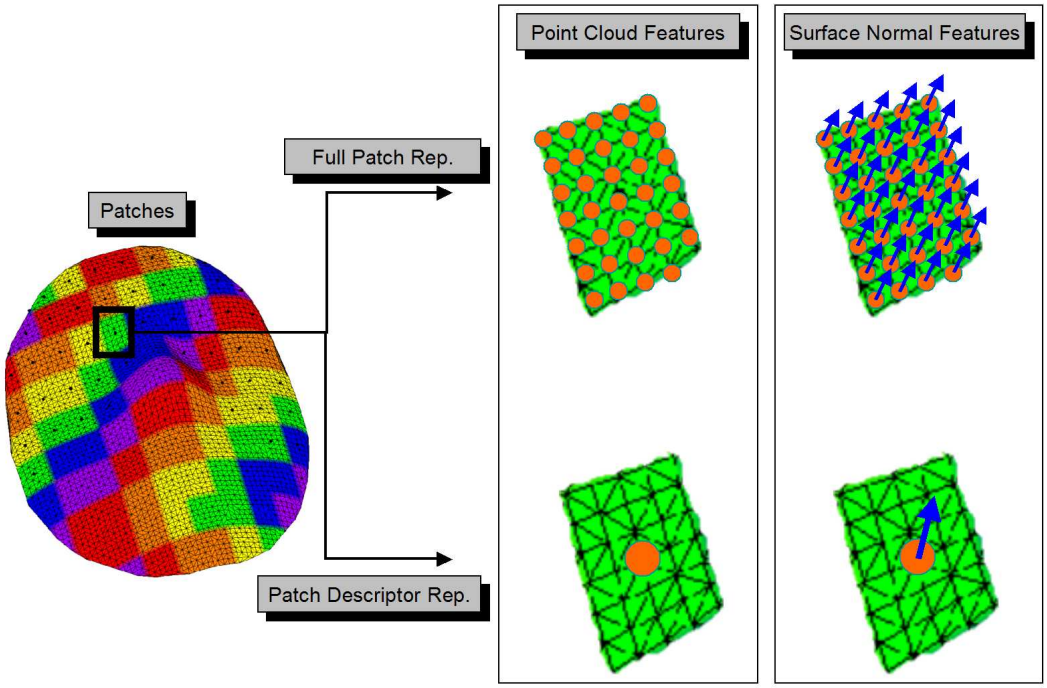


Figure 5.7. Illustration of the *dense feature scheme* and *patch descriptor scheme* for point cloud and surface normal features.

5.4. High-level Feature Analysis: Selection and Extraction

In this section, we briefly overview some of the feature analysis techniques that can be used to select or extract high-level features from local parts. We focus on two different methodologies: 1) feature subset selection techiques, and 2) statistical feature extraction techniques. Both of them will be applied to patch-based representation scheme in order to boost the identification accuracy.

5.4.1. Local Region Selection

We use near-optimal feature selection techniques to find the most discriminating patch subsets for identification. Our aim is to find the patch subset $\Psi = \cup_{i=1}^{c} \Psi_i$ where $c << n$ (n = the number of patches over the facial surface). In this method, *dense feature scheme* is used. Formulating a local feature-based 3D face recognition problem as a subset selection methodology has three important advantages: 1) Floating backward elimination algorithm takes into account the dependencies between features, 2)

Regions which are not selected can be discarded from the representation, thus allowing to reduce the representation complexity 3) Floating backward elimination is a supervised procedure which uses the class information in determining the subsets. Similar methodology was presented in Section 5.1 to find discriminatory feature subsets for 2D face recognition problem. In feature selection, the goal is to find a subset maximizing a selected criterion. This criterion can be inter-class distance measure or the classification rate of a classifier. The optimal solution could be found by using exhaustive search. However, for higher dimensional problems, this solution is unusable. Alternative to optimal algorithms, several fast sub-optimal algorithms can be used. In order to find the most discriminative image locations of faces for recognition, we have used floating backward search (SFBS). SFBS tries to remove a feature from the initial set, and then tries to add previously removed features to the current set if the inclusion is beneficial. Nested removal and addition operators in SFBS increases the run-time complexity of the search process, however this methodology produces near-optimal subsets.

5.4.2. Statistical Feature Extraction

Feature subset selection method can be viewed as a dimensionality reduction technique. It selects the most useful features according to some criteria such as classification rate. An alternative would be to use statistical feature extraction techniques for dimensionality reduction. For this purpose, we propose to use Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) to extract features. For these methods, we use *patch descriptor representation* of faces. Formally, let the face Ψ be represented by n patch descriptors: $\Psi = \cup_{j=1}^{n} \Psi_j$ where $\Psi_j = d(p_k), k \in 1 \ldots m$. If p_k's are point coordinates, then the patch descriptor operation $d(.)$ produces 3-vectors for each patch. Concatenation of n 3-vectors constructs a feature vector. Once these vectors are formed, PCA and LDA analysis can be carried out. Note that we apply the statistical dimensionality reduction techniques to the patch descriptors, not to all of the low-level features. By applying PCA or LDA, we form a new subspace of dimensionality s, ($s << n \times 3$), and represent any face using PCA or LDA coefficients: $\Phi = \{c_1, c_2, ...c_s\}$.

5.5. Experimental Results on the 3DRMA Database

In this section, we present the results of the identification experiments on the 3DRMA database for both local region selection and statistical feature extraction methods. We specifically deal with patch-based representations in this section, and does not report on part-based scheme. As low-level features, we employ point coordinates and surface normals. We exclude the use of curvature-based features since the quality (i.e., resolution) of the 3DRMA database is poor. In our experiments, we have used a subset of the 3DRMA dataset [40], which consists of 106 subjects each having five or six shots. The data is obtained with a stereo vision assisted structured light system. On the average, faces contain about 4,000 3D points, and they cover different portions of the faces and the entire data is subject to expression and rotation changes. To be able to statistically compare the algorithms, we have designed five experimental sessions. Training and test set configurations of each experimental session are: $S_1 = Tr : \{1,2,3,4\}, Ts : \{5,6\}$, $S_2 = Tr : \{1,2,3,5\}, Ts : \{4,6\}$, $S_3 = Tr : \{1,2,4,5\}, Ts : \{3,6\}$, $S_4 = Tr : \{1,3,4,5\}, Ts : \{2,6\}$, $S_5 = Tr : \{2,3,4,5\}, Ts : \{1,6\}$. Numbers in Tr and Ts sets denote which images of each subject are placed into the training and test set, respectively. At each session, there are 193 test shots. In order to determine the best subset Φ_{best}, we have to use only training instances, and then test the accuracy of Φ_{best} on the test instances. For this purpose, four cross-validation sets have been formed from training examples.

5.5.1. Local Region Selection Results

Suppose that at the j^{th} iteration of the SFBS algorithm, we have a subset Φ^j containing several patches. The recognition performance of Φ^j is calculated as the average of the four cross-validation experiments. We have divided the whole facial region into 93 non-overlapping rectangular patches. Each face contains 3,389 points which are densely registered to the average face model. On the average, central patches contain 36 points. Let Φ_{ALL} be the set containing all 93 patches. The average recognition performance of Φ_{ALL}^{PC} for *point cloud representation* in five experiments is found to be 95.96 percent (See *PC-All regions* entry in the Table. 5.2). In *surface normal*

representation, the average recognition accuracy of all regions, Φ_{ALL}^{SN}, is 99.17 percent. By applying the SFBS algorithm, we have found best subsets, Φ_{BEST}^{PC} and Φ_{BEST}^{SN} for point cloud, and surface normal representations, respectively. The average recognition performances of Φ_{BEST}^{PC} and Φ_{BEST}^{SN} in the test set are 96.79 and 98.14 percent.

Table 5.2. Average classification accuracies of the floating backward selection of non-overlapping regions. Dense feature scheme is used. (PC = Point cloud, SN = Surface normals).

Method	Dimensionality	Accuracy
PC - All regions	3,389	95.96
PC - Best Subset	53×36	96.79
SN - All regions	3,389	99.17
SN - Best Subset	48×36	98.14

The selected patches of Φ_{BEST}^{PC} and Φ_{BEST}^{SN} are shown in Figure 5.8 as dark regions. These results confirm that by using SFBS method, we can reduce the dimensionality of the face representation by half, and still have a comparable recognition accuracy. Note that the recognition performance of Φ_{BEST}^{PC} is better than Φ_{ALL}^{PC}.

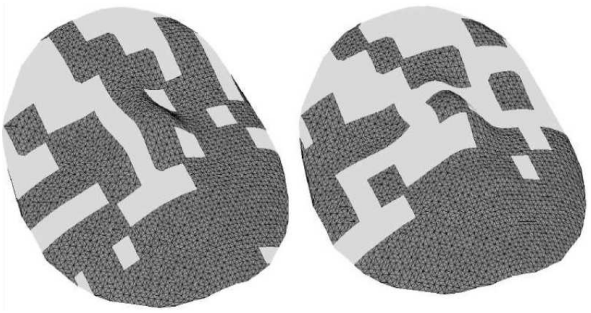


Figure 5.8. Selected regions (in dark color): left: point cloud, and right: surface normal representations.

5.5.2. Statistical Feature Extraction Results

For PCA and LDA-based face representation methods, we use a different patch formation scheme. In this scheme, we have formed overlapping regions over the face, thus increased the number of patches. In these experiments, 327 overlapping regions

are formed, where the size of each patch is the same as in the non-overlapping scheme. In the non-overlapping case, each region is described by the 3D points lying on that region. In the overlapping case, each patch is represented by a *patch descriptor*. In point cloud representation, the mean of the 3D coordinates of each patch's point cloud is used as a patch descriptor. In surface normal representation, the mean of the surface normals of a patch is used as a patch descriptor.

The mean recognition accuracies of the point cloud and surface normal representations using patch descriptors are found to be 96.06 and 99.28 percent respectively (See *PC-All regions* and *SN-All regions* entries in the Table 5.3). These are the classification accuracies of using all patch descriptors without applying PCA or LDA. The use of patch descriptors in the overlapping division scheme improved the classification accuracy when compared to the non-overlapping case. It is found that dimensionality reduction using PCA decreases the recognition performance to 90.88 and 94.51 percent for point cloud and surface normal-based representations, respectively (See *PC-PCA* and *SN-PCA* entries in Table 5.3). However, LDA is found to be very beneficial in reducing the dimensionality of patch descriptor-based face representation scheme. LDA obtained 99.69 percent accuracy in both point cloud and surface normal representations, using 60 and 40 features, respectively.

Table 5.3. Average classification accuracies of the statistical dimensionality reduction techniques on patch descriptor-based feature scheme. 327 overlapping regions are formed. (PC = Point cloud, SN = Surface normals).

Method	Dimensionality	Accuracy
PC - All regions	327×3	96.06
PC - PCA	70	90.88
PC - LDA	60	99.69
SN - All regions	327×3	99.28
SN - PCA	70	94.51
SN - LDA	40	99.69

5.5.3. The Effect of Patch Resolution

In the previous section, we have presented the results of the classification experiments where a fixed patch resolution is used for both overlapping and non-overlapping patch division strategy. The patch height and width are selected to be $\frac{H}{10}$ and $\frac{W}{10}$, where H and W denote the height and width of the cropped mean face. We have shown that the use of 327 patch descriptors is slightly better than using raw point clouds and surface normals, where patch-based classification accuracies are $d_{depth} = 96.06, d_{normals} = 99.28$ and raw recognition accuracies are 95.96 and 99.17 percent for $3,389$ point clouds and surface normals, respectively. This finding motivates us to analyze the effect of patch resolution on the classification performance. For this purpose, we have used different patch resolutions for segmenting the whole facial region. Figure 5.9 depicts a subset of various patch resolutions that we have used. From coarse to fine scale, we have extracted different face segmentations where the numbers of patches used are : 9, 16, 25, 34, 45, 72, 105, 124, 145, 166, 183, and 211.

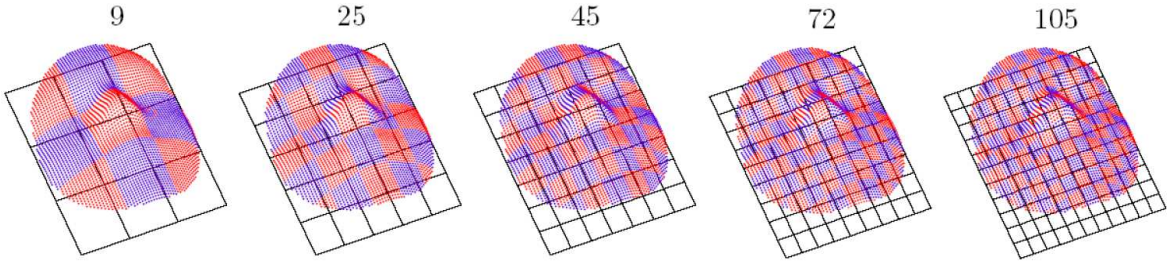


Figure 5.9. Different patch resolutions and the total number of patches found over a facial surface

Table 5.4 displays the classification accuracies of surface normal-based and point cloud-based patch descriptors on different patch resolutions. The first column shows the number of local patches formed over the face region and the second column shows the average number of 3D points at each local patch. Patch descriptors form a feature vector, and as in previous experiments, 1-nn algorithm is used as a pattern classifier. Figure 5.10 graphically displays the recognition rates found in Table 5.4.

It is evident by analyzing Table 5.4 that significant dimensionality reduction is

possible without a significant loss in classification accuracy. Recognition system can obtain a very good accuracy using approximately 100 patch descriptors. Using only 105 non-overlapping patches, the system obtains 99.17 and 96.17 percent recognition accuracies, and using more patches does not improve the accuracy significantly. These results indicate that the local patch idea which can be considered as a local averaging operator, helps to filter out redundant information. Another advantage would be the de-noising characteristic of local averaging operation. However, in our experimental face database, since we have no local perturbations or noise, this behavior is not visible.

Table 5.4. Classification accuracies of surface normal and point cloud representations for different patch resolutions. First column denotes the number of patches and the second column shows the average number of 3D points in each patch.

Number of Patches	Patch Density	Surface Normal Accuracy	Point Cloud Accuracy
9	375	92.64	72.64
16	225	94.61	86.01
25	136	95.96	91.81
34	99	97.62	92.64
45	84	97.62	94.82
72	47	98.86	95.34
105	32	99.17	96.17
124	27	99.17	95.86
145	24	99.28	96.37
166	26	99.07	96.17
183	19	99.28	96.37
211	18	99.17	96.48
230	15	99.17	96.06

5.6. Experimental Results on the FRGC v2.0 Database

In this section, we present the identification accuracies of the part-based and patch-based representation schemes on the FRGC v2.0 database. Compared to the

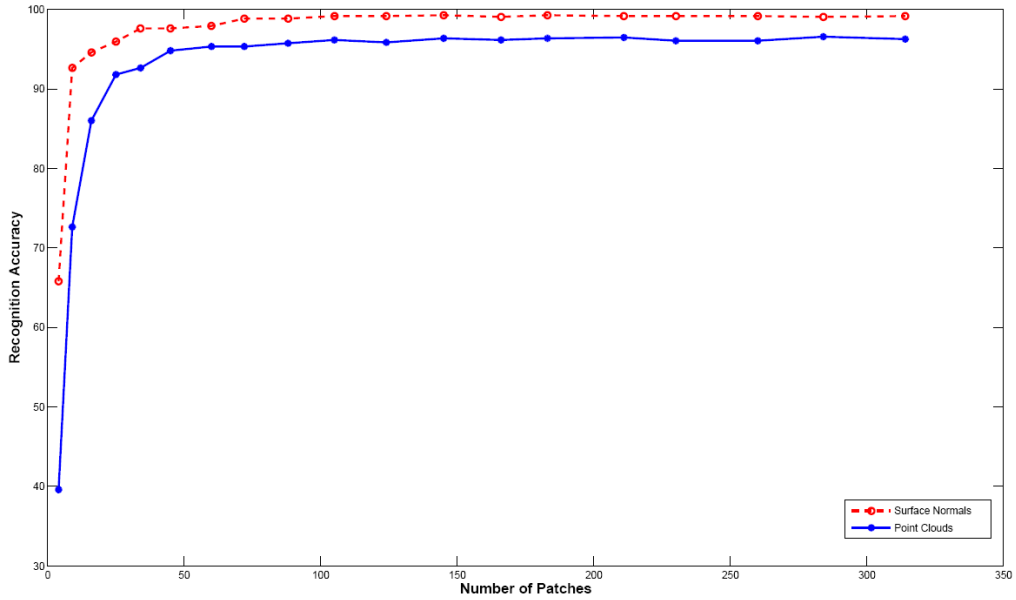


Figure 5.10. Recognition accuracy versus patch number plot for surface normal and point cloud-based face representations.

3DRMA database, FRGC v2.0 face database is much more bigger and contains texture information. The utilized part of the FRGC v2.0 database contains 3D scans of 310 subjects. Each subject has more than three scans. There are 3602 3D facial scans in total. Data was acquired from subjects under challenging illumination conditions, and it exhibits expression variations. There are six expression categories: smiling, astonished, sad, frowning, puffy cheeks, disgust. Sample images of these expression categories can be seen in Figure 5.11.b. On the average there are 35,000 to 40,000 3D points over the facial region which is a very sufficient resolution for the identification task. Note that, in the 3DRMA database, there are 4,000 points over the facial region typically. The adequate resolution of the FRGC v2.0 database makes it reasonable to apply more sophisticated feature extraction methods such as curvatures.

We have formed two different experimental protocols. The first protocol implements neutral-to-neutral matching (E_n) whereas the second one implements neutral-to-expression matching (E_e). Both of these experiments use the same training set which contains single neutral images of 310 subjects. In E_n, there are 1804 probe (test) scans, and in E_e, there are 1488 probe scans. As low-level features, we use point clouds (PC),

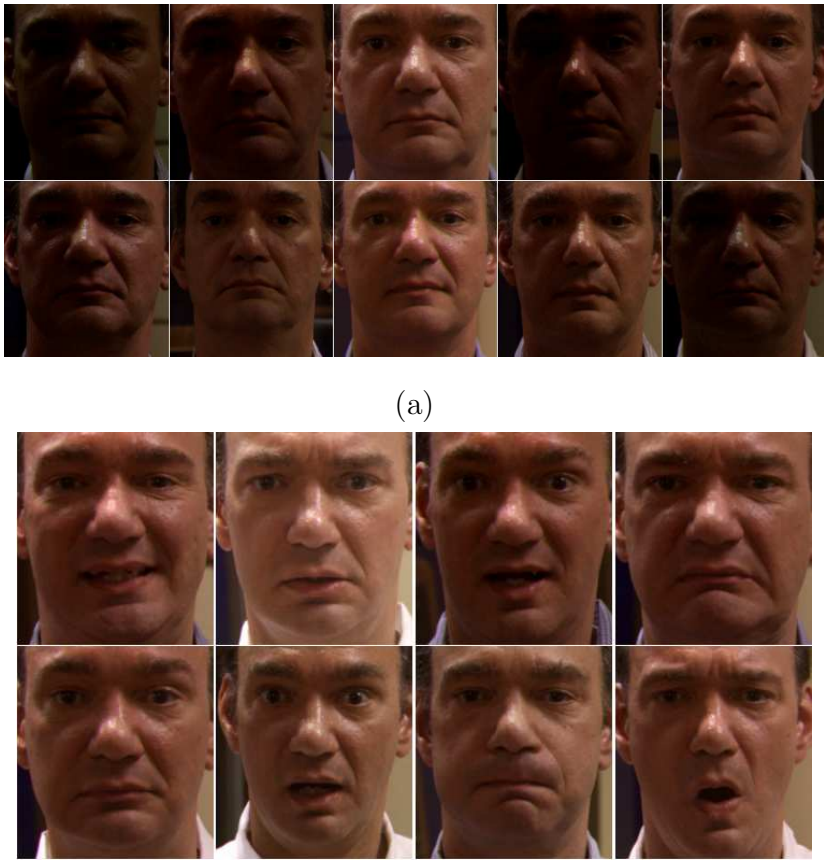


Figure 5.11. Sample images from the FRGC v2.0 database: (a) Neutral images, and (b) expression variations

surface normals (SN), and principal curvature directions (CURV). Table 5.5 presents the identification accuracies of the PC, SN and CURV-based matchers. Each 3D facial surface contains approximately 33,000 points. In neutral-to-neutral protocol E_n, CURV method obtains the best rank-1 identification accuracy with 80.43 per cent classification accuracy. The second best matches in the E_n protocol is found to be the SN approach with 77.05 per cent accuracy. In the neutral-to-expression protocol E_e, identification accuracies decrease significantly, where the two best classifiers SN and CURV obtain 67.33 and 66.06 per cent correct classification rates, respectively. An important observation when we compare the E_n and E_e protocols is that PC approach seems to be the most sensitive matcher to the expression variations. It has a 20.54 per cent performance degradation in the E_e protocol. The most resistant matcher is found to be the SN approach with a 9.72 per cent performance loss in the E_e.

Table 5.5. Baseline rank-1 identification performances of the point cloud (PC), surface normal (SN) and principal curvature directions (CURV) on the FRGC v2.0 database.

	E_n	E_e	$(E_n - E_e)$
PC	75.78	55.24	20.54
CURV	80.43	66.06	14.37
SN	77.05	67.33	9.72

5.6.1. Local Representation Results on the FRGC v2.0

In the FRGC v2.0 experiments, we have employed both part-based and patch-based surface division schemes. Manually defined facial parts are shown in Figure 5.6. Similar to the methodology used in the 3DRMA database, we obtain regular rectangular patches. The patches obtained with different window sizes are depicted in Figure 5.12. The depicted window sizes are 5, 10, 15 and 20 (from left to right) in Figure 5.12. Patch-based division is first performed on the average face model. Then, for a given probe image, these patches are found by using the dense correspondence information between the probe image and the average face model.

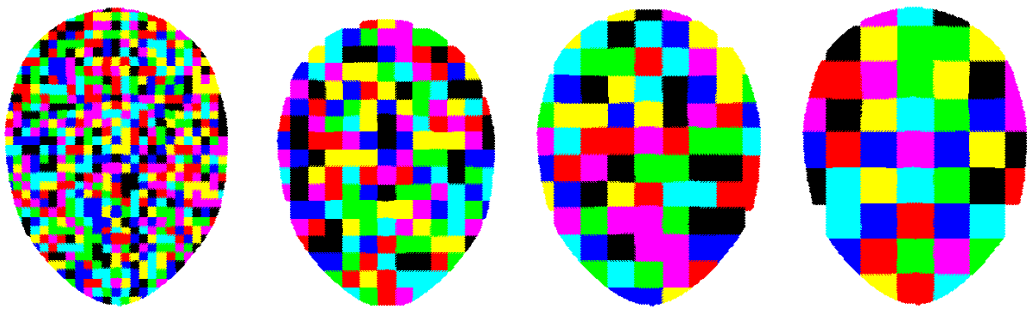


Figure 5.12. Rectangular divisions of a sample facial surface. From left to right, window sizes are 5, 10, 15, and 20. Patches are colored randomly.

As low-level features, point clouds, surface normals, and principal curvature directions are used. Using a patch window size of 10, we have formed 174 local rectangular regions (See Figure 5.12). For each patch, we extract patch descriptors using averaging. Illustration of these patch descriptors is provided in Figure 5.13. Each patch is represented by its mean point coordinate, its mean surface normal, and its mean curvature directions. In Figure 5.13, point coordinates are depicted as black dots, surface nor-

mals are shown by blue lines, and curvature directions are expressed as two red lines. Note that there are two principal curvature directions that correspond to minimum and maximum curvature values. Using these patch-descriptors of size 10, any face can be represented by 1) 174×3 point cloud features, 2) 174×3 surface normal features, and 3) $2 \times 174 \times 3$ curvature features.

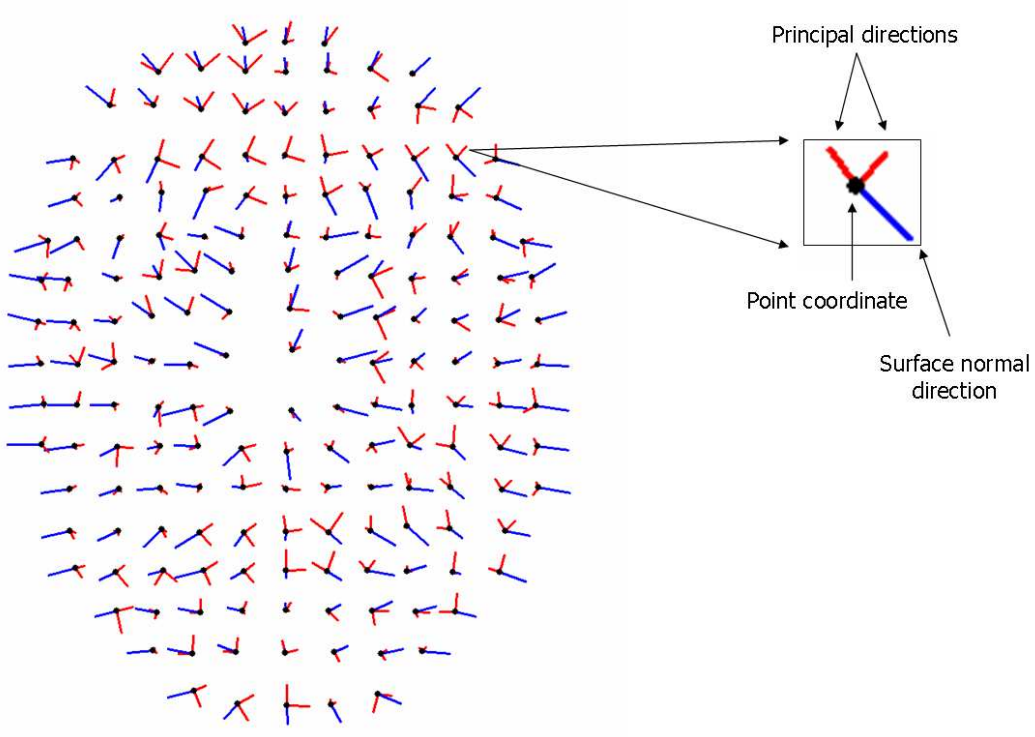


Figure 5.13. Patch descriptors: point coordinates (black dots), surface normals (blue lines), and principal curvature directions (red lines).

Table 5.6 presents the rank-1 correct classification accuracies of the patch descriptor-based representation schemes. Identification performance of each method is given for different patch sizes from 5 to 20 (rows). For each method, we present the identification performances for two different experimental protocol: neutral-to-neutral E_n and neutral-to-expression E_e. Performance figures presented in Table 5.6 should be compared to the accuracies of the PC, SN, and CURV methods presented in Table 5.5. Based on the recognition accuracies for different patch sizes, we see that patch size of 5 and 10 generally performs better than other patch sizes. This is especially visible for the PC method. For each method in Table 5.6, we also provide their performance deviations (in parentheses) according to the baseline performances found in Table 5.5.

By looking at these deviations, we see that PC-based patch descriptors almost perform equally or slightly better than using all of the low-level features. For example, using a patch size of 10, PC-based patch descriptors attain 76.11 per cent identification rate, whereas baseline PC performance is 75.78 per cent (See Table 5.5). Similarly SN-based patch descriptors improve the accuracy by 1 to 3 per cent, depending on the patch size. When we compare the experimental protocols, we see that performance improvement is generally bigger in the neutral-to-neutral protocol.

Table 5.6. Patch descriptor results for patch sizes 5,10,15, and 20.

PC		SN		CURV	
E_n	E_e	E_n	E_e	E_n	E_e
76.33(0.55)	**56.65**(1.41)	78.22(1.17)	**67.94**(0.61)	87.03(6.59)	73.5(5.42)
76.11(0.33)	55.11(-0.13)	**80.38**(3.33)	67.20(-0.13)	87.47(7.04)	74.6(8.54)
74.78(-1.00)	53.09(-2.15)	79.10(2.05)	63.51(-3.82)	**88.14**(7.71)	74.7(4.43)
74.00(-1.78)	50.00(-5.24)	79.27(2.22)	61.96(-5.37)	88.08(7.65)	68.41(2.35)

Among the three low-level features, curvature-based patch descriptors attain the best performance improvement. In the E_n protocol, patch-based curvature descriptors (patch size 10) obtain 87.47 per cent correct classification rate which is 7.04 per cent better than the baseline CURV method. 8.54 per cent improvement is also present in the E_e protocol. This finding is very important and makes it clear that individual matching of curvature directions at all points may be not optimal. This may be due to the local surface noise that may degrade the curvature direction estimation performance. Another important advantage is the reduced dimensionality of the feature vectors. It is possible to reduce the 3D point count from 30,000 to 174 (for patch size of 10), and still improve the identification rate significantly. Among all of our identification experiments, patch-based descriptors of curvature directions attained the best accuracies.

We have also applied the part-based division scheme to the FRGC v2.0 database. Manually determined regions are shown in Figure 5.6. These regions are segmented for the average face model, and the dense point-to-point correspondence information is

utilized to construct the parts in the given probe image. This approach assumes that facial registration between the average face model and the probe image is accurate enough. Another method would be to divide the probe facial surface independently by using its surface characteristics. Although this approach may provide better segmentation, it requires correct localization of fiducial landmarks. In our generic face model, we identify the following regions coarsely: 1) upper forehead, 2) lower forehead, 3) eyes, 4) nose and its neighborhood, 5) upper cheek, 6) central cheek, 7) lower cheek, 8) upper mouth, 9) lower mouth, and 10) chin.

In part-based representation scheme, we choose to focus on the expression variations present in the FRGC v2.0 database. Experimentally, we select different combinations of local parts, and perform identification experiments on the E_e. As stated before, in part-based representation scheme, all of the low-level features are combined to represent single parts. Therefore, the dimensionality of the feature vectors are bigger than the ones used in the patch-based descriptor approach. Symmetric regions such as left eye/right eye, or left/right cheek are always selected together, so these regions are considered as single parts.

Using different combinations of facial parts, we have formed an exhaustive list of candidate subsets of facial parts. Among them, the one which discards the mouth, lower cheeks, and chin region obtained the best identification accuracy. The rank-1 correct classification rates of this subset is shown in Table 5.7. Although we select these regions according to the neutral-to-expression protocol, we also provide their recognition accuracies for the neutral-to-neutral protocol in Table 5.7.

Table 5.7. Identification accuracies of the part-based local representation technique. Selected parts of the face is shown in Figure 5.6 in red. Mouth, lower cheeks, and chin regions are removed (these regions are shown in blue).

Method	E_n		E_e	
PC	69.96	(-5.82)	57.33	(2.09)
SN	75.44	(-1.61)	68.78	(1.45)
CURV	83.7	(3.27)	73.79	(7.73)

By looking at the results presented in Table 5.7, we see that removal of these lower facial regions generally does not degrade the identification rate in the E_e protocol. Point cloud and surface normal features obtained marginal improvements when compared to the baseline recognition rates presented in Table 5.5. As in the patch-based descriptor approach, curvature directions significantly increase the identification rate when these regions are discarded. In the E_e protocol, CURV method obtains 73.79 per cent identification rate which is 7.73 per cent better than using all regions. However, when we look at the classification rates of the selected regions in the E_n protocol, we observe that point clouds and surface normals degrade the baseline performance. The only exception is the CURV method which improves the recognition rate by 3.27 per cent. This observations reveal that these discarded regions still carry discriminatory information when there are no expression variations.

6. Fusion of 3D Face Classifiers

The 3D face recognition task is inherently multi-modal. Most of the 3D acquisition devices produce both shape and texture data registered to each other. Therefore, it is possible to fuse the information present in both shape and texture channels when arriving to a decision. Aside from being multi-modal, it is also beneficial to integrate various classifiers that are based on a single modality. For instance, pattern recognizers trained on different shape representations may complement each other, and may produce better identification rates when used together.

A survey of classifier fusion techniques used in 3D face recognition systems reveals the dominance of fixed combination rules such as sum and product rules which operate on measurement level outputs [15]. The preference for these methods stems from the following facts: i) they are simple, yet effective; ii) the number of training samples per subject is very limited in face recognition applications, and this limits the use of more advanced classifier combination methods; iii) decision-level integration is flexible, in that a new expert's opinion can be easily incorporated without affecting the existing experts.

In this thesis, we review and compare various decision-level fusion algorithms. Specifically, we perform detailed examination of the following: 1) which fusion rules attain the best identification rates, 2) how many individual experts are needed to obtain a good ensemble, and 3) which type of individual experts should be integrated. Based on our findings, we propose two different fusion architectures, where the first one incorporates confidences during the fusion, and the second one utilizes a serial or cascaded architecture. For each of these two architectures, we develop several variants. In the next section, we start our discussion with a general taxonomy of decision-level fusion algorithms, and then formulate specific instances that are used in our experiments. Lastly, we explain our proposed confidence-aided and cascaded fusion architectures.

6.1. Overview of Fusion Methods

Combining different classifiers with the aim of increasing classification accuracy is a common technique in the pattern recognition discipline. Classifiers which are different from each other can be constructed 1) by using different classes of pattern recognizers such as neural networks, decision trees, or k-nearest neighbor algorithms, 2) by training the same algorithm on different training sets, or 3) by designing classifiers that use different representations (or, modalities) of the same object. It is widely believed that fusion of diverse experts produces the best ensemble performance in terms of recognition accuracy [93].

Combination methods differ according to the type of information that comes from individual classifiers. Although individual classifiers can be different, their outputs can be grouped according to the following categories [94]:

- *Type 1 (The Abstract Level)* Let C_i denote the i^{th} individual classifier. In abstract level fusion, each classifier C_i provides a class label $s_i \in \Omega$, where Ω is the set of class labels.
- *Type 2 (The Rank Level)* The output of each C_i is a list of class labels ranked in order of the probability of being the true class.
- *Type 3 (The Measurement or Score Level)* In measurement or score level, classifier C_i provides a c-dimensional vector $[d_{i,1}, \ldots, d_{i,c}]^T$ where c is the number of classes. The value $d_{i,j}$ denote the support for the unknown pattern to come from class ω_j. $d_{i,j}$'s can be probabilities or similarity values.

In the sequel, we briefly review the decision-level fusion methods applicable to 3D face recognition systems.

6.2. Plurality Voting

In abstract level category, plurality voting is the most commonly used one, which just outputs the class label having the highest vote. More formally, plurality voting

can be defined as follows: Assume that classifier C_i outputs binary valued vectors $[d_{i,1}, \ldots, d_{i,c}]^T \in [0,1]$, $i = 1, \ldots, L$, where L is the number of classifiers, and c is the number of classes. $d_{i,j} = 1$ if classifier C_i thinks that the unknown pattern belongs to the class ω_j, and $d_{i,j} = 0$ otherwise. Plurality voting assigns the unknown pattern x to the class having the highest vote:

$$\arg \max_{j=1}^{c} \sum_{i=1}^{L} d_{i,j} \tag{6.1}$$

If there are ties, random assignment is performed. Plurality voting is different from majority voting since the number of votes does not have to be greater than $\frac{L}{2}$ for a pattern to be assigned to a class.

6.3. Borda Count Method

Borda count method can be applied as a combination rule if the individual pattern classifiers output rank lists. For a c class problem, ranks produced by any classifier C_i are in the range of $[1, \ldots, c]$, where 1 is the topmost rank that denotes the highly probable class. Assume that C_i produces $[d_{i,1}, \ldots, d_{i,c}]^T$ where $d_{i,j} \in 1, \ldots, c$, then Borda count method simply selects the class label which has the minimum total rank:

$$\arg \min_{j=1}^{c} \sum_{i=1}^{L} d_{i,j} \tag{6.2}$$

Borda count method is similar to the rank sum method. A modification of Borda count method is possible by restricting the number of classes to be fused. For instance, it is also possible to consider the fusion of top-k ranked classes when using the Borda count method. Selected classes form the so-called *combination set*, and this approach

has the assumption that most probable class of the unknown pattern is at the top ranks of the individual classifiers. Figure 6.3.a illustrates rank-based parallel fusion architecture schematically. Since the individual experts output their decisions in parallel, these type of fusion architectures are called *parallel*.

6.4. Fixed Arithmetic Combination Rules

When the individual classifiers produce class similarity scores or support values, these scores can be combined by using simple arithmetic rules such as sum, product, min, max and median rules [95]. Assume that classifier C_i outputs continuous valued score values: $[d_{i,1}, \ldots, d_{i,c}]^T$ where $d_{i,j} \in [0, \ldots, 1]$. Without any loss of generality, assume that greater values close to 1 mean high similarity. In this case, fixed arithmetic rules such as sum/product/max reach a decision according to the following equations:

- *Sum rule:*

$$\arg \max_{j=1}^{c} \sum_{i=1}^{L} d_{i,j} \tag{6.3}$$

- *Product rule:*

$$\arg \max_{j=1}^{c} \prod_{i=1}^{L} d_{i,j} \tag{6.4}$$

- *Max Rule:*

$$\arg \max_{j=1}^{c} (\max_{i=1}^{L} [d_{i,j}]) \tag{6.5}$$

Note that arithmetic rules assume the classifier outputs to be in a common range. In practice, this is hardly the case, especially when using k-nearest neighbor classifiers. It is therefore necessary to normalize scores before fusing them. There are several ways to perform score normalization:

- Min-max normalization: If the bounds of the score values are known, or can be estimated, min-max normalization can be easily applied to normalize raw scores. Let d be the original score value. The normalized score can be computed as:

$$d^{'} = \frac{d - d_{MIN}}{d_{MAX} - d_{MIN}} \tag{6.6}$$

  where d_{MAX} and d_{MIN} represent the minimum and maximum values of the score range. Depending on the application, d_{MAX} and d_{MIN} values may be known beforehand. Otherwise, they are generally estimated from the training set. When the training set is sparse, these values may be estimated inaccurately. However, this problem is not the case only for the min-max normalization: it is the general problem for all of the score normalization methods.
- z-score normalization: z-score normalization method transforms the raw scores into a new range with the help of sample arithmetic mean and standard deviation as follows:

$$d^{'} = \frac{d - \mu}{\sigma} \tag{6.7}$$

  where μ and σ denote sample arithmetic mean and standard deviation, respectively.
- Tanh-estimators: Tanh-estimators method is similar to the z-score method, but it is designed to be more robust. The normalization is given by:

$$d^{'} = \frac{1}{2}\{tanh(0.01(\frac{d - \mu_{GH}}{\sigma_{GH}})) + 1\} \tag{6.8}$$

  where μ_{GH} and σ_{GH} are robust estimates of the score distribution statistics computed with the help of Hampel estimators. See [96] for the details on the Hampel estimators.

6.5. Confidence-aided Fusion Rules

We have devised a method to improve the abstract-level combination methods using estimated confidences of individual classifiers [2]. The confidences can be attributed according to the similarity scores the classifiers report. A caveat is that it may be misleading to use scores of the top-ranking class labels due to mismatched score normalizations. Score normalization techniques, such as the min-max method, use only the training set, and their generalization ability is not optimal. An example case is presented in the piecewise linear curves in Figure 6.1.a, where the x axis denotes ranks of the nearest class labels, and the y axis denotes the distance scores in increasing order, for a given test pattern. Reading off from the graph at the first label (x=1), we see that the nearest class found by the second classifier has a distance of 0.06, and the nearest distance found by the first classifier is 0.13. Accordingly, the second classifier seems to be more confident than the first one since it outputs a much lower score value. However, in this particular case this is wrong since the score range of classifier 1 (0.13-0.9) is different from that of classifier 2 (0.06-0.8), and generally second classifier gives lower score values. This pitfall is due to insufficient training data in estimating the score normalization parameters. Obviously, if the classifier score ranges were the same, then scores could be used right away as confidences.

To compensate for range disparity, we propose to use a differential confidence measure, that is, the relative distance between the first two nearest neighbors of the classifier. The procedure is as follows: Given a probe image, we generate $d = [d_1, d_2, \ldots d_c]$ the vector of sorted distances to the c classes. Here d_1 is the distance between the test sample and its nearest class in the training set, while d_c is that of the least similar one. An obvious score range for a classifier is $r = d_c - d_1$, while we prefer the more robust median estimate $r = Med(d_1, d_2, \ldots d_c) - d_1$. The score normalization is then effected via Eq. 6.9:

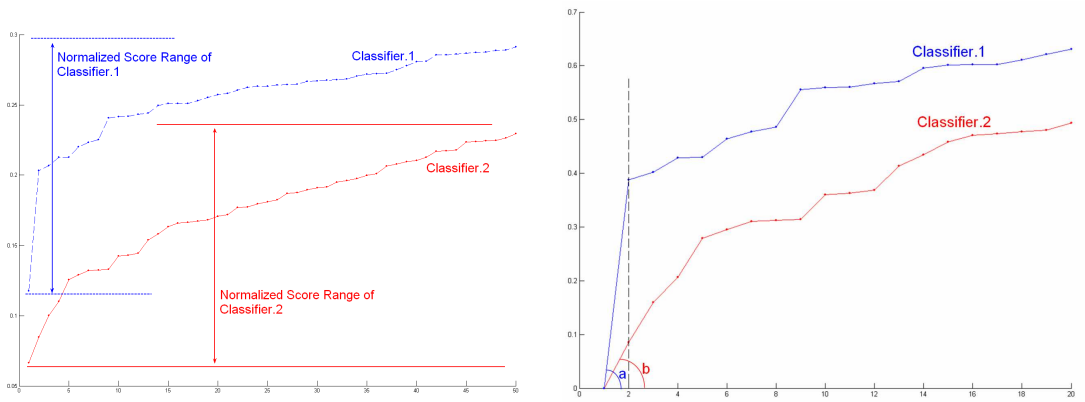
$$d_i^{'} = \frac{(d_i - d_1)}{Med(d_1, d_2, \ldots d_c) - d_1}, i = 2 \ldots c. \tag{6.9}$$

Finally, the classifier confidence score is declared as simply $d_2^{'}$, as plotted in Figure 6.1.b. One can interpret the $d_2^{'}$ corrected confidence as the slope of the $d^{'}$ curve at the x-intercept. With this correction, Classifier 1 becomes the higher confidence classifier. Note that confidence of a classifier is estimated dynamically when a new unknown sample is presented to the system. In addition, it is important to observe that in confidence-aided fusion scheme explained above the outputs of the classifier C_i are 1) the nearest class label ω_j found, and 2) the classifier's confidence value τ_i. With the exception of the availability of τ_i's, this case is similar to the abstract level fusion schemes.

Equipped with confidences, several fusion schemes can be proposed. Here we propose two additional combination alternatives. The first alternative, named *modified plurality voting*, operates exactly like original plurality voting, but differs from it when there are ties. If ties are present, modified plurality voting calculates the average confidences of the classifiers which cause a tie. Then, the class label having the highest average confidence is chosen as output. The second alternative, *highest confidence fusion*, operates similarly as the max-rule in the arithmetic fusion category, but uses confidences as opposed to scores. In practice, highest confidence fusion rule selects classifiers, and does not fuse them as in the max-rule.

6.6. Two-stage Cascaded Fusion

All of the aforementioned fusion methods are examples of parallel fusion schemes: all of the individual pattern classifiers output their decisions in parallel, and the combiner fuses all of their decisions in a single step. This scheme is illustrated in Figure 6.3.a. It is possible to have different structures when combining classifiers such as hierarchical or cascading schemes. We have developed two different two-stage cascaded



(a) (b)

Figure 6.1. An illustrative example of the estimation of confidences for the top ranked classes. (a) Normalized scores (distances) of a test example for each class in the training set (in increasing order). Classifier.1 and classifier.2 have different score ranges (denoted by double arrows), (b) Re-normalized distances calculated from the Eq. 6.9. Slopes a and b denote the estimated confidences for the top ranked class for classifier.1 and classifier.2, respectively.

fusion architectures [2]. Both are based on the following principle: Given an unknown test pattern, the first classifier eliminates the unlikely classes, and retains the highly probable ones and forwards them to the second classifier. The second classifier is a more complex and accurate classifier which implements LDA. The motivation to use LDA in the second classifier is based on the capability of LDA to generate a better discriminative feature subspace when confronted with similar instances. Illustration of the cascaded fusion scheme is given in Figure 6.3.b.

In the first cascaded architecture, named *forward-always fusion*, the first classifier C_f, always forwards the labels of the the highly probable classes to the second classifier, C_s. The number of labels forwarded to (r) is determined experimentally such that the probability of the true class being in the forwarded labels is close to 1, and is set to a fixed number beforehand. Alternatively, r parameter can be tuned automatically using a cross-validation scheme. The task of the second classifier C_s is now to construct a feature space using the LDA. Given r class labels, C_s constructs the LDA space using the training examples of the forwarded r classes. Note that C_s dynamically determines

the LDA subspace according to the forwarded class labels. This procedure could be cumbersome and slow if r is very large. However, if the first classifier is chosen among the accurate ones properly, then r is always a small number, and this does not result in time complexity problems. Once the LDA space is constructed, the unknown face pattern is projected to this LDA space and its feature coefficients are computed. One possible way of obtaining the final decision could be just selecting the nearest class found by C_s. However, we choose to fuse the rank lists of C_f and C_s using Borda count method. It was observed experimentally that combining the decisions of the C_f and C_s produces better classification accuracies.

The second cascaded architecture, named *forward-if-unconfident fusion*, has exactly the same structure as the *forward-always fusion* scheme. The only difference is the use of confidence-assisted decision making module at the output of the first classifier C_f. Using the confidence estimation procedure presented previously, C_f determines whether to use the second classifier or not. If the confidence of C_f is below a certain threshold for a given test pattern, then the help of C_s is needed. In this case, C_s performs the same operations as in the *forward-always fusion* scheme. In this architecture, the only parameter that can be tuned is the confidence threshold. If the threshold is set to a large number, then most of the time the consultation of C_s will be needed. Otherwise, if it is too low, then most of the time the decision of the C_f will be selected. However, the latter case may result in a performance degradation since the reliability of the C_f may be low.

ALGORITHM: Confidence-aided Fusion

FUNCTION: Performs confidence-aided fusion for the identification scenario.

INPUT:

Classifiers: $C = \{C_i, \ldots, C_L\}$

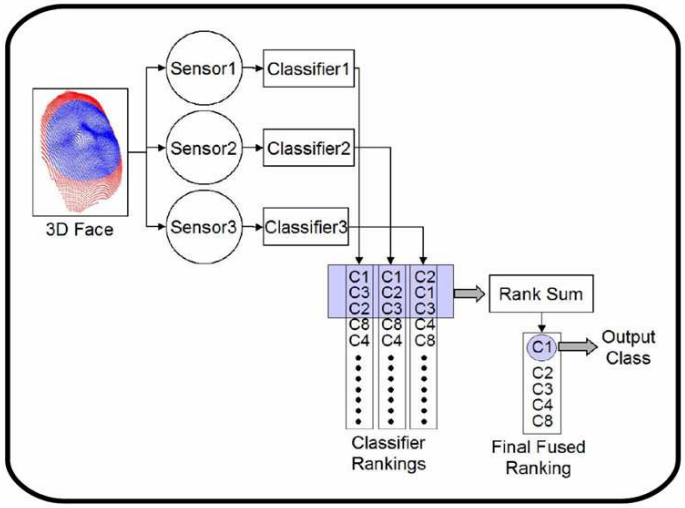
Training Examples: $x_{tr} = \{x_1, \ldots, x_{n_{tr}}\}$

Test Examples: $y_{ts} = \{y_1, \ldots, y_{n_{ts}}\}$

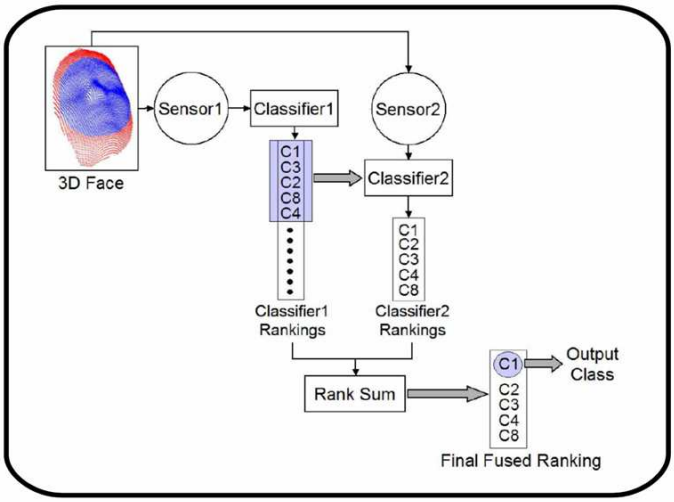
OUTPUT: Error rate : a

```
1   ConfidenceFusion(C, x_tr, y_ts)
2       Initialize error: err = 0
3       For each test example y_i, i = {1, ..., n_ts}
4           For each classifier C_j, j = {1, ..., L}
5               Compute the nearest class label, k_j and confidence value, w_j
6           /* Highest Confidence Rule */
7           Select the class label having the highest confidence:
            c = k_j where arg max_j w_j
8           /* Modified Plurality Voting */
9           If (No ties)
10              c = k_j /* most frequent class label */
11          Elseif
12              Compute average confidences of equiprobable classes: w_j^e
13              Select the class having highest average confidence:
                c = k_j where arg max_j w_j^e
14          If c ≠ the class ID of y_i
15              err = err + 1
16      Return: Error rate: a = err / n_ts
```

Figure 6.2. Pseudocode of the confidence-aided fusion schemes: If the highest confidence rule is employed, lines 6-7 is executed. If the modified plurality rule is employed, lines 8-13 is executed.



(a)



(b)

Figure 6.3. (a) Illustrative example of *parallel fusion* using rank-level combination scheme, (b) Illustrative example of *cascaded fusion* using rank-level combination scheme. Classifier2 performs LDA on the training samples of the forwarded classes.

ALGORITHM: Two-Stage Cascaded Fusion

FUNCTION: Performs two-stage cascaded fusion for the identification scenario.

INPUT:

Classifiers: The first classifier, C_f and the second classifier, C_s

Training Examples: $x_{tr} = \{x_1, \ldots, x_{n_{tr}}\}$

Test Examples: $y_{ts} = \{y_1, \ldots, y_{n_{ts}}\}$

Threshold to determine whether to use C_s or not: $thres$

Candidate class list size: s

OUTPUT: Error rate : a

1 CASCADEDFUSION$(C_f, C_s, x_{tr}, y_{ts}, thres, s)$

2 Initialize error: $err = 0$

3 **For** each test example y_i, $i = \{1, \ldots, n_{ts}\}$

4 Obtain ranked list $r = \{r_1, \ldots, r_s\}$ of similar classes using C_f and compute the confidence w for the top class r_1.

5 **If** $w > thres$ */* Then do not consult C_s */*

6 $c = r_1$ */* Output the top class found by C_f */*

7 **Elseif** */*Consult C_s */*

8 Apply LDA to the samples of classes in r, construct LDA transformation: Λ

9 Represent y_i using LDA: $y_i^\Lambda = \Lambda(y_i)$

10 Use y_i^Λ and obtain new ranked list $r^\Lambda = \{r_1^\Lambda, \ldots, r_m^\Lambda\}$ using C_s

11 Fuse two ranked lists (r, r^Λ) by Borda Count method, select the top class as c

12 **If** $c \neq$ the class ID of y_i

13 $err = err + 1$

14 RETURN: Error rate: $a = \frac{err}{n_{ts}}$

Figure 6.4. Pseudocode of the *forward-if-unconfident fusion* scheme: Note that the second classifier C_s may use another representation other than y_{ts}. For example, C_f may use point cloud features as y_{ts}, and C_s may use depth images to construct LDA transformation Λ.

7. Experimental Results

In this chapter, we provide the results of the identification and verification experiments performed on two different databases: the 3DRMA, and the `UND` (also called the `FRGC v1.0`). In Section 7.1, experimental analysis mainly focuses on the effect of two different registration methods: the one which allows rigid transformation and the one which allows non-rigid transformations. For both of these approaches, various face representation schemes, and their corresponding classifiers are presented. Integration of different 3D face classifiers is also attempted. All of the experimental simulations presented in Section 7.1 use the 3DRMA face database.

According to the conclusions drawn from the experiments in Section 7.1, we continue to develop more sophisticated fusion schemes in Section 7.2. The main focuses of Section 7.2 are i) to evaluate various face representation and facial feature extraction algorithms, and ii) to evaluate their contribution in rich ensembles. For this purpose, a wide range of different face classifiers were realized. The `UND` face database which includes texture information is used throughout in Section 7.2.

7.1. The Effects of Registration and Representation on the 3DRMA

7.1.1. Comparison of 3D face classifiers

In our experiments, we have used the 3DRMA dataset [40]. Specifically, a subset of the automatically prepared faces were used in experiments. The subset consists of 106 subjects each having five or six shots. The data is obtained with a stereo vision assisted structured light system. Due to errors in the acquisition steps, some of the faces contain significant amount of noise. There are slight pose variations in the 3DRMA for left/right and up/down directions. Although faces are generally neutral, some subjects have smiling expressions. Some artifacts such as glasses, hair, beard and moustache are also present. All of the faces have almost uniform scale. Sample depth images from the 3DRMA are shown in Figure 7.1. On the average, faces contain about 4000 3D

points. After preprocessing steps, faces contain 2,239 3D points in RegTPS, and 3,389 points in RegICP. The resolution of depth images in these experiments are 99×77. In profile set representation, each profile curve contains 220 points approximately, and the total number of profile points in seven curves is 1,557.

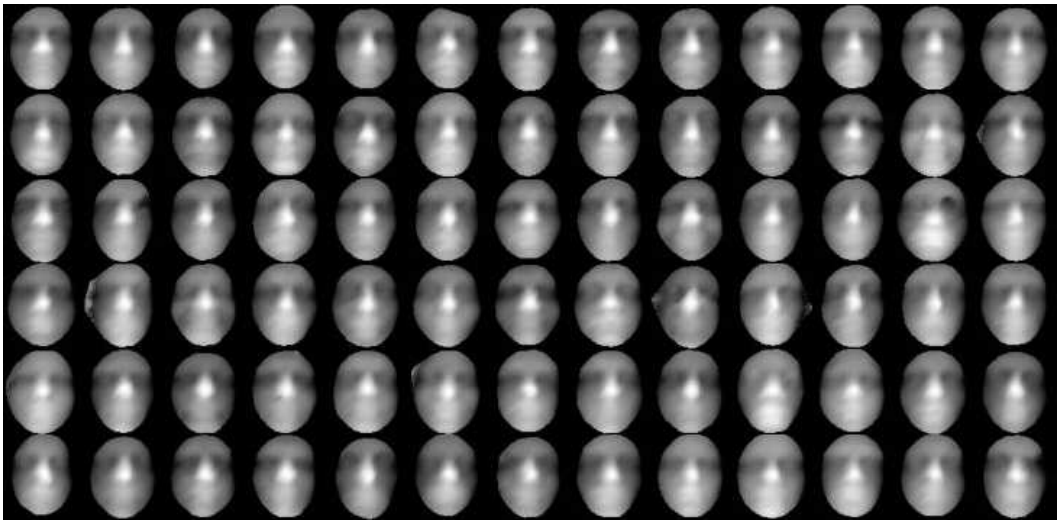


Figure 7.1. Sample depth images from the 3DRMA database.

In the experiments, we present the classification accuracies of the algorithms that are based on i) different 3D face representations and ii) two registration methods: RegTPS and RegICP. The acronyms of the algorithms are shown in Table 7.1. The names of the 3D face recognition algorithms which operate on ICP-based registered faces start with `WN` and the ones which use TPS registered faces start with `WY`. We form four different experimental setups with different number of shots in their respective training sets. In the first experiment, E_1, only one shot for each person is put into the training set, and in E_2, E_3 and E_4, there are two, three and four shots per person in the training set, respectively. For each experimental setup, we perform k-fold runs and report the mean accuracies and standard deviations. Each fold represents a different combination of training and test set samples. In E_1, E_2, E_3, and E_4, the number of folds are 5, 10, 10, and 5, respectively.

Which facial features are best?: Upper part of Table 7.2 shows the classification accuracies of the different feature extraction methods for RegTPS for experiment E_4. Best performance is obtained using surface normal representation with 97.72 per

Table 7.1. Acronyms of the algorithms for the two registration algorithms RegTPS and RegICP.

RegTPS	RegICP
`WYPC`: point cloud with warping	`WNPC`: point cloud w/o warping
`WYSN`: surface normals with warping	`WNSN`: surface normals w/o warping
`WYSI`: shape indices with warping	`WNSI`: shape indices w/o warping
`WYPCA`: PCA of depth with warping	`WNPCA`: PCA of depth w/o warping
`WYLDA-DI`: LDA of depth with warping	`WNLDA-DI`: LDA of depth w/o warping
`WYPRO`: profile sets with warping	`WNPRO`: profile sets w/o warping

cent correct identification rate. Point cloud and shape-index representations have accurately identified 92.95 and 90.26 per cent of the test examples, respectively. In depth image-based statistical methods, PCA performs worst, whereas LDA performs significantly better than PCA. In general, depth image-based methods perform poorly when compared to other representation techniques. *Profile set representation* have outperformed depth image methods, by obtaining 81.15 per cent recognition accuracy. We have included the recognition performance of using only the central profile in Table 7.2 for comparative analysis. It is seen that using only the central profile reaches 60.48 per cent recognition accuracy, which is worse than using profile sets.

The lower part of Table 7.2 shows the identification rates for the RegICP method. As in the RegTPS, best recognition performance is obtained by surface normal representation with 99.17 per cent accuracy. Point cloud and Depth-LDA methods have obtained similar accuracies and Depth-PCA again performed the worst. The comparative analysis of the different facial feature extractors have shown that the direction of the surface normals carries more information than any other method. The most important contribution of this analysis is that surface normals are better descriptors than the 3D coordinates of the facial points. This contribution is significant because most of the 3D face recognizers proposed so far largely depend on the 3D coordinate information.

Table 7.2. The mean and the standard deviations of the classification rates of different features extracted from RegTPS- and RegICP-based registration methods for E_4.

RegTPS							
	WYPC	WYSN	WYSI	WYPCA	WYLDA	Central Profile	WYPRO
Mean	92.95	**97.72**	90.26	45.39	75.03	60.48	81.14
STD	1.01	0.46	2.21	2.15	2.87	3.78	2.09
RegICP							
	WNPC	WNSN	WNSI	WNPCA	WNLDA	Central Profile	WNPRO
MEAN	96.48	**99.17**	88.91	50.78	96.27	82.49	94.51
STD	2.02	0.87	1.07	1.10	0.93	1.34	1.89

Which face registration technique is better?: When we compare the registration techniques RegTPS and RegICP, we see the advantage of registering faces without warping. The accuracies of the best two feature extractors, namely surface normals and point clouds, are improved by 1.45, and 3.53 per cent, respectively. Another important point is that the performance of the Depth-LDA method significantly improves from 75.03 per cent to 96.27 per cent when the warping is not carried out. A similar improvement is also present in using profiles. These results confirm that producing shape-free faces with warping in 3D loses the discriminative shape information in faces. Therefore, it is useful to extract facial features from the original faces.

The Effect of Training Set Size on the Recognition Performance: Up to now, we have analyzed the performance characteristics of all algorithms in experiment E_4 which contains four training samples per subject. However, it is crucial to observe the identification power under more realistic conditions where fewer samples are present in the training set. For this purpose, we have compared the algorithms in experimental configurations from E_1 to E_4. Note that the index i in experiment E_i denotes the number of training patterns in the training set. Table 7.3 shows the average recognition accuracies for all k-folds in each experiment. We have added (WNLDA-SN) method for the RegICP-based registration method, and removed PCA-based methods in Table 7.3. In all of the four experiments, surface normal-based features in RegICP (WNSN and

WNLDA-SN) obtained the best accuracies. An interesting outcome of these experiments is that even when using a single training shot, WNSN obtains a very good classification accuracy, 90.14 per cent. The superiority of the features extracted from the RegICP-based registration algorithm is more visible in E_1 and E_2 . This observation supports the advantage of using RegICP when the recognition task becomes harder. The best face recognizer in E_4, WNLDA-SN, obtains 99.79 per cent average accuracy in five folds of E_4. There are a total of 193 test instances in each fold of E_4. This corresponds to misclassifying approximately two test instances out of $5 \times 193 = 965$ test instances. Figure 7.2 shows some of the misclassified faces of the WNSN classifier. There are six face image pairs. At each pair, left image is the query face and the right face is the nearest face that is found by the classifier. Note that although the identities are different, query faces are very similar to the found faces, i.e., smiling faces, faces with large or narrow noses. It is also noticed that mistakes in the preprocessing stage, such as incorrect cropping, or noisy data cause misclassifications.

Table 7.3. Mean and standard deviations of the recognition accuracies for four experiments.

	E_1	E_2	E_3	E_4
WNPC	86.03 (2.75)	93.43 (3.05)	95.42 (2.90)	96.48 (2.02)
WNSN	**90.14** (3.47)	**96.72** (2.42)	98.33 (1.60)	99.17 (0.87)
WNSI	69.47 (2.98)	81.65 (3.63)	86.69 (2.30)	88.91 (1.07)
WNLDA-DI	N/A	74.84 (3.30)	93.58 (2.05)	96.27 (0.93)
WNPRO	78.90 (4.07)	89.01 (3.57)	92.71 (2.92)	94.51 (1.89)
WNLDA-SN	N/A	95.46 (1.24)	**99.46** (0.48)	**99.79** (0.28)
WYPC	72.56 (2.38)	84.42 (3.58)	89.50 (2.94)	92.95 (1.01)
WYSN	84.85 (3.74)	94.47 (1.93)	96.89 (1.62)	97.72 (0.46)
WYSI	70.02 (4.26)	82.00 (3.81)	87.36 (2.54)	90.57 (2.52)
WYLDA	N/A	47.48 (3.28)	63.51 (3.76)	70.67 (2.06)
WYPRO	58.67 (3.30)	70.81 (3.31)	77.29 (3.58)	81.14 (2.09)

Face Authentication Experiments: In this section, we have performed authentication experiments on the 3DRMA dataset. In these experiments, a single shot

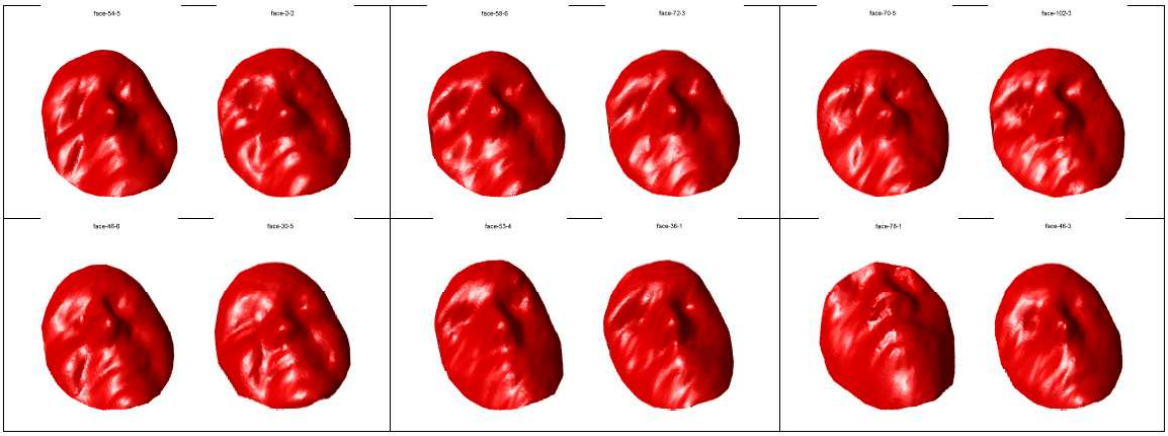


Figure 7.2. Misclassified faces of the WNSN algorithm. For each image pair, left image is the query face, and the right image is the nearest face found by the WNSN algorithm.

for each subject is used in the enrollment phase, i.e., experiment E_1. In order to obtain *receiver operator characteristics* (ROC) curves and to compute *equal error rates* (EER), a varying threshold value t is used for the decision. Figure 7.3 shows eight ROC curves for the algorithms: WNPC, WNSN, WNSI, WNPRO, WYPC, WYSN, WYSI, and WYPRO. The best authentication results are obtained by WNSN which is the closest curve to the origin. The worst performance is obtained by WYPRO. Equal Error Rates (EER) for warping-based classifiers are: WYPC = 12.56, WYSN = 12.36, WYSI = 16.59, and WYPRO = 17.02 per cent. When you compare these EER's with the ones in the second column (E_1) of Table 7.4 for RegICP, it is seen that ICP-based authenticators perform significantly better. Table 7.4 shows authentication performances of RegICP-based classifiers for all four experiments. Best authenticators WNSN and WNPC have 8.06 and 8.36 per cent EERs in E_1, respectively. As in the recognition experiments, if you have enough samples for the WNLDA-SN method, it outperforms other methods: for experiment E_4 in Table 7.4, the WNLDA-SN method attained 0.72 per cent EER which is the best EER among all classifiers.

Table 7.5 shows the comparative results for both authentication and recognition algorithms on the 3DRMA database which were proposed in the literature so far. In [13], comparative analysis of several approaches on the 3DRMA dataset were presented. We only include the best of them in Table 7.5. The 3DRMA dataset contains six shots

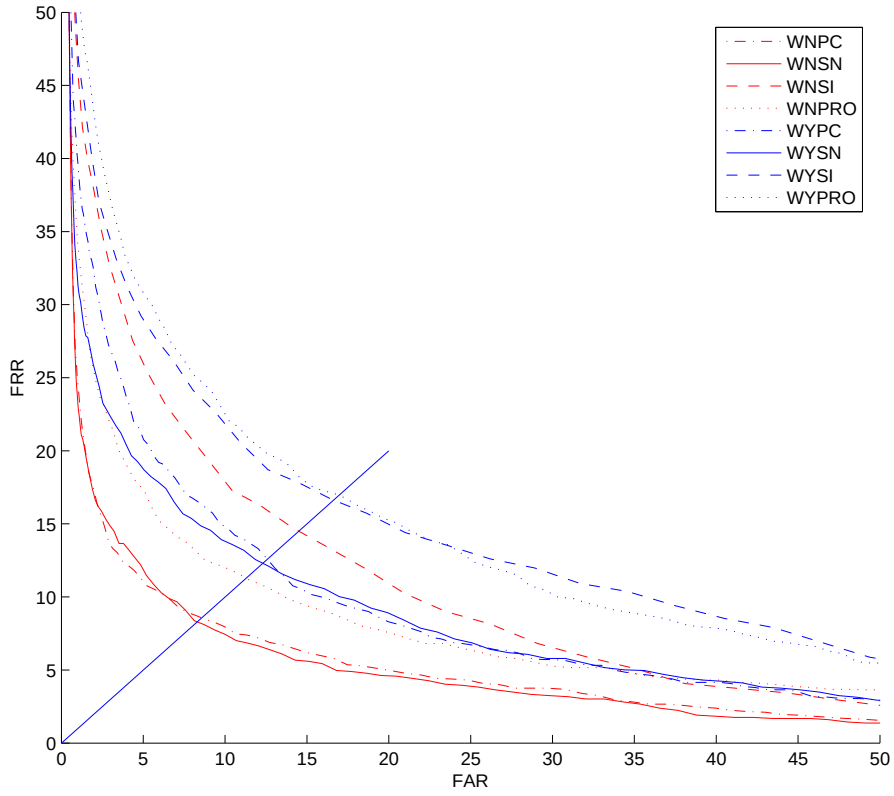


Figure 7.3. ROC curves for each method in experiment E_1. $x-$axis and $y-$axis denote false acceptance and rejection rates, respectively.

per each of the 120 subjects. However, a significant portion of the faces contains a huge amount of noise. In order to reveal the true performance of the proposed algorithms, a subset of these images are generally used which makes it harder to compare algorithms. In addition, the number of training samples were not provided in [13] for authentication experiments, so we include both `WNSN` and `WNLDA-SN` with one and four training samples in the Table 7.5, respectively. In the third column of Table 7.5 we provide the number of subjects selected for each experimental configuration, and provide EER and correct classification rates (CCR) for authentication and recognition experiments in the fourth column, respectively.

Table 7.4. Authentication rates (equal error rates) for RegICP-based classifiers for four experimental configurations. Best accuracies are typed in boldface.

Method	E_1	E_2	E_3	E_4
WNPC	8.36	4.48	3.14	2.35
WNSN	**8.06**	4.36	3.29	2.92
WNSI	14.19	8.00	5.62	4.20
WNPRO	10.08	5.82	4.56	4.07
WNLDA-DI	N/A	10.20	3.74	2.85
WNLDA-SN	N/A	**3.81**	**1.38**	**0.72**

7.1.2. Fusion Experiments on the 3DRMA Database

In this section, we provide the experimental results of the fusion techniques that are presented in 6. The following methods are analyzed: Plurality voting (CV), Borda count(BC), product rule(MMP), modified plurality voting(ICV), highest confidence fusion(HC), forward-always(FA), forward-if-not-confident(FC). For min-max normalization, the minimum and maximum score values are calculated by the intra-class score distributions of the training set. In ICV,HC, and FC fusion techniques, each classifier should output the nearest class label together with its confidence value. The estimation confidence values were presented in Chapter 6.

In the fusion experiments, we have used all the individual pattern recognizers that are based on RegICP. From now on, they will be referred to as C_i's where C_1 = WNPC, C_2 = WNSN, C_3 = WNSI, C_4 = WNPRO, C_5 = WNLDA-DI, and C_6 = WNLDA-SN. Among them, C_2 and C_6 are the best ones in terms of classification accuracy. We report the performance figures of the combined classifiers in three experiments: E_2, E_3, and E_4. Remember that E_j means that there are j training patterns for each class, and the rest is put into the test set. We have constructed a number of distinct ensembles from six pattern classifiers. These ensemble configurations are all combinations of six classifiers with the size of six, five, four, and three: for instance ensemble Ω_1 fuses classifiers $\{C_1, C_2, C_3, C_4, C_5, C_6\}$, $\Omega_2 = \{C_2, C_3, C_4, C_5, C_6\}$, $\Omega_3 = \{C_1, C_3, C_4, C_5, C_6\}$,...,

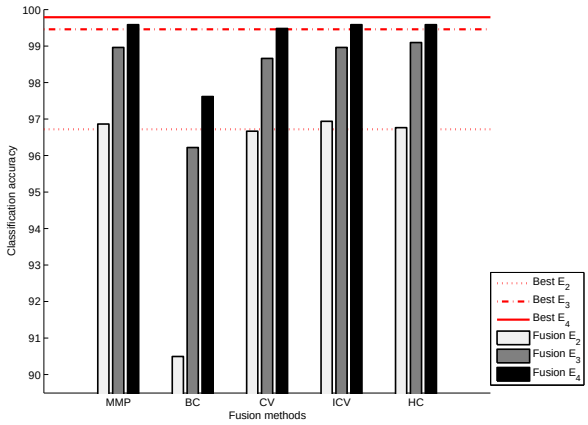
Table 7.5. Comparison of face authentication and recognition results on the 3DRMA database.

Face Authentication Experiments		
Algorithm	Experimental Configuration	EER
Facial profile (Beumier [39])	30 subjects	9.50
SDM (Pan et al. [13])	30 subjects	6.67
SDM (Pan et al. [13])	120 subjects	8.79
`WNSN`	106 subjects, one training sample	8.06
`WNLDA-SN`	106 subjects, four training samples	0.72
Face Recognition Experiments		
Algorithm	Experimental Configuration	CCR(%)
3D Eigenfaces (Xu et al. [97])	120 subjects, five training sample	71.1
3D Eigenfaces (Xu et al. [97])	91 subjects, five training sample	80.2
`WNLDA-SN`	106 subjects, four training sample	99.8

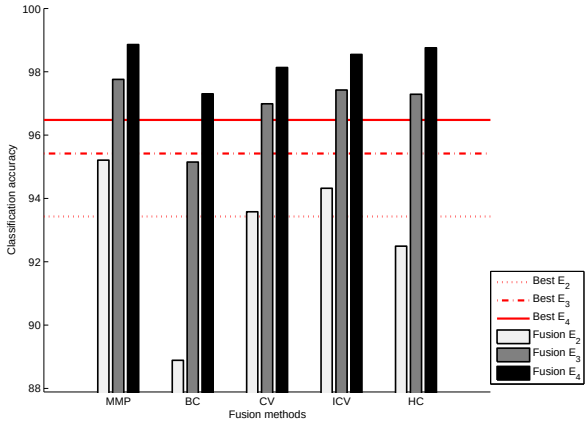
$\Omega_{41} = \{C_1, C_2, C_4\}$, and $\Omega_{42} = \{C_1, C_2, C_3\}$. There are a total of 42 distinct ensemble configurations. Ensemble Ω_1 deserves a special attention since it uses all pattern classifiers. Figure 7.4.a shows the classification performances of all parallel fusion algorithms for Ω_1 for the experiments E_2, E_3 and E_4. For each fusion method, white, gray, and black bars denote the correct classification rates for E_2, E_3 and E_4, respectively.

Which fusion algorithms are good?: By looking at Figure 7.4.a, we see that MMP and ICV attain similar accuracies and they are the best fusion techniques. Borda count is the worst fusion method in Ω_1. However comparison of the fusion algorithms for just ensemble Ω_1 may not be valid. Therefore, in Figure 7.5.a, we show the average performances of all fusion algorithms in all ensemble configurations from Ω_1 to Ω_{42}. Figure 7.5.a makes clear that MMP, ICV, and HC are top performers.

When is the fusion useful? Figure 7.4.a contains horizontal lines, which represent the best individual pattern classifier's performance in the ensemble Ω_1. The horizontal dotted line represents the classification rate of the best individual classifier



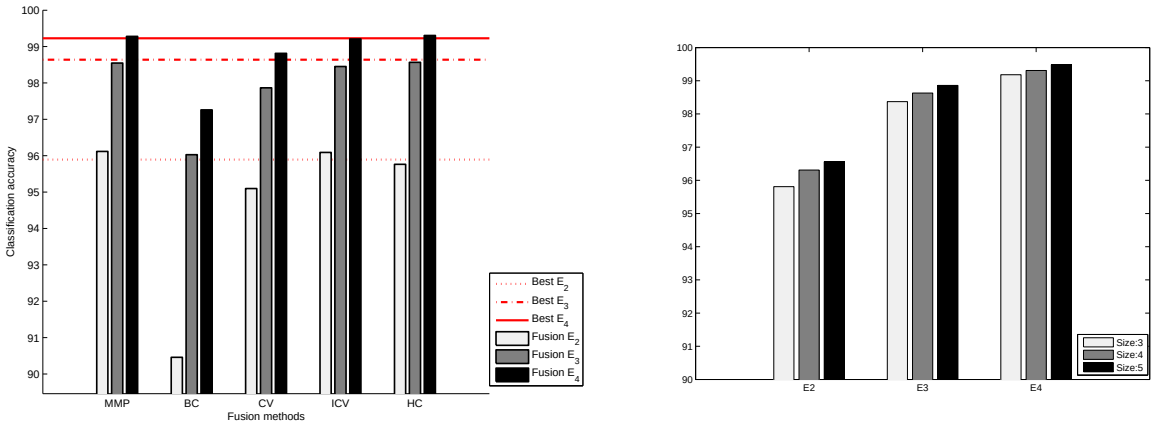
(a)



(b)

Figure 7.4. (a) Recognition accuracies of the ensemble architecture which consists of all individual face recognizers, and (b) Recognition accuracies of the ensemble architecture which consists of WNPC, WNSI, WNPRO, WNLDA.

(See Table 7.3: `WNSN` = 96.72 per cent) for experiment E_2. Similarly, dashed and solid lines denote best individual accuracies for E_3 and E_4, respectively. Generally, we expect the performance of the ensemble architecture to be superior to the performance of the best individual classifier. However, in experiments E_3 and E_4, none of the fusion methods improve the single best accuracies. Only the best fusion methods improve the classification rates in E_2 (See white bars over the horizontal dotted line). This behavior is due to the so-called *ceiling effect* in these experiments. E_3 and E_4 are relatively easy problems when compared to E_2, and there are two very strong individual classifiers in the ensemble, namely, C_2 = `WNSN`, and C_6 = `WNLDA-SN`. For instance, C_6 obtains 99.46 and 99.79 per cent recognition accuracies in E_3 and E_4. These performance figures are



(a) (b)

Figure 7.5. (a) Average recognition accuracies of all ensemble architectures, and (b) The average classification rates of different ensemble sizes for MMP fusion algorithm. White, gray, and black bars denote the ensemble sizes of three, four, and five, respectively.

very good in this dataset, and is very hard to get better results. In order to confirm this analysis, we have to look at the fusion performances where the ensemble does not contain these two strong classifiers C_2 and C_6. In this way, we can simulate a much harder task and expect better improvements. A good candidate is the ensemble Ω_{16} which contains C_1, C_3, C_4, and C_5. Fusion results for Ω_{16} are shown in Figure 7.4.b. These confirm our initial result that all of the better fusion schemes outperform the best individual rates significantly. For example, the MMP technique improves the best rates by 1.78, 2.34, and 2.38 per cent for E_2, E_3, and E_4, respectively. We conclude that if you have weak or moderate 3D face classifiers, then fusion is beneficial.

What should be the size of the ensemble?: An important question regarding the fusion performance is the number of pattern classifiers in the ensemble. Theoretically, it is highly probable to get better performance by combining fewer complementary classifiers than by combining many but similar classifiers. For this purpose, we have analyzed the number of 3D face classifiers that should be used in the ensembles. We have selected the MMP fusion algorithm for this experiment. We have six ensembles (Ω_2 to Ω_7) of size five, 15 ensembles (Ω_8 to Ω_{22}) of size four, and 20 ensembles (Ω_{23} to Ω_{42}) of size three. Figure 7.5.b shows the average ensemble accuracies for

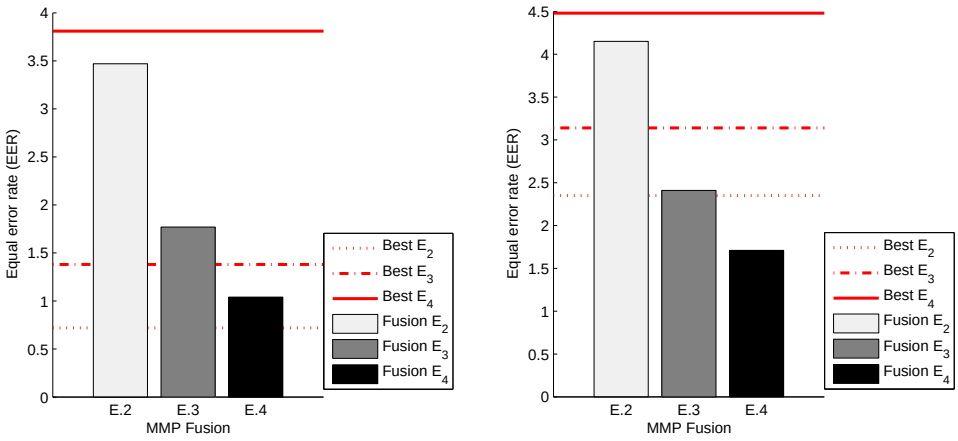
ensemble sizes three(white bars), four(gray bars) and five(black bars) for experiments E_2, E_3, and E_4. It is found that for all of the experiments, increasing the ensemble size improves the classification rate. This finding confirms the complementary behavior of the individual 3D face classifiers.

What is the performance of serial fusion?: We propose two serial fusion schemes: FA, and FC. In the realization of these algorithms, we use `WNSN` algorithm as a first classifier C_f, and LDA of surface normals as a second classifier C_s. In FA, given a test face, C_f sorts the classes in increasing order of distance scores, and passes $N = 10$ nearest class labels to C_s. N is found empirically using the training set such that the presence of the true class is highly probable in the first 10 classes. C_s then dynamically constructs an LDA subspace using the training examples of these 10 classes. As features, surface normals are used in C_s. Notice that C_s is different from `WNLDA-SN`. In `WNLDA-SN`, all classes are used to form the LDA subspace. On the other hand, in FC, given a test face, if the confidence of C_f is lower than the confidence threshold $\tau = 0.5$ for the nearest class found, then C_f passes the nearest 10 class labels to C_s. Otherwise, C_f just outputs the nearest class as a final decision. The average recognition accuracies of the FA algorithm for the experiments E_2, E_3, and E_4 are 91.65, 98.86, and 99.90 per cent respectively. The correct classification rate of FA in experiment E_2 is worse than other parallel fusion schemes. This is due to the fact that it is practically very hard to estimate the within class scatter using only two training examples per class. However, when you use enough training faces, as in E_3 and E_4, then the performance of FA quickly gets better, and even exceeds the accuracies of the other fusion schemes. The accuracy of 99.90 per cent in E_4 is the top performance obtained among all single classifiers and other fusion algorithms. This behavior is also observed in FC experiments. The average recognition accuracies of the FC algorithm for the experiments E_2, E_3, and E_4 are 93.14, 98.90, and 99.90 per cent respectively. The classification accuracies of FC fusion algorithm are better than FA. This validates the benefit of the use of confidence idea in fusion. It is also worth noting that FC is considerably faster than FA since it does not need to perform LDA for every test instance. The identification performances of the serial fusion schemes prove that if you have enough training samples, they can be better alternatives to parallel architectures,

and it is possible to improve best individual accuracy even in the presence of strong classifiers. Another interesting outcome of the serial fusion idea is that you can get these performance figures using a feature vector of size nine which is the maximum dimensionality in the LDA for a 10-class classifier C_s. This is worth noting since the dimensionality of the original 3D face recognition problem using surface normals is typically $\sim 3,389 \times 3$.

What are the authentication accuracies of the fusion algorithms?: We have also analyzed the benefits of decision-level fusion algorithms for the face authentication problem. Figure 7.6 depicts the authentication performance of the MMP algorithm on experiments E_2(white), E_3(gray), and E_4(black). y-axis shows the equal error rate. Similar to the recognition experiments, horizontal lines illustrates the best individual authentication accuracies in the ensembles. Note that the corresponding bars should be below the horizontal lines since these are error rates. In Figure 7.6.a, the accuracies of ensemble Ω_1 which contains all pattern classifiers from C_1 to C_6 are shown. In Figure 7.6.b, the accuracies of ensemble Ω_{16} which contains pattern classifiers C_1, C_3, C_4, and C_5 (excluding strong classifiers) are shown. Authentication results display a similar behavior as in the recognition experiments: if you have strong classifiers in the ensemble such as Ω_1, then the fusion algorithms can not perform better than the best individual accuracy. However, if you fuse moderate classifiers as in Ω_{16}, you can get significant accuracy improvements.

The effect of classifier weighting Up to now, we treat all individual classifiers equally and does not employ a weighting scheme in the fusion process. However, it is possible to weight the experts according to their individual performances. Basically, the motivation is to trust more to the decision of the better classifiers. Although this mechanism may generally result in better ensembles, automatic estimation of classifier weights from a limited amount of validation set which is usually the case in face recognition systems, may not produce optimal weights. Here, our aim is to show the identification behavior of the ensembles where the expert weights are automatically estimated from a separate validation set. For this purpose, we design a new experimental protocol for the 3DRMA database. In our simulations, we put three images per



(a) Ensemble Ω_1 (b) Ensemble Ω_{16}

Figure 7.6. The equal error rates for the face authentication experiments for the MMP fusion algorithm. White, gray and black bars denote the experiments E_2, E_3, and E_4, respectively.

subject into the training set, one image to validation set, and remaining images into the test set. We have formed four experimental folds where different combinations are put into the training and validation sets. In each fold there are a total of 318 gallery images, and 193 probe images. Validation set is used to determine classifier weights.

Table 7.6 presents the correct classification accuracies of the four individual 3D face classifiers for four experimental folds. Point cloud-based (PC) classifier attains the best performance with 91.19 per cent average accuracy. Facial profile (PRO) and depth image-based (LDA-DI) have similar rates, whereas shape index-based classifier (SI) performed worst.

In addition to the previously analyzed fusion methods, we implement two weighting-based schemes: *Weighted Sum Rule* (WS): In original sum rule, individual pattern classifiers are considered to be equally powerful. However, if some of them are more accurate, then it is useful to weight them in the fusion process. Let w_j's be the weights of classifiers computed on a separate validation set, then weighted sum rule is written as: $y_j = \sum_{k=1}^{K} w_j \times s_{kj}$. *Weighted Consensus Voting* (CV-W): Similar to the weighted sum

Table 7.6. 3D face classifier performances.

	PC	SI	PRO	LDA-DI
E_1	86.01	78.76	83.42	88.08
E_2	92.75	88.08	92.23	91.71
E_3	92.75	86.53	93.26	89.64
E_4	93.26	89.12	94.30	92.23
Mean	91.19	85.62	90.80	90.41

rule, individual classifier's strengths can be embedded into a consensus voting scheme where the counts of the top-ranked classes are increased by w_j's (in the original `CV`, we simply count each vote as one). Table 7.7 presents the identification accuracies of sum (`SUM`), weighted sum (`WS`), plurality voting (`CV`), weighted plurality voting (`CV-W`), improved plurality voting (`CV-A`), Borda count (`BC`), highest confidence (`HC`), product (`PROD`), and serial fusion (`SF`) methods.

We find that employing a weighting scheme in the sum rule (`WS`) degrades the classification rate. Let P_i be the accuracy of classifier C_i on the validation set, then weights w_i are computed by:

$$w_i = \frac{P_i}{\sum_{k=1}^{L} P_k} \tag{7.1}$$

We have also tried other techniques (found in [94]) for weight estimation, but this approach gave the best performance. The failure of weighting-based fusion can be explained by the insufficient amount of validation data. However, in practical systems, we generally do not have sufficient amount of training samples per subject. So, this situation will always be the limiting factor. The same degradation is also present in weighted consensus voting (`CV-W`).

Table 7.7. Fusion performances on the 3DRMA database.

	SUM	WS	CV	CV-W	CV-A	BC	HC	PROD	SF
E_1	92.23	91.71	88.60	87.05	91.19	86.53	91.19	92.23	94.30
E_2	96.37	96.89	95.34	94.82	95.86	94.82	96.37	96.89	98.96
E_3	96.89	96.37	95.86	95.86	94.82	94.82	96.37	96.37	99.48
E_4	96.37	94.82	96.89	95.86	95.34	94.82	96.37	95.86	98.96
MEAN	95.47	94.95	94.17	93.39	94.30	92.75	95.08	95.34	97.93

7.2. The Effects of Representation, Feature Extraction, and Ensemble Construction

Our findings in the previous section (Section 7.1) reveal that AFM-based rigid registration of faces is superior in terms of classification accuracy to more elaborate TPS warping-based registration algorithm. Therefore, in this section we focus on the ICP-based registration method. As opposed to the previous section, our aim in this section is to concentrate on the benefits of classifier combination techniques where a vast amount of individual face experts are present. The construction of face experts is divided into two phases: first the representation of 3D facial images is considered. In this context, *the representation* means the way 3D face signals are stored, i.e., as point clouds, or depth images. The second phase is related to the feature extraction techniques used for a specific type of face representation method. For instance, given a texture-based (intensity image) representation, PCA and 2D Gabor wavelet can be considered as two different feature extractors. Combining different representation and feature extraction methods yields a large number of face experts. Complete list of these experts are provided in Table 7.8, and we follow the notation presented in that table. The second part of the experimental studies presented in this section is devoted to constructing useful combination of individual face classifiers. To this end, various ensemble construction methods are presented and compared.

7.2.1. Summary of representation and feature extraction methods

Table 7.8 shows all of the representation techniques, and their corresponding feature extraction methods. For each method, we provide feature dimensionality and the distance measure used. Each expert relies on a different feature extraction method, and thus has different input feature vector size. Note the wide disparity in feature dimensionality: For example, transform-domain features have compression ratios of 1:350 vis-à-vis the raw point cloud data. Moreover, a distance measure appropriate for each feature type was determined experimentally from the L1, L2 and COS set as already given in Table 7.8. For the multidimensional features, i.e., surface normals, principal directions, and point cloud coordinates, the distances are simply calculated for each dimension separately and then summed. For these cases, the distance measure has the additional symbol $\sum$. One exception is the CURV-PD method, since this feature consists of two 3-vectors. Therefore, the distance is calculated by the sum of two 3-vector differences. After the preprocessing operations of alignment and cropping, the original 3D point clouds with varying number of samples are reduced to a fixed number of registered 16,560 points and correspondences are established. Similarly, depth and texture image resolutions are 281×321. Note that the original face texture image containing both face and non-face regions was 480×640. After cropping the face region, the new image size becomes 281×321.

7.2.2. Face Database and Experimental Protocols

For the recognition tests, we have used the University of Notre Dame (UND) 3D face database [98], also known as the Face Recognition Grand Challenge (FRGC) v1.0 database in the literature. The original UND database contains 943 3D scans of 275 subjects. We had to use a subset of the original database, since 75 subjects had only one scan, and 14 3D scans were badly registered with the texture data. Thus, the part of the database involved in our experiments contained 854 2D and 3D scans of 195 subjects. Each subject had at least two, and at most eight 3D scans. The UND database consists mostly of frontal faces and does not exhibit significant expression variations. However, some scans have slight in-depth pose variations, and different expressions. Texture

Table 7.8. Representations, features, dimensionalities, and distance measures for face experts. Registered faces contain 16,560 3D points.

Representations	Features	Acronym	Dim.	Distance M.
Point Clouds	(x,y,z) coordinates	PC-XYZ	49,680	$\sum L2$
	NMF coefficients	PC-NMF	90	COS
	ICA coefficients	PC-ICA	90	COS
Surface Normals	(nx,ny,nz) unit normals	SN	49,680	$\sum L2$
Depth Images	Pixels	DI-PIXEL	90,201	L2
	DCT coefficients	DI-DCT	49	COS
	DFT coefficients	DI-DFT	49	COS
	ICA coefficients	DI-ICA	80	COS
	NMF coefficients	DI-NMF	70	COS
Curvature	Shape-index (SI)	CURV-SI	16,560	L1
	Principal directions	CURV-PD	99,360	$\sum(COS + COS)$
	Mean curvature	CURV-H	16,560	L1
	Gaussian curvature	CURV-K	16,560	L1
Voxel	3D DFT coefficients	VOXEL-DFT	53	COS
Texture Images	Pixels	TEX-PIXEL	90,201	L2
	2D Gabor wavelets	TEX-GABOR	35,480	L1

information is stored as RGB values with 480×640 resolution. Shape data contains approximately 30,000 - 40,000 3D coordinates. Although the quality of the scanned data is high, there are two types of noise affecting 3D faces: small protrusions and impulse noise-like jumps. Median filtering is first used to remove impulse noise, and then mean filtering is applied to smooth the facial surface. Figure 7.7 shows sample faces from the UND database.

We have designed four different experimental configurations, as shown in Table 7.9. Each configuration contains a different number of training samples per subject. The subscript i in experiment E_i denotes the number of training samples per subject in that experiment. The reason for different populations is that in the UND database, 195 subjects have more than two 3D scans, 164 subjects have more than three scans

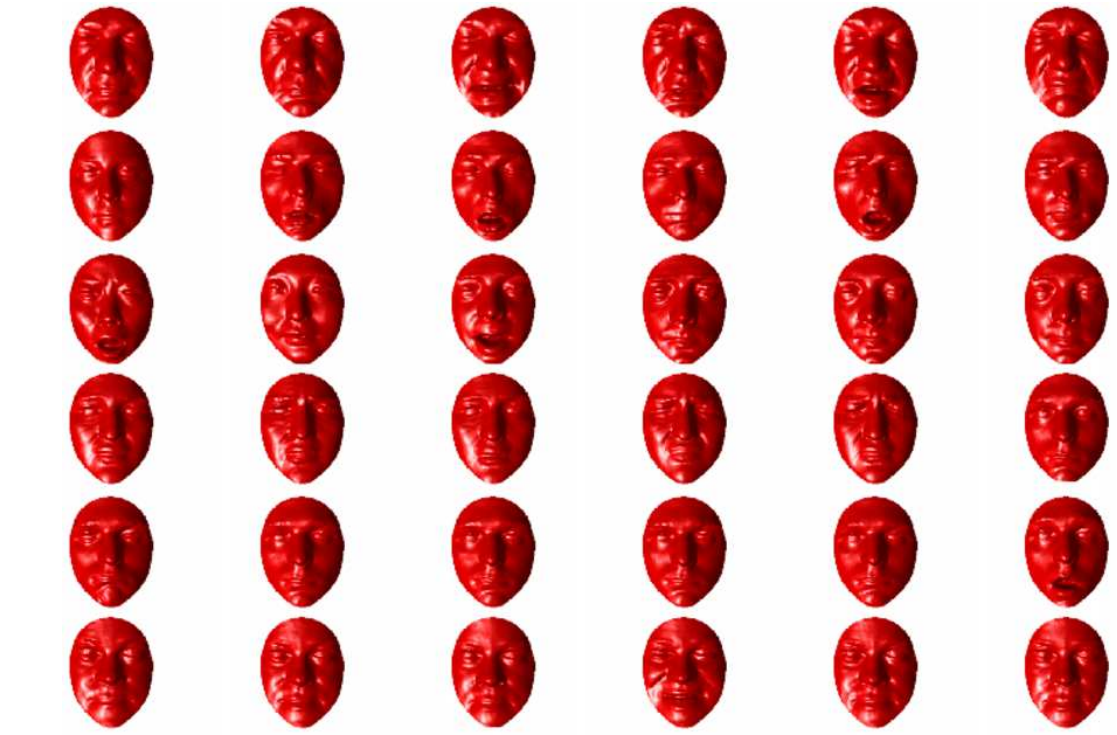


Figure 7.7. Sample faces from the FRGC database.

etc. Thus E_1 is designed so that every subject possesses only one image in the training set, and while the rest of $854 - 195 = 659$ images are placed in the test set. For each experiment, we have run several folds, and the number of folds for each experiment is shown in the last column of Table 7.9. In the rest of the this report, we report only the average of the recognition accuracies of the folds. The most difficult experiment is obviously E_1 (single gallery experiment) since not only there exists a single training image per person, but also both the enrollment size and the number of test scans are larger. Conversely, the easiest experiment is E_4, since it contains four training images per person and the test size is smaller. We choose not to report the even easier identification experiments, such as E_5, and E_6, since they are not sufficiently challenging. Note that when the number of images in the training set increases, the number of subjects that participate in that experiment decreases.

Table 7.9. Experimental configurations.

	Training samples per subject	Number of Subjects	Total training scans	Total test scans	Fold count
E_1	1	195	195	659	2
E_2	2	164	328	464	3
E_3	3	118	354	300	4
E_4	4	85	340	182	5

7.2.3. Comparative Analysis of Individual Face Experts

Table 7.10 shows the rank-1 correct classification rates of the face experts for the four experimental setups, where the boldface figures denote the three top competitors in that category. While it may not be correct to generalize these results out of the UND database case, nevertheless we find useful to state the following comments:

- Not surprisingly, there is a jump difference in performance between single gallery case and the experiments with at least two training images per subject. In fact, almost half of face experts attain nearly perfect classification whenever at least two training face samples are provided.
- For the single gallery experiment E_1, the top three experts are all related to surface curvature (CURV-PD, CURV-SI, SN). We can consider the surface normals as a different form of curvature-related descriptor. In the multi-gallery experiments (E_2, E_3, and E_4), subspace techniques PC-NMF, PC-ICA, and DI-DCT outperform others. This shift from surface experts to subspace experts as more data becomes available is intriguing. For example, PC-NMF which had the perfect score in E_3 falls to a mediocre position in E_1 with a score of only 85%. We conjecture that the subspace techniques achieve their full potential when adequate training data is supplied to construct their feature subspaces. The subspace techniques need more training samples to model the within class variability through the analysis into basis faces and the corresponding coefficients.
- One important observation is that the discrimination abilities of surface-based

descriptors i.e., CURV-PD, CURV-SI, SN, and PC-XYZ are better than others. Another observation is that the 3D directions of the minimum and maximum curvatures carry better discriminatory information as compared to scalar version of the curvature information, that is, the mean or Gaussian curvatures. For example, principal directions-based CURV-PD expert improves the identification accuracy by almost 2 per cent when compared to shape index-based CURV-SI classifier.

- An in-depth analysis of the performance scores in Table 7.10 reveals that face recognition experts using similar face representation methods achieve similar scores. Thus, it is not the feature type that is the determining factor, but the underlying information representation. In fact, we may group the face experts in decreasing order as follows: curvature-based, point cloud-based, depth image-based, texture-based, and voxel-based. Once the representation type is chosen, the performance variations due to features become relatively small. Hence, one should shift one's focus from choice of feature to the choice of representation. To give an example, consider ICA- versus NMF-based features for experiment E_1. The depth image-based classifiers DI-ICA and DI-NMF obtain 72 per cent average performance rate. On the other hand with the point cloud representation, PC-ICA and PC-NMF achieve 85 per cent average recognition rate. In any case, the feature extraction methods and the face representation methods should be considered together.
- In terms of the usefulness of shape and texture modalities, we observe the clear superiority of shape-based face classifiers. While we have implemented both eigenface-based and Gabor wavelet-based 2D recognition algorithms, we only provide the Gabor case since eigenface-based method is worse than both pixel and Gabor-based texture classifiers. However, even the best 2D texture-based Gabor method can only attain 74.73 per cent correct classification rate in E_1.

We also analyze the identification behavior of the individual face experts in a retrieval setting, and perform Rank-k experiments. From this analysis, we obtain cumulative match characteristics (CMC) curves. Figure 7.8 shows five CMC curves for the face experts for experiment E_1: PC-XYZ, CURV-PD, TEX-GABOR, DI-DCT, and

Table 7.10. Rank-1 correct classification accuracies of the face experts.

	E_1	E_2	E_3	E_4
PC-XYZ	87.71	94.68	97.92	98.90
PC-NMF	85.13	**97.77**	**99.25**	**100.00**
PC-ICA	85.66	**98.71**	**99.67**	**99.89**
SN	**89.07**	96.84	98.92	99.45
DI-PIXEL	55.99	70.19	79.75	87.69
DI-DCT	78.53	**97.63**	**99.58**	**99.78**
DI-DFT	75.95	97.13	99.08	99.56
DI-ICA	72.46	96.55	98.92	99.01
DI-NMF	71.55	95.83	98.67	99.67
CURV-SI	**90.06**	96.55	98.67	99.34
CURV-PD	**91.88**	97.13	99.08	99.45
CURV-H	87.41	95.69	98.50	98.90
CURV-K	84.37	93.89	97.25	98.46
VOXEL-DFT	64.26	91.16	97.92	99.34
TEX-PIXEL	64.04	77.16	84.33	92.53
TEX-GABOR	74.73	87.36	91.92	96.26

PC-ICA. From the CMC curves, it is seen that Rank-1 classification rates presented in Table III are correlated with the CMC performances of the face experts.

Figure 7.9 shows four sample face images misclassified by all of the 16 face experts in experiment E_1. Blue face (lighter) is the gallery image and the red (darker) face is the misclassified probe image. Errors generally stem from incorrect registration of faces. Pose variations in both vertical and horizontal axes are visible in the first and third images. Another source of error is especially visible in the forehead regions due to the presence of hair (see the fourth, and the first image).

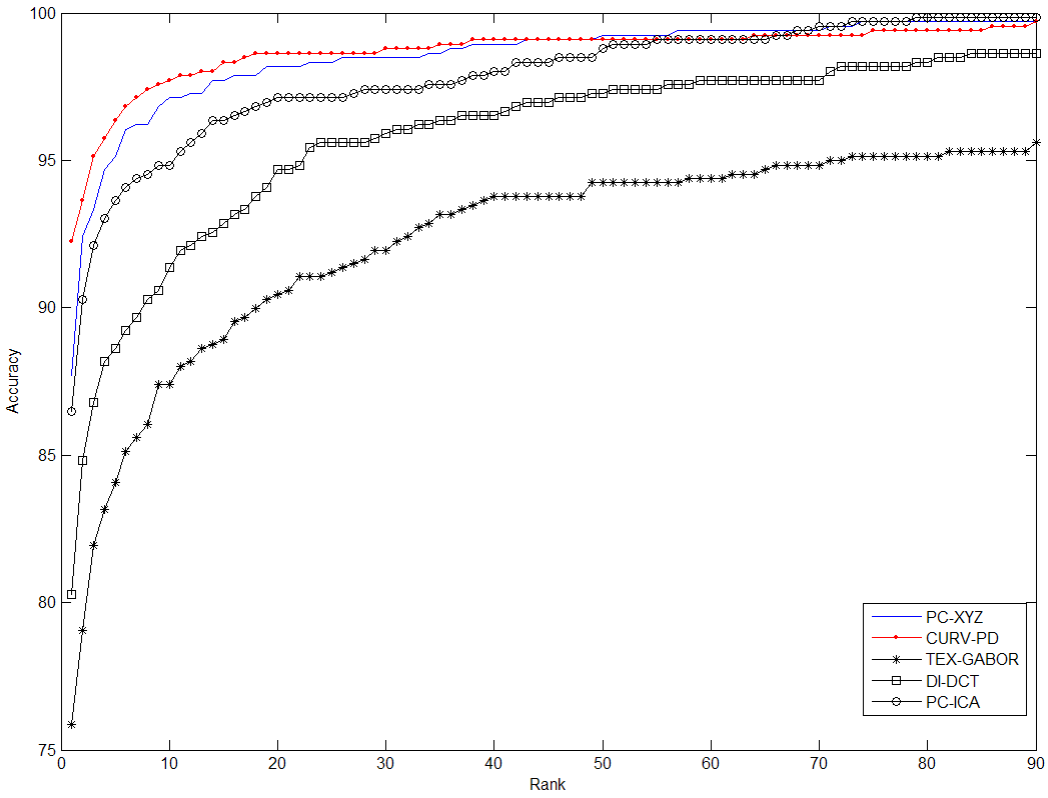


Figure 7.8. CMC curves of five face experts for E_1.

7.2.4. Comparative Analysis of Fusion Methods

In this section, we discuss the impact of the decision fusion methods in improving the recognition performance. Table 7.11 presents the rank-1 correct classification accuracies of the sum, product, plurality, highest confidence, modified plurality, Borda count, min, max, and median fusion techniques for the four experimental configurations. In each fusion method, all of the 16 base face experts listed in Table 7.10, are combined. In the columns of Table 7.11, the numbers in parentheses denote the classification rate improvement (or loss) with respect to the best individual face expert in that experimental category. For example, in experiment E_1, the best face expert (CURV-PD) obtains 91.88 percent classification rate, and all improvement figures in the E_1 column of Table 7.11 are calculated with respect to this base. To facilitate comparisons, these base expert accuracies are shown in the second row of the table with legend "Best Individual". For each experiment, the best fusion accuracies are highlighted in boldface. As expected, the fusion gains diminish from harder toward

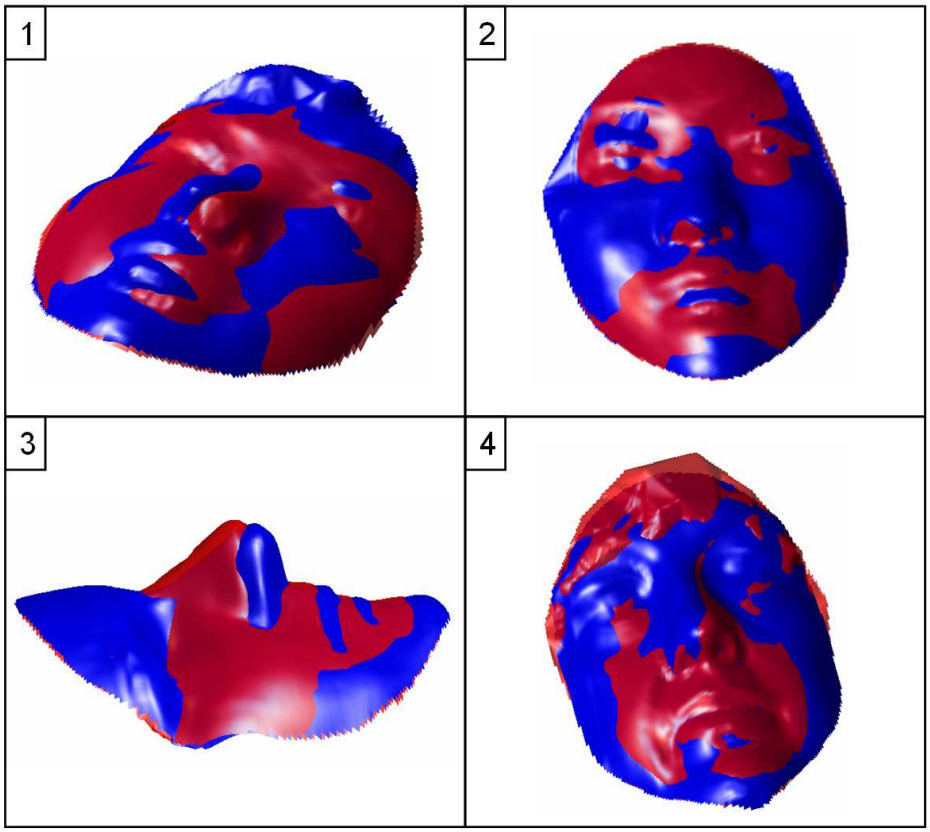


Figure 7.9. Misclassified faces in experiment E_1. 1) Pan variation, 2) Incorrectly normalized faces, 3) Tilt variation, and 4) Errors due to hair region.

simpler experiments, i.e., from E_1 to E_4. Therefore, it makes sense to analyze the advantage of fusion methods especially for the most challenging experiment, E_1.

One can observe generally that fusion may cause significant classifier performance losses with respect to the best expert's, if the fusion method is not correctly chosen. On the other hand, the contribution of fusion methods remains modest, even in the most difficult experiment. More specifically, the sum rule, which is the most widely used fusion technique in the 3D face community, reports a slight improvement of 0.15 per cent. The product rule performs very badly in E_1, but contributes positively in other experiments, pointing out again to the singularity of single-gallery experiment. This performance degradation in E_1 is due to the insufficient amount of training samples to estimate the score range with the min-max technique. If the training set is small, the estimated normalization parameters do not generalize well in the identification phase. This problem does not occur in experiments E2, E3, and E4 where sufficient training

data exists.

Table 7.11. Rank-1 correct classification accuracies of the fusion methods.

	E_1	E_2	E_3	E_4
Best Individual	91.88	98.77	99.67	100.00
MIN	88.54 (-3.34)	95.62 (-3.09)	97.75 (-1.92)	98.68 (-1.32)
MAX	61.15 (-30.73)	84.70 (-14.01)	89.00 (-10.67)	93.85 (-6.15)
MEDIAN	83.08 (-8.80)	95.19 (-3.52)	98.00 (-1.67)	99.34 (-0.66)
BORDA(C:All)	88.31 (-3.57)	97.34 (-1.37)	99.33 (-0.34)	99.67 (-0.33)
SUM	92.03 (0.15)	99.21 (0.50)	99.75 (0.08)	**100.00 (0.00)**
PROD	72.23 (-19.65)	99.35 (0.64)	**99.83 (0.16)**	**100.00 (0.00)**
PLUR	93.40 (1.52)	**99.50 (0.79)**	**99.83 (0.16)**	**100.00 (0.00)**
HC	93.32 (1.44)	98.78 (0.07)	99.42 (-0.25)	**100.00 (0.00)**
MOD-PLUR.	**93.63 (1.75)**	99.43 (0.72)	**99.83 (0.16)**	**100.00 (0.00)**

Min, max, median and Borda count methods do not surpass the accuracy of the best face expert in the respective experiment. A few words of comment are in order for the Borda count method: Should one use all the possible ranks from one up to the subject size or the top ranking ones? We have observed that combining the ranks of the top three achieves better results as compared to combining, for example, those of 195 subjects or any other subset. With this top-three Borda, the fusion loss in Table Table 7.11, becomes a fusion gain, of 1.97, 0.64, 0.16, 0.00 points for the four experiments E_1, E_2, E_3, and E_4, respectively. Apparently the most important information is in the top ranking face experts.

Plurality voting, despite its simplicity, performs very well. For example, it improves the best expert's classification rate by 1.44 per cent in E_1. The modified plurality (MOD-PLUR), presented in Chapter 6, performs better than its classical version (cf. MOD-PLUR with PLUR). The advantage of using confidence-aided fusion becomes more evident when we compare it with the performance of the highest confidence (HC) fusion rule. Essentially, HC rule is a classifier selection method similar to the frequently used MIN fusion rule. The only difference between HC and MIN rule is that HC uses

confidences to reach a decision whereas MIN rule uses normalized scores to select the final class label. For experiment E_1, MIN rule has 88.54 per cent identification rate. However, HC obtains 93.32 per cent identification rate which is better than the best face expert in the ensemble by 1.44 per cent. This observation is very important, and proves the superiority of the confidence-assisted fusion scheme. To recap, we recommend the use of the relative distance between the first and the second class candidates, if correctly normalized score measurements are not available, which is usually the case in single gallery experiments.

7.2.5. Construction of Face Classifier Ensembles

7.2.5.1. Classifier Selection by Sequential Floating Backward Search. In Section 7.2.4, we have fused the decisions of all of the 16 face experts. However, it is not immediately obvious whether inputting all experts in a fusion scheme is the best scheme to follow, simply because these individual experts may be correlated. One idea is to use a classifier selection method to design a better ensemble to be fused. Brute-force solution would be to construct all possible ensembles with number of selected classifiers ranging from 1 to 16. However, this is not practical given the number of combinations. Instead, the problem can be formulated as a feature selection problem. In analogy to the feature selection methods, we consider each classifier as a feature, and apply the sequential floating backward search (SFBS) [91] to find the near-optimal subset for each fusion technique. SFBS-based classifier selection algorithm can be stated as follows:

1. *Initialization step:* Start with the total ensemble set (Ω_{in}) of all of the face experts: $\Omega_{in} = \{e_1 \dots e_n\}$. Set the discarded face expert subset to an empty set: $\Omega_{out} = \emptyset$
2. *Exclusion step:* For each face expert $e_i \in \Omega_{in}$, remove this expert from Ω_{in} and obtain the candidate subset $\Omega_{cand} = \{\Omega_{in} - e_i\}$. Calculate the classification rate of the candidate ensembles. Select the candidate subset, which produces the best classification rate (Ω^*_{cand}). If the accuracy of the selected candidate is greater than or equal to the accuracy of the set Ω_{in}, then perform the following updates,

$\Omega_{in} = \Omega^*_{cand}$, and $\Omega_{out} = \{\Omega_{out} \bigcup e_i\}$. Otherwise, stop.

3. *Inclusion step:* (If the cardinality of the $\Omega_{out} > 2$) Form a candidate subset Ω_{can} by including a single face expert e_i from the previously discarded face experts subset Ω_{can} : $\{\Omega_{in} \bigcup e_i\}$. If the classification rate of the candidate set Ω_{can} is better than the accuracy of the set Ω_{in}, then include expert e_i to the subset $\Omega_{in} = \{\Omega_{in} \bigcup e_i\}$, and remove e_i from the $\Omega_{out} = \{\Omega_{out} - e_i\}$. Try all of the remaining experts in the subset Ω_{out} to include to Ω_{in} in this fashion.
4. Try the exclusion and inclusion steps successively until there is no performance improvement. Output the subset Ω_{in}.

We have applied the SFBS algorithm to SUM, PROD, PLUR, HC, and MOD-PLUR fusion schemes, and found near-optimal subsets for E_1. The results are shown in Table 7.12 and Table 7.13. The second column of Table 7.12 shows the selected face experts in the found subsets. It is clear from the classification accuracies of experiment E_1 that it is possible to get better ensembles in terms of identification performance. For example, MOD-PLUR fusion rule attains 95.22 per cent identification rate by combining only eight face experts which is significantly better than using all of the 16 face experts in the original MOD-PLUR method (93.63 per cent). It should be noted that these subsets were found by applying SFBS for experiment E_1, and the recognition accuracies of the other experiments were reported for these specific subsets. This explains the performance degradation of the PRO fusion rule in E_4. We have chosen to report the accuracies for experiments E_2, E_3, and E_4 in order to test the generalization ability of the SFBS-based classifier selection algorithm. It is more appropriate to apply SFBS algorithm to a separate validation set, and then to report the final classification rates on an independent test set. However, our main concern is not to design a classifier selection algorithm, but to give a proof of the concept, that it is possible to construct better ensembles without using all of the available face experts.

7.2.5.2. Correlation Analysis of Face Experts. The SFBS-based construction of the face ensembles has proven that some of the base classifiers are redundant and inclusion of them may lead to sub-optimal identification rates. To validate this finding, we want

Table 7.12. Selected classifier subsets for different fusion methods.

Fusion	Expert Subset
SUM	PC-XYZ, CURV-{SI,PD,K}, TEX-PIXEL, TEX-GABOR, DI-ICA, PC-NMF
PRO	PC-XYZ, CURV-SI, CURV-{PD,H,K}, TEX-GABOR, DI-DFT, DI-ICA, DI-NMF
PLUR	CURV-SI, CURV-PD, CURV-K, TEX-PIXEL, TEX-GABOR, DI-DCT, PC-ICA
HC	CURV-SI, CURV-PD, CURV-H, TEX-GABOR, PC-ICA
MOD-PLUR	PC-XYZ, CURV-{SI,PD}, TEX-PIXEL, TEX-GABOR, DI-DFT, DI-NMF, PC-ICA

Table 7.13. Identification accuracies of the selected classifier subsets for different fusion methods (See Table 7.12).

Fusion	E_1	E_2	E_3	E_4
SUM	94.23 (2.35)	99.14 (0.43)	99.75 (0.08)	**100.00 (0.00)**
PRO	88.39 (-3.49)	99.14 (0.43)	99.75 (0.08)	99.89 (-0.11)
PLUR	94.84 (2.96)	99.28 (0.57)	99.75 (0.08)	**100.00 (0.00)**
HC	94.69 (2.81)	98.99 (0.28)	99.42 (-0.25)	**100.00 (0.00)**
MOD-PLUR	**95.22 (3.34)**	**99.50 (0.79)**	**99.92 (0.25)**	**100.00 (0.00)**

to conduct a correlation analysis of the binary decision outputs of the face experts. Correlation is a particular type of binary similarity measure techniques [94], and given two classifier outputs with [0,1] values, it can be computed as:

$$\rho_{i,j} = \frac{N_{11}N_{00} - N_{10}N_{01}}{\sqrt{(N_{11}+N_{10})(N_{01}+N_{00})(N_{11}+N_{01})(N_{10}+N_{00})}} \tag{7.2}$$

where, for classifiers C_i and C_j, N values denote the probabilities for the respective pair of correct/incorrect outputs, and can be calculated as in Table 7.14.

Given the 16 experts in Table 7.10, we have computed 120 pairwise correlation values, and we have found significant correlations between certain pairs of face experts. In order to visualize the multidimensional relationships between face experts, we have

Table 7.14. 2×2 probability computation for two classifiers C_i and C_j where $N_{11}+N_{01}+N_{10}+N_{00} = 1$

	C_j correct (1)	C_j wrong (0)
C_i correct (1)	N_{11}	N_{10}
C_i wrong (0)	N_{01}	N_{00}

first obtained the dissimilarities between pairs of classifiers by using $d_{i,j} = 1 - \rho_{i,j}$. Here $d_{i,j}$'s can be considered as a distance measure between classifier pairs. Then we apply multidimensional scaling (MDS) algorithm to construct a two dimensional space $\Re^2$ where the coordinates denote the individual classifiers. We have found that the space of the two largest eigenvectors suffices to reasonably reproduce the space of face experts. Figures 7.10.a and 7.10.b show the reproduced face expert coordinates in $\Re^2$. The coordinates of the face experts (black dots) are the same in both figures. Figures 7.10.a also shows visually delineated face expert clusters (dashed ellipses). There are five salient clusters, and with few exceptions, each cluster matches one of the face representation methods. For instance, curvature-based, depth image-based, point cloud-based and texture-based face experts form their own clusters.

In Figure 7.10.a, gray circles denote the selected face classifiers for the SUM fusion rule. Selected face experts for the MOD-PLUR fusion scheme are shown as gray circles in Figure 7.10.b, similarly. When we examine the selected face experts for both the SUM and MOD-PLUR fusion methods, we see that these experts come from different clusters. This finding shows that if you have different base classifiers, then their combination can attain better accuracies, although their own accuracies are moderate.

7.2.5.3. Classifier Selection by Best-N Method. A second alternative method to construct the ensemble would be to combine the best N face experts. However, since this method does not exploit the diversity of decision takers, we conjecture that it may not perform as well. In order to validate this hypothesis, we have combined best N face experts where $N \in \{2, 3, \ldots 16\}$ with MOD-PLUR fusion rule. The fusion per-

formance of the best ensembles is plotted in Figure 7.11 (black dotted curve), while the two lines correspond to the accuracies of the total ensemble (the N=16 case and the SFBS ensemble, respectively. As expected, the indiscriminate ensemble of the best ones performs worse than the judiciously chosen SFBS subset case.

7.2.5.4. Classifier Selection by Correlation Analysis. The third and final ensemble construction method is to hand-pick them using the cluster map of Figure 7.10. The heuristic for the ensemble construction is to select one face expert from each cluster in order to enforce diversity of opinions. Although each cluster offers the choice of more than one expert, we follow the greedy approach of choosing the best performing face expert from each cluster. Table 7.15 shows the performance of the five fusion rules applied to the hand-picked ensemble, namely, CURV-PD, TEX-GABOR, DI-DCT, and PC-ICA for the experiment E_1. We exclude the VOXEL-DFT method because its solo classification performance is very low. The second row of Table 7.15 replicates the fusion performances from Table 7.11. Comparison of these results reveals the advantage of getting guidance from clustering of experts. The only drop in accuracy occurs for the PLUR method since voting within an ensemble of small cardinality is known not to perform well.

Table 7.15. Selection of classifier ensembles by clustering.

	SUM	PRO	PLUR	HC	MOD-PLUR
All 16 experts	92.03	72.23	93.40	93.32	93.63
Best of Clusters	92.26	85.82	91.35	94.23	94.01

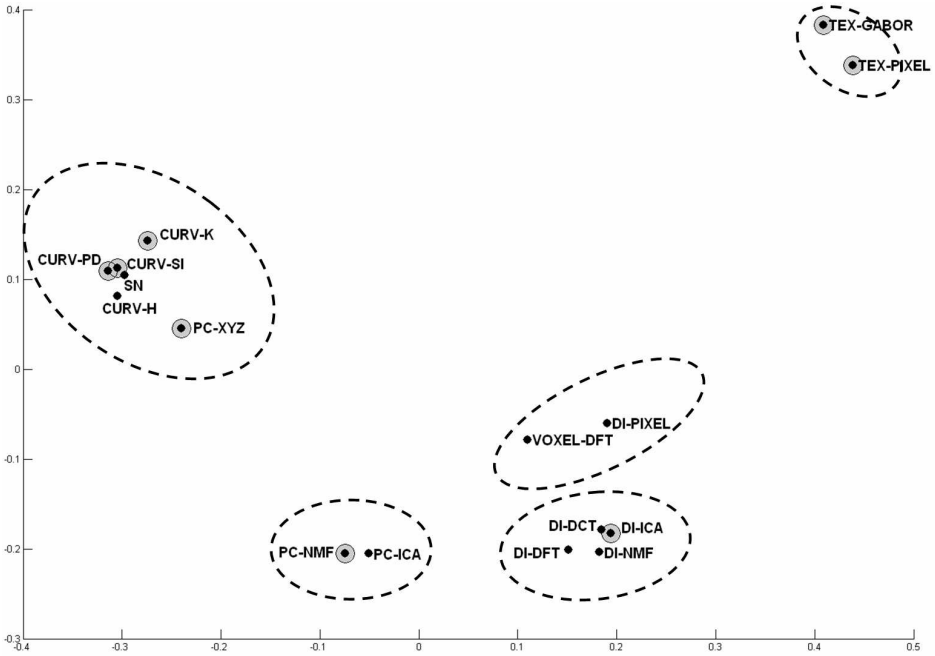
7.2.5.5. Fusion of single shape and single texture expert. So far, we have let the algorithms to treat the experts taking role in the consultation session. However, most papers in the 3D face recognition literature fuse only two face experts with the SUM rule: one for the shape modality, and one for the texture modality. Typically, for the shape modality, ICP-based point cloud algorithm is chosen. For completeness, we present the results of this restricted fusion scheme, where one choice from texture category and another one from shape category is imposed. In Table 7.16 we provide the

results of three ensembles, with TEX-GABOR being the only choice for the texture-based expert. In the shape category three different shape experts take role, namely PC-XYZ, DI-DCT, CURV-PD, each a best representative of its own group. From the performance figures in Table 7.16, one can see that the combination of TEX-GABOR and CURV-PD algorithms significantly outperforms the other two ensembles, and that this twosome has classification performance comparable to that of all 16 face experts fused.

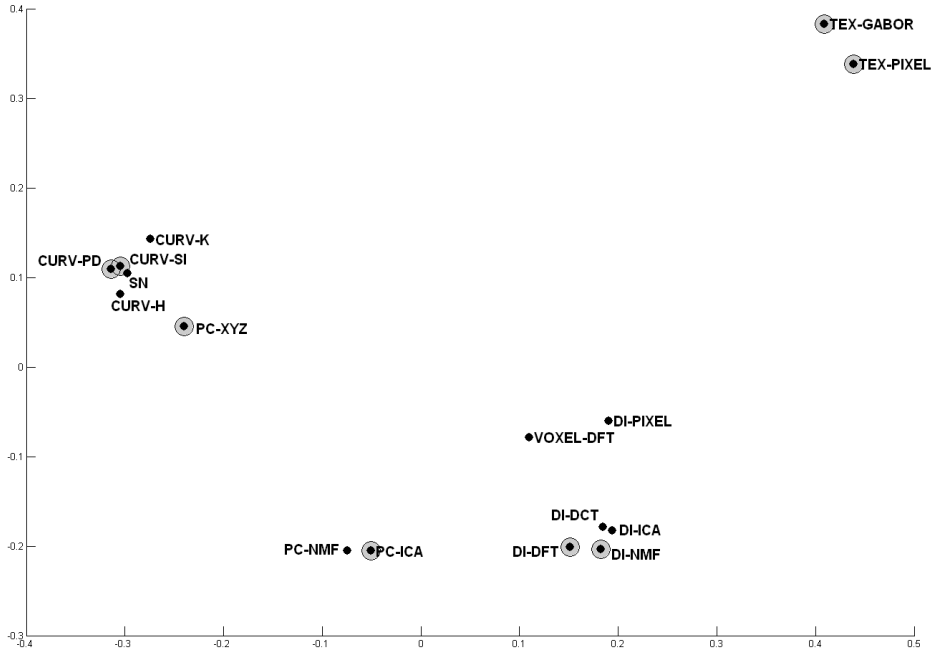
Table 7.16. Fusion of shape and texture experts using the SUM rule.

Texture Expert	Shape Expert	Rank-1 Accuracy in E_1
TEX-GABOR	PC-XYZ	89.98
TEX-GABOR	DI-DCT	84.67
TEX-GABOR	CURV-PD	93.63

7.2.5.6. Overall comparison of fusion schemes and classifier selection methods. The bar chart in Figure 7.12 summarizes the fusion results given in the previous sections. There are four fusion schemes (SUM, PLUR, HC, and MOD-PLUR) and four ensemble construction algorithms (Ens-All, Ens-SFBS, Ens-Cluster, and Ens-BestN). We have already seen that the optimal ensemble formation is SFBS, adopted from feature selection literature. The essential pre-requisite for a good ensemble formation is the complementariness. Since fusing best N individual experts does not enforce complementariness, it only attains a moderate performance (see Ens-BestN in Figure 7.12), while the cluster-guided method (Ens-Cluster) satisfies it heuristically, and performs better. In terms of fusion algorithms, the following conclusions can be drawn: 1) in good ensembles, sum rule does not perform as well as the others, 2) if one has several face experts, plurality voting can be a better alternative to the sum rule, 3) it is possible to improve plurality voting with the aid of confidence weights, 4) if there are few experts, then selecting the class having the highest confidence (not the smallest score or distance) can lead to better identification rate than plurality voting



(a)



(b)

Figure 7.10. Correlation analysis of the face experts. In both images, black dots denote the two-dimensional positions of the face experts calculated from the MDS analysis of pairwise correlations for the first experiment E_1. Gray circles denote the expert subset found from (a) the SUM rule, and (b) the MOD-PLUR rule. In (a), large dashed ellipses denote visually salient clusters.

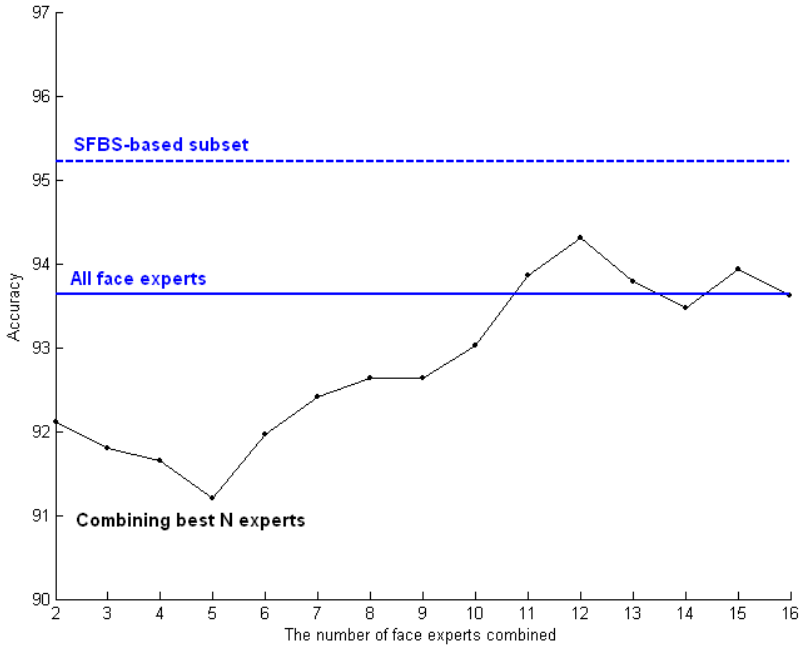


Figure 7.11. The identification performance of fusing best N classifiers where where $N \in \{2, 3, \ldots 16\}$ (x-axis) for the MOD-PLUR rule. Black-dotted curve denotes the Rank-1 accuracy of the best N fusion method. Horizontal dashed line, and the horizontal solid line denote the performance of the fused ensemble for i) SFBS-based face expert subset, and ii) using all 16 face experts in the ensemble, respectively.

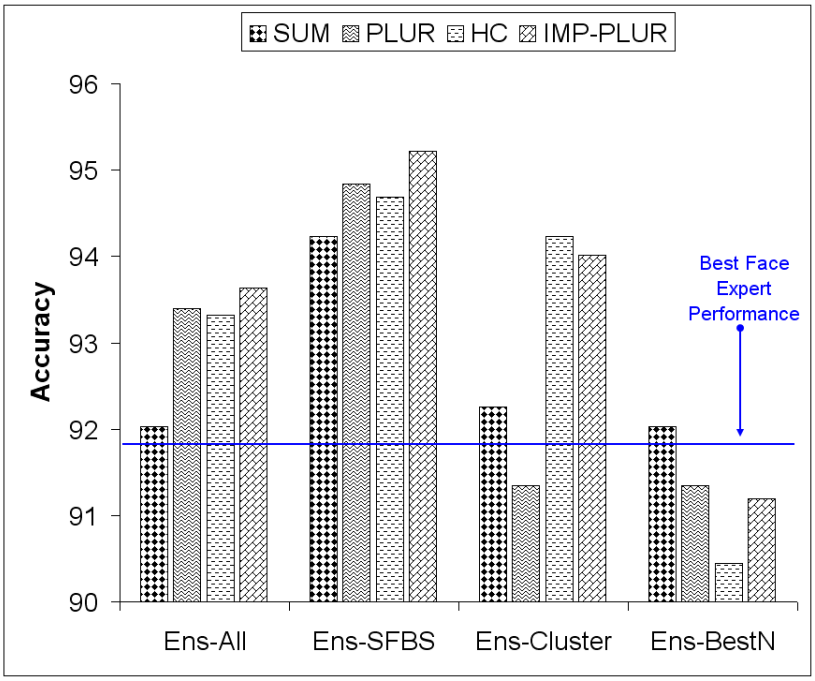


Figure 7.12. Overall comparison of the i) fusion techniques and ii) ensemble construction methods: Ens-All: All 16 experts in the ensemble, Ens-SFBS: Subset of face experts selected by the SFBS method, Ens-Cluster: Selection of best experts from each cluster (see Figure 7.10), Ens-BestN: Selection of the most accurate five face experts. Horizontal line denotes the best face expert's accuracy in the experiment E_1.

8. Conclusions

In this thesis, we have developed a complete 3D face recognition system. The developed system essentially matches 3D facial shapes in order to identify or verify a person. Depending on the availability of the texture modality, the 3D face recognition engine may also incorporate the 2D appearance information during its decision phase. Currently, the recognition system requires only the coordinates of the nose tip for coarse localization. However, if several landmark coordinates corners are available, it employs this information for more advanced alignment algorithms such as the Procrustes analysis. The overall system is divided into four main modules: 1) registration, 2) face representation, 3) facial feature extraction, and 4) decision-level fusion for the pattern identification phase. For all of these modules, we provide our novel algorithms and compare them with baseline approaches. Based on our findings in this thesis, we can draw conclusions under the following categories:

Registration Task: For the face recognition task, efficient normalization, alignment and registration of faces play an important role due to the similarities between faces. Many of the previously proposed systems register gallery and probe faces at the identification phase. However, this approach may be infeasible in terms of time complexity. In order to overcome this disadvantage, we propose to use a generic face model. Our approach register the gallery images with the face template at the enrollment phase, and requires only single registration with the probe image at the identification phase. Our findings show that using this registration method, we can obtain sufficiently good identification performance in a very fast manner.

Secondly, since faces are typical examples of deformable surfaces, local deformations may produce unacceptable correspondences. In order to examine this, we realize and compare two different dense point-to-point registration algorithms. Both of these algorithms employ a generic face model. The first one employs rigid estimation of correspondences using the Iterative Closest Point algorithm, and the second one allows the warping of individual faces to the generic face template. The warping-based approach

requires the localization of several fiducial landmarks. In our experimental simulations, we observe that rigid estimation of correspondences offers better classification rates than the warping-based approach. However, this finding should be validated under more extreme conditions such as when significant expression variations are present in the probe set. Identification experiments performed on the 3DRMA face database which contains largely neutral faces reveal that it is possible to improve the correct classification rates by 1.45 to 21.24 per cent for different face recognizers when ICP-based rigid registration is used. The best individual face expert which is based on surface normals obtains 97.72 per cent accuracy for the non-rigid case, and 99.17 per cent accuracy for the rigid case. The greatest difference (21.24 per cent improvement) is obtained by the LDA-based pattern recognizer.

Representation and Feature Extraction Task: There are various ways of representing faces. Two of the most commonly used techniques utilize point coordinates and depth images. In addition to these techniques, we propose the use of other surface descriptors such as surface normals and curvature-related descriptors. Depending on the representation technique used, discriminative features should be extracted. For this task, we employ various schemes such as statistical feature extraction methods (i.e., PCA and LDA) for depth images or principal curvature directions for the curvature-based representation scheme. Our findings show that i) it is possible to design better 3D face matchers using surface normals or curvature-based descriptors, and ii) it is possible to significantly reduce the feature dimensionality with the use of efficient feature extraction techniques without degrading the identification accuracy significantly. However, the efficiency of these extraction methods heavily depends on the availability of enough training samples. For instance, in single-gallery experiments on the 3DRMA database, surface normals attain the best classification rate with 90.14 per cent rate. If there are three gallery images per subject, then LDA features extracted from surface normals obtain 99.46 per cent identification rate. For the same configuration, the direct use of surface normals can only reach 98.33 per cent classification rate. Similarly, for the FRGC face database, surface normal features and principal curvature directions attain 89.07 and 91.88 per cent accuracies in the single-gallery experiments, respectively. In multiple-gallery experiments (i.e., four training images per person) NMF and ICA-

based extraction of point cloud features yield 100.00 and 99.89 per cent classification accuracies, respectively. These results confirm the benefit of using feature extraction techniques when sufficient training samples are available.

In addition to the use of feature extraction methods, selection of local features can also be viewed as another way of determining optimal feature sets. For this purpose, we offer to use local region based surface descriptors. Our experimental results show that it is possible to provide some invariance with the careful elimination of several facial regions. Based on our previous work for 2D face recognition, we propose to use a floating search-based subset selection mechanism to determine the useful regions for identification. Our preliminary results on the 3DRMA show that it is possible to accurately identify a person by focusing on these important regions. We found that these regions cover nearly the half of the whole facial surface. The locations of these regions can be found from a separate validation set, and thus, it is important to use big training and validation sets to converge to global optimum. We also observe that, without the selection of local regions, regular patch-based surface descriptors can both reduce the feature dimensionality significantly, and improve the identification rate considerably. For example, in the FRGC database, the best face recognizer which use principal directions obtain 80.43 per cent identification rate. By employing an averaging-based curvature descriptor, it is possible to improve the identification rate by 7.71 per cent. Another advantage of this approach is that the input feature dimensionality also reduces from 33.000 to 174.

Decision-level Fusion Task: Multi-modal nature of the 3D face signals makes it popular to use information fusion methods in 3D face recognition systems. Currently, most of the previously proposed approaches fuse shape and texture channels. In this thesis, we consider the fusion task from a more general perspective, and use various face experts (that use both shape and texture channels) as individual classifiers in the ensemble. The diversity of the face experts are provided by designing them by selecting different representations and feature extraction methods. We observe that it is beneficial to construct combined ensembles by selecting them according to their complementariness, i.e., fusing experts that are based on different representations may

be better than fusing different feature extraction-based algorithms that use the same representation scheme. This conclusion is observed in the FRGC fusion experiments where the best classifier ensemble is constructed using a floating search-based subset selection algorithm. We see that the best performing ensemble always contains individual face experts from different representations even if some of the base experts are weak. It is also observed that eliminating some experts may increase the identification rate. For instance, by fusing all of the 16 base classifiers, 92.03 per cent identification rate is obtained for the sum rule in the single-gallery experiment. However, by selecting only eight of them using floating search, it is possible to attain 94.23 per cent classification rate.

For the fusion task, we also propose to use two different decision-level combination algorithms. The first one employs the confidence idea where each classifier output its nearest class label together with its confidence. Thus it is possible to use these confidence values to determine the final class. We have implemented two variants of this approach. The first variant consults the confidence information when ties are present in the plurality voting algorithm, and the second variant simply selects the output of the classifier having the highest confidence. So, the second variant can be treated as a dynamic classifier selection algorithm. Experimentally, we notice that if the score normalization is problematic due to insufficient amount of training samples, it is better to rely on confidences since the confidences are estimated by the relative distance between the first and the second nearest neighbor. In our FRGC experiments, we observe that both confidence-aided plurality voting and highest confidence-based classifier selection improves the fusion performance of standard sum rule (92.03 per cent) by attaining 93.63 and 93.32 per cent identification rates, respectively. In addition to the confidence-assisted fusion scheme, our second proposal benefits from a two-stage architecture. At the first stage, first base classifier finds the nearest classes given a test image, and forwards this class information to the second classifier. The second classifier dynamically constructs an LDA space using the training samples of the forwarded classes. Therefore, it better distinguishes the classes in this new subspace, and works more accurately. In multiple-gallery experiments for the FRGC database, we see that two-stage serial fusion may improve the accuracies of the sum-based and confidence-

based combination schemes by approximately 3 per cent, obtaining 97.93 per cent classification rate.

In summary, we have shown that it is possible to improve the performance of a 3D face recognition system by utilizing the proposed algorithms in the registration, representation, feature extraction and fusion stages. However, it is worthwhile to perform a comparative analysis of two different registration techniques, namely: registration with the aid of generic face template, and the registration without the generic face template (i.e., matching the probe image to all of the gallery images at the identification phase). It would also be beneficial to estimate coarsely the expression of a face before proceeding to the matching step. Local representations may provide such information, and it is thus possible to concentrate on other parts of the facial surface for identity recognition. Another future work is related to the fusion phase. It is shown that the confidence idea is promising for both fusion or selection of base face experts. It would be interesting to explore possible alternatives for the estimation of confidences or the use of confidences. More general and long-term future interests should also include the followings 1) search for alternative 3D descriptors, 2) robust methods for significant pose and expression variations (i.e., uncontrolled acquisition), 3) design of matchers for occluded facial surfaces, 4) issues related to the operating conditions of a typical 3D face recognizer (i.e, what should be the quality of the data?) and 5) conformance of the proposed algorithms to the needs of real-time systems.

APPENDIX A: Iterative Closest Point Algorithm

The ICP algorithm is a quaternion-based least square solution for finding rotation and translation transformations between two point sets [72]. The pseudocode of the ICP algorithm is illustrated in Figure A.1. Aim is to rotate and translate point set P such that it aligns best to the point set X. The ICP algorithm is an iterative algorithm and it terminates when the alignment error is below a certain threshold τ or a maximum number of iteration is reached. At step 4, for each point in the set X, its corresponding nearest point is found in the set P (operation $\mathcal{C}$). Once the correspondence established, registration parameters can be found at step 5 (operation $\mathcal{Q}$). The operation $\mathcal{Q}$ is explained in the next section (Section A.1).

A.1. Calculation of Registration Parameters

A 3×3 rotation matrix can be expressed via quaternion notation as:

$$\mathbf{R}(\overrightarrow{q_R}) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 + q_2^2 - q_1^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 + q_3^2 - q_1^2 - q_1^2 \end{bmatrix} \tag{A.1}$$

where the unit *rotation* quaternion is a four vector $\overrightarrow{q_R} = [q_0 q_1 q_2 q_3]^t$. $q_0 \geq 0$ and $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$. Let $\overrightarrow{q_T} = [q_4 q_5 q_6]^t$ be a translation vector. The complete registration vector $\overrightarrow{q}$ is denoted by $\overrightarrow{q} = [\overrightarrow{q_R}\overrightarrow{q_T}]^t$. Let $P = \{\overrightarrow{p_i}\}$ be a measured data point set to be aligned with a model point set $X = \{\overrightarrow{x_i}\}$, where $N_p = N_x$ and the points having the same index correspond to each other. The mean square objective function to be minimized is:

$$f(\overrightarrow{q}) = \frac{1}{N_p} \sum_{i=1}^{N_p} \| \overrightarrow{x_i} - \mathbf{R}(\overrightarrow{q_R})\overrightarrow{p_i} - \overrightarrow{q_T} \|^2 \tag{A.2}$$

ALGORITHM: Iterative Closest Point Algorithm (ICP)

FUNCTION: Finds the rigid transformation between two point sets.

INPUT:

Scene Point Set: $P = \{\overrightarrow{p}_i\}$ with N_p points

Model Point Set: $X = \{\overrightarrow{x}_i\}$ with N_x points

Convergence Threshold: τ

OUTPUT:

Transformation parameters: $\overrightarrow{q}$

Registration error: d

1 ICP(P, X, τ)

2 Initialize $P_0 = P$, $q_0 = [1, 0, 0, 0, 0, 0, 0]^T$, and $k = 0$.

3 **Repeat until** $d_k - d_{k+1} < \tau$

4 Compute the closest points: $Y_k = \mathcal{C}(P_k, X)$

5 Compute the registration: $(\overrightarrow{q}_k, d_k) = \mathcal{Q}(P_0, Y_k)$

6 Apply the registration: $P_{k+1} = \overrightarrow{q}_k(P_0)$

7 RETURN: $(\overrightarrow{q}, d)$

Figure A.1. Pseudocode of the Iterative Closest Point algorithm.

In the minimization of $f(\overrightarrow{q})$, first the rotation matrix is computed, then the translation translation matrix is estimated. In order to show the computation of rotation parameters, the following notation should be presented. The cross-covariance matrix Σ_{px} of the sets P and X is given by

$$\Sigma_{px} = \frac{1}{N_p}\sum_{i=1}^{N_p}[(\overrightarrow{p_i} - \overrightarrow{\mu_p})(\overrightarrow{x_i} - \overrightarrow{\mu_x})^t] = \frac{1}{N_p}\sum_{i=1}^{N_p}[\overrightarrow{p_i}\,\overrightarrow{x_i}^t] - \overrightarrow{\mu_p}\overrightarrow{\mu_x^t} \tag{A.3}$$

where the center of mass of P and the center of mass of X are given by

$$\overrightarrow{\mu_p} = \frac{1}{N_p}\sum_{i=1}^{N_p}\overrightarrow{p_i} \quad \text{and} \quad \overrightarrow{\mu_x} = \frac{1}{N_X}\sum_{i=1}^{N_X}\overrightarrow{x_i} \tag{A.4}$$

The optimal rotation ($\overrightarrow{q_R} = [q_0 q_1 q_2 q_3]^t$) corresponds to the maximum eigenvalue of the matrix $Q(\Sigma_{px})$. The symmetric 4×4 matrix $Q(\Sigma_{px})$ is formed by

$$Q(\Sigma_{px}) = \begin{pmatrix} tr(\Sigma_{px}) & \Delta^T \\ \Delta & \Sigma_{px} + \Sigma_{px}^T - tr(\Sigma_{px})\mathbf{I_3} \end{pmatrix} \tag{A.5}$$

where $\mathbf{I_3}$ is a 3×3 identity matrix and Δ is the column matrix: $\Delta = [A_{23} A_{31} A_{12}]^t$. Here, the A_{ij} is calculated by $A_{ij} = (\Sigma_{px} - \Sigma_{px}^T)_{ij}$. Lastly, the optimal translation vector is given by

$$\overrightarrow{q_T} = \overrightarrow{\mu_x} - \mathbf{R}(\overrightarrow{q_R})\overrightarrow{\mu_p} \tag{A.6}$$

REFERENCES

1. Akarun, L., B. Gökberk, and A. A. Salah, "3D Face Recognition for Biometric Applications", *Proc. of the 13th European Signal Processing Conference (EUSIPCO)*, 2005.

2. Gökberk, B., M. O. İrfanoğlu, and L. Akarun, "3D shape-based face representation and facial feature exration for face recognition", *Image and Vision Computing*, Vol. 24, No. 8, pp. 857–869, 2006.

3. İrfanoğlu, M. O., B. Gökberk, and L. Akarun, "3D Shape based Face Recognition using Automatically Registered Facial Surfaces", *International Conference on Pattern Recognition*, pp. 183–186, 2004.

4. Gökberk, B., H. Dutağacı, L. Akarun, and B. Sankur, "Representation Plurality and Decision Level Fusion for 3D Face Recognition", *IEEE Transactions on Systems, Man and Cybernetics: Part-B (under revision).*

5. Gökberk, B., M. O. İrfanoğlu, L. Akarun, and E. Alpaydın, "Learning the Best Subset of Local Features for Face Recognition", *Pattern Recognition*, Vol. 40, No. 5, pp. 1520–1532, 2007.

6. Gökberk, B., M. O. İrfanoğlu, L. Akarun, and E. Alpaydın, "Optimal Gabor Kernel Selection for Face Recognition", *Proceedings of the International Conference on Image Processing*, pp. 677–680, 2003.

7. Gökberk, B., L. Akarun, and E. Alpaydın, "Feature Selection for Pose Invariant Face Recognition", *Proceedings of the 16th International Conference on Pattern Recognition*, pp. 306–309, 2002.

8. Gökberk, B. and L. Akarun, "Selection and Extraction of Patch Descriptors for 3D Face Recognition", *Proc. of the 20th International Symposium Computer and*

Information Sciences (ISCIS). Lecture Notes in Computer Science, Vol. 3733, pp. 718–727, 2005.

9. Gökberk, B., M. O. İrfanoğlu, L. Akarun, and E. Alpaydın, "Selection of Location, Frequency, and Orientation Parameters of 2D Gabor Wavelets for Face Recognition", *Proc. of the Summer School on Biometrics: Advanced Studies in Biometrics. Lecture Notes in Computer Science*, Vol. 3161, pp. 138–146, 2005.

10. Gökberk, B., A. A. Salah, and L. Akarun, "Rank-based Decision Fusion for 3D Shape-based Face Recognition", Kanade, T., A. Jain, and N. K. Ratha (editors), *Proceedings of Audio- and Video-based Biometric Person Authentication, Lecture Notes in Computer Science*, Vol. 3456, pp. 1019–1029, 2005.

11. Gökberk, B. and L. Akarun, "Comparative Analysis of Decision-level Fusion Algorithms for 3D Face Recognition", *Proc. of the 18th International Conference on Pattern Recognition*, pp. 1018– 1021, 2006.

12. Lu, X., A. Jain, and D. Colbry, "Matching 2.5D Face Scans to 3D Models", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 28, No. 1, pp. 31–43, 2006.

13. Pan, G. and Z. Wu, "3D Face Recognition From Range Data", *International Journal of Image and Graphics*, Vol. 5, No. 3, pp. 573–583, 2005.

14. Papatheodorou, T. and D. Reuckert, "Evaluation of automatic 4D face recognition using surface and texture registration", *Sixth International Conference on Automated Face and Gesture Recognition*, pp. 321–326, 2004.

15. Bowyer, K. W., K. Chang, and P. Flynn, "A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition", *Computer Vision and Image Understanding*, Vol. 101, No. 1, pp. 1–15, 2006.

16. Medioni, G. and R. Waupotitsch, "Face recognition and modeling in 3D", *Proc.*

of the IEEE Int. Workshop on Analysis and Modeling of Faces and Gestures, pp. 232–233, 2003.

17. Amor, B. B., M. Ardabilian, and L. Chen, "New Experiments on ICP-Based 3D Face Recognition and Authentication", *ICPR 2006*, 2006.

18. Russ, T., C. Boehnen, and T. Peters, "3D Face Recognition Using 3D Alignment for PCA", *Proc. of. the IEEE Computer Vision and Pattern Recognition (CVPR06)*, 2006.

19. Koudelka, M., M. Koch, and T. Russ, "A prescreener for 3D face recognition using radial symmetry and the Hausdorff fraction", *IEEE Workshop on Face Recognition Grand Challenge Experiments*, 2005.

20. Achermann, B. and H. Bunke, "Classifying range images of human faces with Hausdorff distance", *15-th International Conference on Pattern Recognition*, p. 809813, 2000.

21. Mian, A., M. Bennamoun, and R. Owens, "2D and 3D Multimodal Hybrid Face Recognition", *Proc. of. ECCV, LNCS 3953*, p. 344355, 2006.

22. Wang, Y., G. Pan, Z. Wu, and Y. Wang, "Exploring Facial Expression Effects in 3D Face Recognition Using Partial ICP", *Proc. of . ACCV, LNCS 3851*, p. 581590, 2006.

23. Passalis, G., I. Kakadiaris, T. Theoharis, G. Toderici, and N. Murtuza, "Evaluation of 3D face recognition in the presence of facial expressions: an annotated deformable model approach", *IEEE Workshop on Face Recognition Grand Challenge Experiments*, 2005.

24. Lu, X. and A. K. Jain, "Deformation Modeling for Robust 3D Face Matching", *CVPR06*, 2006.

25. Papatheodorou, T. and D. Rueckert, "Evaluation of 3D Face Recognition Using

Registration and PCA", *AVBPA, LNCS Vol.3546*, pp. 997–1009, 2005.

26. Hong, T., H. Kim, H. Moon, Y. Kim, J. Lee, and S. Moon, "Face Representation Method Using Pixel-to-Vertex Map (PVM) for 3D Model Based Face Recognition", *ECCV, LNCS 3979*, pp. 21–28, 2006.

27. Bellon, O. R., L. Silva, and C. C. Queirolo, "3D Face Matching Using the Surface Interpenetration Measure", *ICIAP 2005, LNCS 3617*, p. 10511058, 2005.

28. Bronstein, A. M., M. Bronstein, and R. Kimmel, "Expression-invariant 3D face recognition", *International Conference on Audio- and Video-Based Person Authentication (AVBPA)*, Vol. 2688, pp. 62–70, 2003.

29. Bronstein, A. M., M. M. Bronstein, A. Spira, , and R. Kimmel, "Face Recognition from Facial Surface Metric", Pajdla, T. and J. Matas (editors), *Proceedings of the 8th European Conference on Computer Vision, Lecture Notes in Computer Science*, Vol. 3022, pp. 225–237, 2004.

30. Bronstein, A., M. Bronstein, and R. Kimmel, "Three-dimensional face recognition", *International Journal of Computer Vision*, Vol. 64, No. 1, pp. 5–30, 2005.

31. Chang, K. I., K. W. Bowyer, and P. J. Flynn, "An Evaluation of Multi-modal 2D+3D Face Biometrics", *IEEE Trans. on PAMI*, Vol. 27, No. 4, pp. 619–624, 2005.

32. Srivastava, A., X. Liu, and C. Hesher, "Face Recognition Using Optimal Linear Components of Range Images", *Image and Vision Computing*, Vol. 24, No. 3, pp. 291–299, 2006.

33. Hesher, C., A. Srivastava, and G. Erlebacher, "A novel technique for face recognition using range imaging", *Proc. of the Seventh Int. Symposium on Signal Processing and Its Applications*, pp. 201–204, Springer Verlag, 2003.

34. Zhong, C., T. Tan, C. Xu, and J. Li, "Automatic 3D Face Recognition Using

Discriminant Common Vectors", *International Conference on Biometrics, LNCS 3832*, p. 8591, 2006.

35. Lee, Y., K. Park, J. Shim, and T. Yi, "3D face recognition using statistical multiple features for the local depth information", *Proc. of the 16th Int. Conf. on Vision Interface*, 2003.

36. Pan, G., S. Han, Z. Wu, and Y. Wang, "3D face recognition using mapped depth images", *IEEE Workshop on Face Recognition Grand Challenge Experiments*, 2005.

37. Russ, T., M. Koch, and C. Little, "A 2D range Hausdorff approach for 3D face recognition", *IEEE Workshop on Face Recognition Grand Challenge Experiments*, 2005.

38. Cook, J., V. Chandran, S. Sridharan, and C. Fookes, "Gabor Filter Bank Representation for 3D Face Recognition", *Proceedings of the Digital Imaging Computing: Techniques and Applications (DICTA)*, 2005.

39. Beumier, C. and M. Acheroy, "Automatic 3D Face Authentication", *Image and Vision Computing*, Vol. 18, No. 4, pp. 315–321, 2000.

40. Beumier, C. and M. Acheroy, "Face verification from 3D and grey level cues", *Pattern Recognition Letters*, Vol. 22, pp. 1321–1329, 2001.

41. Zhang, L., A. Razdan, G. Farin, J. Femiani, M. Bae, and C. Lockwood, "3D face authentication and recognition based on bilateral symmetry analysis", *Visual Comput (22)*, p. 4355, 2006.

42. Feng, S., H. Krim, I. Gu, and M. Viberg, "3D Face Recognition using Affine Integral Invariants", *Proc. of. ICASSP*, pp. 189–192, 2006.

43. Gordon, G., "Face Recognition Based on Depth and Curvature Features", *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pp. 108–110, 1992.

44. Tsutsumi, S., S. Kikuchi, and M. Nakajima, "Face identification using a 3D gray-scale image-a method for lessening restrictions on facial directions", *Proc. of the* 3^{rd} *IEEE Int. Conf. on Automatic Face and Gesture Recognition*, pp. 306–311, 1998.

45. Moreno, A. B., A. Sanchez, J. F. Velez, and F. J. Diaz, "Face recognition using 3D surface-extracted descriptors", *Proc. of the Irish Machine Vision and Image Processing Conf.*, 2003.

46. Tanaka, H., M. Ikeda, and H. Chiaki, "Curvature-based face surface recognition using spherical correlation principal directions for curved object recognition", *Third International Conference on Automated Face and Gesture Recognition*, pp. 372–377, 1998.

47. Lee, Y.-H., C.-W. Han, , and T.-S. Kim, "Soft Computing Based Range Facial Recognition Using Eigenface", *ICCS 2006 LNCS 3994*, p. 862 869, 2006.

48. Abate, A., M. Nappi, S. Ricciardi, and G. Sabatino, "Fast 3D Face Recognition Based On Normal Map", *IEEE International Conference on Image Processing*, Vol. 2, pp. 946–949, 2005.

49. Abate, A. F., M. Nappi, D. Riccio, and G. Sabatino, "3D Face Recognition using Normal Sphere and General Fourier Descriptor", *ICPR 2006*, 2006.

50. Riccio, D. and J.-L. Dugelay, "Asymmetric 3D/2D Processing: A Novel Approach for Face Recognition", *ICIAP 2005 LNCS 3617*, pp. 986–993, 2005.

51. Lee, Y., H. Song, U. Yang, H. Shin, and K. Sohn, "Local Feature Based 3D Face Recognition", *AVBPA 2005, LNCS 3546*, p. 909918, 2005.

52. Chua, C.-S., F. Han, and Y.-K. Ho, "3D Human Face Recognition Using Point Signature", *Proc. of Int. Conf. on Automatic Face and Gesture Recognition*, pp. 233–237, 2000.

53. Wang, Y., C. Chua, and Y. Ho, "Facial feature detection and face recognition from 2D and 3D images", *Pattern Recognition Letters*, Vol. 23, No. 10, pp. 1191–1202, 2002.

54. Wu, Z., Y. Wang, and G. Pan, "3D face recognition using local shape map", *Processings of the Int. Conf. on Image Processing*, Vol. 3, pp. 2003– 2006, 2004.

55. Xu, C., Y. Wang, T. Tan, and L. Quan, "Automatic 3D face recognition combining global geometric features with local shape variation information", *Proc. of the Sixth Int. Conf. on Automated Face and Gesture Recognition*, pp. 308–313, 2004.

56. Xu, C., T. Tan, S. Li, Y. Wang, and C. Zhong, "Learning Effective Intrinsic Features to Boost 3D-Based Face Recognition", *Proc. of ECCV, LNCS 3952*, p. 416427, 2006.

57. Wang, Y., G. Pan, Z. Wu, and S. Han, "Sphere-Spin-Image: A Viewpoint-Invariant Surface Representation for 3D Face Recognition", *ICCS 2004, LNCS 3037*, p. 427434, 2004.

58. Tsalakanidou, F., D. Tzocaras, and M. Strintzis, "Use of depth and colour eigenfaces for face recognition", *Pattern Recognition Letters*, Vol. 24, pp. 1427–1435, 2003.

59. Tsalakanidou, F., S. Malassiotis, and M. Strintzis, "Integration of 2D and 3D images for enhanced face authentication", *Sixth International Conference on Automated Face and Gesture Recognition*, pp. 266–271, 2004.

60. Malassiotis, S. and M. G. Strintzis, "Robust face recognition using 2D and 3D data: Pose and illumination compensation", *Pattern Recognition*, Vol. 38, No. 12, pp. 2537–2548, 2005.

61. Tsalakanidou, F., S. Malassiotis, and M. Strintzis, "Face localization and authentication using color and depth images", *IEEE Transactions on Image Processing*,

Vol. 14, No. 2, pp. 152–168, 2005.

62. BenAbdelkader, C. and P. A. Griffin, "Comparing and combining depth and texture cues for face recognition", *Image and Vision Computing*, Vol. 23, No. 3, pp. 339–352, 2005.

63. Wang, Y. and C.-S. Chua, "Robust face recognition from 2D and 3D images using structural Hausdorff distance", *Image and Vision Computing*, Vol. 24, No. 2, pp. 176–185, 2006.

64. Wang, Y. and C.-S. Chua, "Face recognition from 2D and 3D images using 3D Gabor filters", *Image and Vision Computing*, Vol. 23, No. 11, pp. 1018–1028, 2005.

65. Maurer, T., D. Guigonis, I. Maslov, B. Pesenti, A. Tsaregorodtsev, D. West, and G. Medioni, "Performance of Geometrix ActiveID 3D face recognition engine on the FRGC data", *IEEE Workshop on Face Recognition Grand Challenge Experiments*, 2005.

66. Chang, K., K. Bowyer, and P. Flynn, "Adaptive rigid multi-region selection for handling expression variation in 3D face recognition", *IEEE Workshop on Face Recognition Grand Challenge Experiments*, 2005.

67. Husken, M., M. Brauckmann, S. Gehlen, and C. von der Malsburg, "Strategies and benefits of fusion of 2D and 3D face recognition", *IEEE Workshop on Face Recognition Grand Challenge Experiments*, 2005.

68. Li, S. Z., C. Zhao, M. Ao, and Z. Lei, "Learning to Fuse 3D+2D Based Face Recognition at Both Feature and Decision Levels", *AMFG 2005, LNCS 3723*, pp. 44–54, 2005.

69. Kakadiaris, I., G. Passalis, T. Theoharis, G. Toderici, I. Konstantinidis, and N. Murtuza, "Multimodal Face Recognition: Combination of Geometry with Phys-

iological Information", *CVPR 2005 (Volume 2)*, pp. 1022–1029, 2005.

70. Chang, K. I., K. W. Bowyer, and P. J. Flynn, "Multiple Nose Region Matching for 3D Face Recognition under Varying Facial Expression", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 28, No. 10, pp. 1695–1700, 2006.

71. Gower, J. C., "Generalised Procrustes Analysis", *Psychometrika*, Vol. 40, pp. 33–50, 1975.

72. Besl, P. and N. McKay, "A Method for Registration of 3D Shapes", *IEEE Trans. on PAMI*, Vol. 14, pp. 239–256, 1992.

73. Dutağacı, H., B. Sankur, and Y. Yemez, "3D Face recognition by projection-based features", *Proc. SPIE Conf. on Electronic Imaging: Security, Steganography, and Watermarking of Multimedia Contents*, 2006.

74. Bartlett, M. S., J. R. Movellan, and T. J. Sejnowski, "Face Recognition by Independent Component Analysis", *IEEE Transactions on Neural Networks*, Vol. 13, No. 6, pp. 1450–1464, 2002.

75. Dutağacı, H., B. Sankur, and Y. Yemez, "A comparison of data representation types, features types and fusion techniques for 3D face biometry", *Proc. of the 14th European Signal Processing Conference (EUSIPCO 2006)*, 2006.

76. Hyvarinen, A. and E. Oja, "Independent Component Analysis: Algorithms and Applications", *Neural Networks*, Vol. 13, No. 4-5, pp. 411–430, 2000.

77. Lee, D. and H. Seung, "Algorithms for Nonnegative Matrix Factorization", *Advances in Neural Information Processing Systems*, Vol. 13, 2001.

78. Colombo, A., C. Cusano, and R. Schettini, "3D face detection using curvature analysis", *Pattern Recognition*, Vol. 39, pp. 444–455, 2006.

79. Turk, M. and A. Pentland, "Eigenfaces for recognition", *Journal of Cognitive Neu-*

roscience, Vol. 3, No. 1, pp. 71–86, 1991.

80. Kalocsai, P., C. Malsburg, and J. Horn, "Face recognition by statistical analysis of feature detectors", *Image and Vision Computing*, Vol. 18, No. 4, pp. 273–278, 2000.

81. Wiskott, L., J. M. Fellous, N. Kruger, and C. Malsburg, "Face recognition by elastic bunch graph matching", *IEEE Tran. on PAMI*, Vol. 19, No. 7, pp. 775–779, 1997.

82. Daugman, J. G., "Complete discrete 2D Gabor transforms by neural networks for image analysis and compression", *IEEE Tran. on Acoustics, Speech, and Signal Processing*, Vol. 36, No. 7, pp. 1169–1179, 1988.

83. Ayinde, O. and Y. H. Yang, "Face Recognition Approach based on Rank Correlation of Gabor-filtered Images", *Pattern Recognition*, Vol. 35, No. 6, pp. 1275–1289, 2002.

84. Wiskott, L., "Phantom Faces for Face Analysis", *Pattern Recognition*, Vol. 30, No. 6, pp. 837–846, 1997.

85. Kruger, N., M. Potzsch, and C. Malsburg, "Determination of face position and pose with a learned representation based on labelled graphs", *Image and Vision Computing*, Vol. 15, pp. 665–673, 1997.

86. Tefas, A., C. Kotropoulos, and I. Pitas, "Using Support Vector Machines to Enhance the Performance of Elastic Graph Matching for Frontal Face Authentication", *IEEE Tran. on PAMI*, Vol. 23, No. 7, pp. 735–746, 2001.

87. Liu, D. H., K. M. Lam, and L. S. Shen, "Optimal Sampling of Gabor Features for Face Recognition", *Pattern Recognition Letters*, Vol. 25, No. 2, pp. 267–276, 2004.

88. Yang, P., S. Shan, W. Gao, S. Z. Li, and D. Zang, "Face Recognition using Adaboosted Gabor Features", *Proceedings of the 16th International Conference on Face and Gesture Recognition*, 2004.

89. Wang, X. and H. Oi, "Face Recognition using Optimal Non-orthogonal Wavelet Basis Evaluated by Information Complexity", *Proceedings of the 16th International Conference on Pattern Recognition*, pp. 164–167, 2002.

90. Salah, A. A., E. Alpaydın, and L. Akarun, "A Selective Attention Based Method for Visual Pattern Recognition with Application to Handwritten Digit Recognition and Face Recognition", *IEEE Tran. on PAMI*, Vol. 24, No. 3, pp. 420–425, 2002.

91. Jain, A., R. P. W. Duin, and J. Mao, "Statistical pattern recognition: a review", *IEEE Tran. on PAMI*, Vol. 22, No. 1, pp. 4–37, 2000.

92. Phillips, P. J., H. Wechsler, J. Huang, and P. Rauss, "The FERET database and evaluation procedure for face recognition algorithms", *Image and Vision Computing*, Vol. 16, No. 5, pp. 295–306, 1998.

93. Demir, C. and E. Alpaydın, "Cost-conscious classifier ensembles", *Pattern Recognition Letters*, Vol. 26(14), pp. 2206–2214, 2005.

94. Kuncheva, L. I., *Combining Pattern Classifiers: Methods and Algorithms*, Wiley, 2004.

95. Kittler, J., M. Hatef, R. Duin, and J. Matas, "On combining classifiers", *IEEE Trans. on PAMI*, Vol. 20, No. 3, pp. 226–239, 1998.

96. Hampel, F., P. Rousseeuw, E. Ronchetti, and W. Stahel, *Robust Statistics: The Approach Based on Influence Functions*, Wiley, 1986.

97. Xu, C., Y. Wang, T. Tan, and L. Quan., "A New Attempt to Face Recognition Using Eigenfaces.", *Proc. of the Sixth Asian Conf. on Computer Vision*, Vol. 2, pp. 884–889, 2004.

98. Chang, K., K. W. Bowyer, and P. J. Flynn, "Face recognition using 2D and 3D facial data", *ACM Workshop on Multimodal User Authentication*, pp. 25–32, 2003.