2017-03-01

3D Ear Biometrics and Surveillance Video Based Biometrics.

Sayan Maity
University of Miami, sayanmaity.10@gmail.com

UNIVERSITY OF MIAMI

3D EAR BIOMETRICS AND SURVEILLANCE VIDEO BASED BIOMETRICS

By

Sayan Maity

A DISSERTATION

Submitted to the Faculty
of the University of Miami
in partial fulfillment of the requirements for
the degree of Doctor of Philosophy

Coral Gables, Florida

May 2017

UNIVERSITY OF MIAMI

A dissertation submitted in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

3D EAR BIOMETRICS AND SURVEILLANCE VIDEO BASED BIOMETRICS

Sayan Maity

Approved:

Shihab S. Asfour, Ph.D.
Chair
W. Alton Jones Professor of
Industrial Engineering

Mohamed Abdel-Mottaleb, Ph.D.
Co-Chair
Professor of Electrical and
Computer Engineering

Francesco Travascio, Ph.D.
Assistant Professor of Industrial
Engineering

Mohamed Fahmy, Ph.D.
Lecturer of Industrial Engineering

Moataz Eltoukhy, Ph.D.
Assistant Professor of Kinesiology
and Sport Sciences

Guillermo Prado, Ph.D.
Dean of the Graduate School

MAITY, SAYAN (Ph.D., Industrial Engineering)

3D Ear Biometrics and Surveillance Video Based Biometrics (May 2017)

Abstract of a dissertation at the University of Miami.

Dissertation supervised by Professor Shihab S. Asfour.
No. of pages in text. (137)

In the current era of Digital Information Technology Biometrics authentication is massively used to protect user's privacy by confirming the legitimacy of their identity. Biometric identifiers are the distinctive, measurable characteristics used to label and describe individuals. Technology to verify a persons identity based on his/her biometrics utilizes the information "someone who they are" instead of "something they know" (passwords) or "something they possess"(ID card). Biometric identifiers are usually categorized as physiological and behavioral characteristics. Physiological characteristics are related to the shape of the body. Examples include, but are not limited to face recognition, ear shape, fingerprint, DNA, palm print, hand geometry, iris recognition, and retina. Behavioral characteristics refer to the pattern of behavior of a person, including but not limited to typing rhythm, gait, and voice. Application of Biometrics authentication can be found in various domains, starting from forensics research, border security maintenance to securely access Bank ATMs and the web or mobile applications.

Current growth trends in different biometrics applications present challenges to researchers. To address these challenges, we need new data storage and retrieval techniques to make the recognition process time efficient. We proposed a system for time efficient 3D ear biometrics from a large biometrics database. The proposed system has

two components that are primarily responsible for: 1) automatic 3D ear segmentation and 2) hierarchical categorization of the 3D ear database using shape information and surface depth information, respectively. We use an active contour algorithm along with a tree structured graph to segment the ear region from the 3D profile images. The segmented 3D ear database is then categorized based on geometrical feature values, computed from the ear shape, into oval, round, rectangular and triangular categories. For the categorization based on the depth information, the feature space is partitioned using tree-based indexing techniques. We used indexing techniques with balanced split (KD tree) and unbalanced split (Pyramid tree) data structures to categorize the database separately, then compared their retrieval efficiency. Experiments are conducted to compare the average computation time per query when performing recognition through hierarchical categorization with the average computation time when recognition is based on sequential search. Experimental results conducted on the University of Notre Dame (UND) collection J2 dataset demonstrate that the proposed approach outperforms state-of-the-art 3D ear biometric systems in both accuracy and efficiency, explicitly the hierarchical clustering of the biometrics dataset result in 5 times faster search/ query compared with the state-of-the-art technique that uses sequential search.

Biometrics identification using multiple modalities has attracted the attention of many researchers as it produces more robust and trustworthy results than single modality biometrics. We proposed a novel multimodal recognition system that trains a Deep Learning Network to automatically learn features after extracting multiple biometric modalities from a single data source, i.e., facial video clips. Utilizing different modalities, i.e., left ear, left profile face, frontal face, right profile face, and right ear, present in the facial video clips, we train supervised denosing autoencoders to automatically extract robust and non-redundant features. The automatically learned features are then used to train modality specific sparse classifiers to perform the multimodal recognition. The proposed system has three components that are responsible for: 1) Automatically detecting images of different modalities present in the facial video clips; 2) Training supervised denoising sparse autoencoders to capture the modaliti specific discriminative representation while maintaining robustness to the variations; and 3) Train modality specific Sparse classifier (SRC), then perform score level fusion of the recognition results of all five modalities, or all the available modalities from the query video to obtain the multimodal recognition result. Experiments conducted on the constrained facial video dataset (WVU) and the unconstrained facial video dataset (HONDA/UCSD), resulted in a 99.17% and 97.14% rank-1 recognition rates, respectively. The multimodal recognition accuracy demonstrates the superiority and robustness of the proposed approach irrespective of the illumination, non-planar movement, and pose variations present in the video clips.

Biometric identification using Surveillance Video has attracted the attention of many researchers as it can be applicable not only for robust identification but also personalized activity monitoring. We present a novel multimodal recognition system that extracts Frontal Gait and Low Resolution face images from frontal walking surveillance video clips to perform efficient biometric recognition. The proposed study addresses two important issues in surveillance video that did not receive appropriate attention in the past. First, it consolidates the Model-Free and Model-Based Gait feature extraction approaches to perform robust gait recognition only using the frontal view. Second, it uses a low-resolution face recognition approach which can be

trained and tested using low-resolution face information. This eliminates the need for obtaining high-resolution face images to create the gallery, which is required in the majority of low-resolution face recognition techniques. Previous studies on frontal gait recognition incorporate assumptions to approximate the average gait cycle. However, we quantify the gait cycle precisely for each subject using only the frontal gait information. The approaches available in the literature use the high resolution images obtained in a controlled environment to train the recognition system. However, in our proposed system we train the recognition algorithm using the low resolution face images captured in the unconstrained environment. The proposed system has two components, one is responsible for performing Frontal Gait recognition and one is responsible for Low Resolution face recognition. Later, score level fusion is performed to fuse the results of the Frontal Gait recognition and the Low Resolution Face recognition. Experiments conducted on the Face and Ocular Challenge Series (FOCS) dataset resulted in a 93.5% Rank-1 for Frontal Gait recognition and 82.92% Rank-1 for Low Resolution face recognition, respectively. The score level multimodal fusion resulted in 95.9% Rank-1 recognition, which demonstrates the superiority and robustness of the proposed approach.

to my Parents

Acknowledgements

I would like to thank my advisor Dr. Shihab S. Asfour and my co-advisor Dr. Mohamed Abdel-Mottaleb for their encouragement, guidance, financial, and emotional support in the past few years through the research and completion of my degree. I believe their personality and technical capability was an indispensable factor for me to finish this endeavor.

My appreciation also extends to the U.S. Department of Energy for funding the University of Miami Industrial Assessment Center (MIIAC), which provided my doctoral assistantship and made available all the equipment and data used in my studies.

Sayan Maity

University of Miami

May 2017

Table of Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

There is an ever-growing need to automatically authenticate and identify individuals. The most commonly used biometric systems for identification, *e.g.*, fingerprint and face recognition, have already been commercialized. Recently, 3D ear biometrics received attention from the biometrics research community [1]. Ear possesses a number of inherent characteristics, *e.g.*, its shape is not affected by facial expressions, and it almost maintains its shape with aging, which makes its use advantageous [1, 2].

Several factors, e.g., changes in illumination and viewing direction affect the accuracy and robustness of unimodal biometrics [3–6]. To overcome these limitations, fusion of different modalities has been used to obtain robust and accurate recognition results in the literature. Additionally, there are several motivations for building multimodal biometric systems that work on facial video clips where some of the modalities are missing. Firstly, acquiring video clips of facial data is straight forward using conventional video cameras, which are ubiquitous. Secondly, the nature of data collection is non-intrusive and the ear, frontal, and profile face can appear in the same video. Thirdly, in a multi-modal biometric identification system, it is expected to encounter missing modalities when working with video data. Different modalities, *e.g.*, left ear, right ear, left profile face, right profile face, and frontal face might exist in the training video clips. If the test data does not contain all the modalities during the classification, we should be able to perform multi-modal classification based on the available modalities.

The importance of identifying and monitoring the activity of registered offenders using video surveillance footage has been proven effective on several occasions, *e.g.*, identifying the Boston bombing suspects, to lead the detectives in the right direction. However, the quality of the video data acquired by the surveillance system poses challenges. The primary causes of poor image quality recorded in most digital video surveillance systems are low resolution, excessive quantization, and low frame rate. Moreover, high-resolution video surveillance systems require excess storage space. These factors result in low-resolution biometric data, *e.g.*, face images, obtained from the video surveillance clips collected using the existing video surveillance systems.

The objective of this dissertation is to introduce novel methods for 3D ear segmentation and recognition, and using survillience video data for efficient multimodal recognition. Majority of the available ear recognition techniques in the literature use sequential search to find a match for the query template among all the gallery templates. Performance of sequential search is satisfactory when the gallery contains a small number of subjects, but it is not adequate for large galleries due to its inefficiency. To perform time efficient recognition, we propose a hierarchical categorization of the gallery. During recognition, we first determine the subcategory in which the query image belongs and then only search the gallery templates in that subset. Hence, the search space for a query image is reduced resulting in time efficient matching.

We proposed a novel multimodal biometrics approach to efficiently recognize subjects from facial video surveillance data irrespective of the illumination, non-planar movement, and pose variations present in the face video clips. Unlike facial videos recorded under a constrained environment, facial video clips collected in unconstrained environments contain significant head pose variations due to non-planar movements. Moreover, detected frames of the same modality from unconstrained facial video clips contain a high degree of non-planar rotation variabilities compared with the constrained counterpart. This makes unconstrained facial video clips more challenging to adequately extract information for efficient recognition.

We also proposed a novel automatic multimodal recognition framework for accurate human identification from low-resolution video surveillance footages by combining gait recognition and low resolution (LR) face recognition. We introduced an efficient Gait recognition technique through a robust Gait Cycle detection using only frontal Gait video clips. The approaches available in the literature for LR face recognition use the high resolution images obtained in a controlled environment to train the recognition system. However, in our proposed system we train the LR face recognition algorithm using the low resolution face images captured in the unconstrained environment.

In Chapter 2, we describe the proposed 3D ear segmentation technique. We use an active contour algorithm along with a tree structured graph to segment the ear region from the 3D profile images. Most of the proposed ear segmentation methods in the literature find the smallest possible bounding box that contains the ear and later use techniques like local surface matching to discard the outliers or nonear pixels, the other way is to segment out only the ear region pixels. As per the best of our knowledge, the only paper available in the literature, used 3D ear segmentation is [7], where coregistered 2D and 3D ear images are used to evaluate the ear shape. But the segmentation accuracy is not empirically mentioned, they used the shape to perform recognition. In the proposed approach, we perform segmentation through bounding box detection. The reason of applying segmentation to make sure that the features which will be extracted from that region only correspond to the ear.

In Chapter 3, we describe the 3D ear classification through Indexing. There are few publications that incorporate categorization and indexing with biometrics recognition. Among them only one study applied indexing on 3D ear gallery. To the best of our knowledge, the proposed approach is the first study in automatic 3D ear recognition to perform time efficient matching through categorizing the 3D ear gallery. The proposed technique uses a hierarchical categorization framework of the 3D ear gallery based on 2D and 3D features, respectively, extracted from 3D ear images.

The main contributions of our work are: 1) Time efficient recognition from 3D ear database using both 2D features and 3D features extracted from 3D ear images. We first categorize the 3D ear database in a hierarchical fashion using 2D and 3D features, respectively, and later perform the recognition. 2) The 3D ear gallery is indexed separately using two different indexing techniques, *e.g.*, KD tree and Pyramid technique, to compare their recognition accuracy and computation time. KD tree uses balanced split data structure and Pyramid technique uses unbalanced split data structure. Experimental results demonstrate that the efficiency of the proposed system outperforms the current state-of-the-art algorithms.

In Chapter 4, we propose a system which consists of three distinct components to perform the task of efficient multimodal recognition from facial video clips even in the presence of missing modalities. First, the object detection technique proposed by Viola and Jones [8], was adopted for the automatic detection of modality specific regions from the video frames. Unconstrained facial video clips contain significant head pose variations due to non-planar movements, and sudden changes in facial expressions. This results in an uneven number of detected modality specific video frames for the same subject in different video clips, and also a different number of modality specific images for different subject. From the aspect of building a robust and accurate model, it is always preferable to use the entire available training data. However, classification through sparse representation (SRC) is vulnerable in the presence of uneven number of modality specific training samples for different subjects. Thus, to overcome the vulnerability of SRC whilst using all of the detected modality specific regions, in the model building phase we train supervised denoising sparse autoencoder to construct a mapping function. This mapping function is used to automatically extract the discriminative features preserving the robustness to the possible variances using the uneven number of detected modality specific regions. Therefore, by applying Deep Learning Network as the second component in the pipeline results in an equal number of training sample features for the different subjects. Finally, using the modality specific recognition results, score level multimodal fusion is performed to obtain the multimodal recognition result.

Due to the unavailability of proper datasets for multimodal recognition studies [9], often virtual multimodal databases are synthetically obtained by pairing modalities of different subjects from different databases. To the best of our knowledge, the proposed approach is the first study where multiple modalities are extracted from a single data source that belongs to the same subject. There are a very few studies in biometrics recognition literature that deal with substantial head pose variation in facial video clips. It may however be noted that most of the previous studies were aimed to overcome the particular variabilities, *e.g.*, expression, viewing angle, and illumination, in different facial images by applying individual transformations. The main contributions of the proposed approach is the application of training a Deep Learning Network for automatic feature learning in multimodal biometrics recognition using a single source of biometrics i.e., facial video data, irrespective of the illumination, non-planar movement, and pose variations present in the face video clips.

In Chapter 5, we propose a solution for accurate human identification from low-resolution video surveillance footages by combining gait recognition and low resolution (LR) face recognition. The proposed system is a fully automatic platform which first extracts the frontal gait silhouettes and low resolution face images from the frontal walking video surveillance clips. Then obtains the feature vectors from the preprocessed frontal gait silhouettes, and the low resolution face images. Later the feature vectors are used to train two separate classifiers to perform the frontal gait recognition, and low resolution face recognition. Finally, the individual recognition results are fused through score level fusion. Given a test surveillance video clip of a subject walking towards the camera, first the gait features and LR face image features are extracted, later Nearest neighbor classifiers are used to separately obtain the Rank-1 frontal gait recognition and LR face recognition results. Finally, score level fusion is performed to fuse the individual recognition results.

Due to the unavailability of proper datasets for multimodal face and gait recognition, the proposed studies in the literature were evaluated only on databases with a small set of subjects [10, 11]. Moreover, majority of the approaches in the literature use lateral Gait view, or use camera calibration, or even require multiple cameras for capturing multiple Gait views to perform Gait recognition. Gait cycle detection is critical for Gait feature extraction and can be efficiently detected from the lateral Gait view. Majority of the studies on Gait recognition [12,13] perform the Gait Cycle detection using various heuristic approaches from biomechanics literature which is not practical for large databases. In a practical situation, a system which estimates the gait parameters from a single view, without depending on the subjects pose or on camera calibration is more realistic. We propose an efficient Gait recognition technique through a robust Gait Cycle detection using frontal Gait video clips. A considerable amount of literature has been published on low resolution face recognition. The majority of these studies use High Resolution (HR) images/ video to synthetically generate the corresponding low resolution counterpart. Then a mapping function is obtained between the High and Low resolution image pair. In this study we only use low resolution face information obtained from the video surveillance data to train and later test the performance of the proposed low resolution face recognition algorithm. To the best of our knowledge, the proposed approach is the first fully automatic multimodal recognition framework using LR face images and frontal gait silhouettes from Surveillance Video clips. Compared to other studies the performance is evaluated on a relatively large dataset.

1.1 Related Work

The remainder of this chapter provides a literary review of methods in 3D ear recognition, application of indexing tree in Biometrics recognition, and multimodal recognition. It is worth noting that direct comparison between the performances of the existing systems is difficult and at times can be misleading. This is due to the fact that biometrics datasets may be of different sizes, the image resolution and the amount of occlusion contained within the region of interest may be different, and some may utilize synthetically obtained dataset by pairing a subject from different databases consists of different modalities.

1.1.1 3D Ear Biometrics

The earliest research publications on ear biometrics used only 2D ear images. In recent years, researchers used either 3D ear or both 3D and co-registered 2D ear images for ear biometrics. Since the proposed approach uses only 3D ear images, we only consider 3D ear biometrics techniques for the literature review.

A two-step, off-line and online, ear detection technique [14] from 3D profile face images is proposed by Chen and Bhanu. First, in an off-line model template building step, a model template is built by averaging the shape index [15] histograms of multiple ear images. Later, in the on-line detection step, ears are detected from the 3D profile face images through template matching. In [16], Chen and Bhanu proposed a two-step iterative closest point (ICP) based approach to match 3D ear images. All the ear regions were manually extracted from the 3D face profile images. In the first step, the ICP algorithm is used to coarsely align the helix of the 3D test ear image with the model ear. Then, the ICP algorithm is iteratively used to obtain the best possible final alignment between the two ear images. Chen and Bhanu proposed a fully automatic 3D ear matching framework in [17], where they use 2D and 3D face profile images to automatically localize the ear helix and anti-helix region. They use two 3D shape representations of the ear, *i.e.*, local surface patch (LSP) representation for the salient feature points and a helix/anti-helix representation from the ear helix/anti-helix localization step. Later, a modified ICP algorithm is used to match the probe and gallery ear images.

In [18] and [19], Yan and Bowyer present an experimental evaluation to compare multiple ear recognition techniques, namely, eigen-ear method for 2D ear images, principal component analysis (PCA) using 3D ear images, Hausdorff matching of depth edge images obtained from 3D ear images, and ICP matching of 3D ear images. Among these techniques, ICP based matching have the best recognition accuracy and shows good scalability with the size of the gallery of ear images. In [20], Yan and Bowyer proposed a multimodal biometrics approach by combining single biometric recognition techniques in different combinations. Moreover, a fusion rule is proposed, based on the interval distribution between the rank-1 and rank-2 recognition to achieve better recognition accuracy compared with any other simple fusion rule. An ear biometrics system is proposed by Yan and Bowyer in [7], where coregistered 2D and 3D face profile images are used to segment the ear region after locating the the ear pit and nose tip. Later, using the segmented ear region, an ICP based shape matching algorithm is used to perform 3D ear recognition.

In [21] and [22], Cadavid and Abdel-Mottaleb, proposed a 3D ear recognition system using two different techniques, namely, structure from motion (SFM) and shape from shading (SFS) to reconstruct 3D ear models using frames in a video sequence. Ear region segmentation was performed through locating the ridges and ravines on the profile face image. Later, in the recognition step, the closest gallery ear model is found using ICP shape matching algorithm where the similarity measure is computed using an RMS. In [23] and [24], Zohu et al. proposed a generalized 3D object recognition using local and holistic feature matching, and evaluated the system using 3D ear recognition. First, a boundary box is detected around the ear region on the 3D profile face image, then, 3D ear matching is performed separately using extracted local and holistic features, respectively. Finally, the results of the local and holistic matching are fused through matching score fusion.

In [25], a 3D ear recognition technique is proposed through aligning the probe and gallery ear images using local 3D features computed around the salient points on the ear surface. Finally, ICP based matching algorithm is used to perform recognition. A coarse-to-fine 3D ear recognition technique based on ICP is proposed in [26], where an Adaboost classifier was used for ear region detection on 2D ear images, and then corresponding 3D ear regions are extracted from co-registered 3D face profile images. Later, ICP based matching algorithm is used to perform recognition. In [27], a rotation and scale invariant ear detection technique from 3D face profile images is proposed using graph connected components obtained from the edges of the 3D ear images. In [28] a two step 3D ear matching technique is proposed based on co-registered 2D and 3D ear images. In the first step, salient points are detected from 2D images and used for aligning 3D ear images. Later, a Generalized Procrustes Analysis and Iterative Closest Point (GPA-ICP) based matching technique is used to compare the 3D ear images that were coarsely alligned in the first step. In [29], Zhang et al. proposed a 3D Ear recognition framework using sparse representation. A template based ear detection technique is used to localize the ear contour, later, local 3D PCA-based feature descriptor [30] is used to represent the 3D ear for sparse classification.

1.1.2 Indexing in Biometrics Recognition

In [31], binary classification tree is utilized to build separate classes using voice and facial images. Classification using decision trees is very instable and sensitive to small changes in the data set which can result in different tree structures. To make the decision tree structure more robust, a random forest algorithm is used in [32], which has linear time complexity. However, in our proposed hierarchical categorization approach, the search is performed using the divide and conquer fashion in logarithmic time complexity.

To perform time efficient recognition of fingerprints, Henry [33] proposed a classification algorithm where fingerprints are indexed in a few classes based on geometric features. Ratha et al. [34] proposed a real time finger print matching engine for large fingerprint databases. Jiang et al. [35, 36] proposed a face biometrics database classification system, where faces are categorized using landmark based features. A few approaches in the literature were developed for efficient retrieval from databases of hand geometry [37, 38], iris [39] and signature [37, 40] using indexing techniques. In [41], KD tree is used to index a forty dimensional feature space populated from a multi-modal biometrics database consisting of two dimensional ear, face, iris and signature images. Gupta et al. [42] proposed an indexing method for two dimensional ear biometrics database using B+ tree. Chen et al. [40] indexed a 3D ear biometrics database using KD tree.

In this section, we compare our approach with the ones disussed in the previous section. In [41,42], the proposed techniques were evaluated only on a low dimensional feature space, computed from databases with relatively few subjects. In our method, we built a high dimensional feature space to compare the performance of different indexing techniques using the largest available 3D ear database (UND Collection J2). In [40], indexing is achieved based upon a lower dimensional feature space, later for an extensive one to one matching SVM-rank learning algorithm is used. It has a time complexity of $O(n^2)$, where n is the number of gallery templates. In our approach, after retrieving a small list of gallery templates, we sequentially match the query image with them, this results in a linear time complexity, $O(n)$, where n is the number of gallery templates retrieved after indexing. In our approach, the same feature vector used for indexing is used later for the extensive sequential comparison.

1.1.3 Multimodal Recognition

Research in face recognition has been active for the past few decades [43–47]. Although most of the work on face recognition is based on 2D images or 3D data, there are few publications that address video-based face recognition [48–51]. In [48], face images extracted from the training video clips are used to build a dictionary where face images of the same subjects containing variations in viewing angle, illumination, and facial expression, reside on the same nonlinear submanifold. Later, the learned dictionary is used to recognize faces from query video clips. Lee et al. [49] proposed probabilistic appearance manifolds, a spatio-temporal manifold model, which computes the transition probabilities between the subspaces. Given a query video, the probabilistic appearance manifold algorithm locates the operating part of the manifold to identify the subject. In [50], a view synthesis method is proposed to reconstruct a 3D frontal face model from multiple non-frontal 2D face images obtained from training video frames. Later, the synthesized frontal face image is used to match against the frontal face image extracted from the query video. In [51], decisions from multiple face matchers are adaptively fused to improve the performance of face recognition from facial video.

The ear, though a relatively new area of biometric research, possesses a number of inherent characteristics, *e.g.*, is not affected by facial expressions, and it almost maintains its shape with aging, which makes its use advantageous [52]. Because of these advantages, several researchers built multimodal ear and face biometric systems [53–55]. Pan et al. [55] proposed a feature fusion algorithm based on Kernel Fisher discriminant analysis (KFDA), and applied it to multimodal recognition based on two dimensional ear and profile face images. Kisku et al. [54] proposed a multimodal biometric system that fuses 2D face and ear biometrics using Dempster-Shafer decision theory. Boodoo et al. [53] proposed a multimodal biometric recognition system based on eigen ear and eigen face for two dimensional ear and profile face images. In [9], a Sparse Representation based multimodal biometric system is proposed. It fuses face and ear at the feature level, where the fusion weights are determined by computing the reliability of each modality.

1.1.4 Deep Learning in Biometrics

Recently, Deep Learning of Artificial Neural Networks has been used in several biometric authentication research studies. Ngiam et al. [56] proposed a Deep Network based unsupervised feature learning for audio-visual speech classification. The features obtained from the audio and video data is used to learn the latent relationship of the lip pose and motions in the video with the articulated phonemes in the audio. Different variants of Convolutional Neural Network [46, 57, 58] have been used to design face verification systems. A Convolutional Neural Network based Siamese Network is proposed in [58] for face verification. Taigman *et al.* [46] proposed a Convolutional Neural Network based face verification system, which significantly outperforms the existing systems on the Labeled Face in the Wild (LFW) database. The above-cited research articles are proposed for face verification, whereas our proposed approach deals with multimodal recognition in which, given a test video, it identifies the subject among many.

Gaurav *et al.* proposed MDLFace [59], a memorability based frame selection algorithm that enables automatic selection of memorable frames for facial feature extraction and matching, by using a deep learning algorithm. In [59], the deep learning algorithm is trained to identify the memorable faces, certain faces that can be more accurately remembered by human subjects as compared to other faces, to resemble the human perception in face recognition. In [60], a deep learning based anti-spoofing algorithm is proposed by using multimodal biometrics, *e.g.*, Iris, Face, and Fingerprint. In [61] a stacked supervised autoencoders-based Single Sample Face Recognition technique is proposed which achieves significantly higher recognition accuracy compared with other deep learning models, such as the deep Lambertian network.

1.1.5 Low Resolution Face Recognition

Even though research in face recognition has been active for the past few decades [43–47], the topic of Low-resolution [62] face recognition has only recently received much attention, for long distance surveillance applications, to recognize faces from small size or poor quality images with varying pose, illumination, and expression. Although the state-of-the-art face recognition accuracy using data collected in constrained environments is satisfactory, the recognition performance in real world applications such as video surveillance is still an open research problem, primarily due to low-resolution (LR) images [63] and variations in pose, lighting conditions and facial expressions.

The literature on low-resolution face recognition can be categorized into three broad classes:

1) Mapping into unified feature space: In this approach the HR gallery images and LR probe images are projected into a common space [64]. However, it is not straight forward to find the optimal inter-resolution (IR) space. Computation of two bidirectional transformations from both HR and LR to a Unified feature space usually incorporate noise.

2) Super-resolution: Many researchers used up-scaling or interpolation techniques, such as cubic interpolation on the LR images. Conventional up-scaling techniques usually are not good for the images with relatively lower resolution. However, Super-Resolution [65, 66] methods can be utilized to estimate HR versions of the LR ones

to perform efficient matching.

3) Down-scaling: Down-sampling techniques [63] can be applied on the HR images followed by comparison with the LR image. However, these techniques are poor in performance for solving LR problem, primarily because the downsampling reduces the high-frequency information which is crucial for recognition.

Due to the challenges and importance for real world applications, Low Resolution (LR) Face Recognition has gradually become an active research area of Biometrics in recent years. Ren *et.al.* [64] proposed a novel feature extraction method for face recognition from LR images, *i.e.*, coupled kernel embedding, where a unified kernel matrix is constructed by concatenating two individual kernel matrices obtained respectively from HR and LR images. Sumit *et.al.* [67], proposed an approach for building multiple dictionaries of different resolutions and after identifying the resolution of the probe image a reconstruction error based classification is obtained. A very low resolution (VLR) face recognition technique is proposed in [63] with a resolution lower than 16×16 by modeling the relationship between HR and VLR images using a piecewise linear regression technique. A super-resolution based LR face recognition technique is proposed by Jiangang *et.al.* [68], where an LR face image is split into different regions based on facial features and the HR representation of each section is learned separately. Jia *et.al.* [69] proposed a unified global and local tensor space representation, to obtain the mapping functions to acquire the HR information from the LR images to perform efficient LR face recognition.

1.1.6 Gait Recognition

Gait recognition [70, 71] is a well proven biometric modality, which can be used to identify a person remotely through inspecting their walking patterns. However, Gait recognition has some subtle shortcomings: it can be affected by the dressing attire, carrying large objects etc. Moreover, the physical state such as injuries can also affect a persons walking pattern. Majority of the proposed gait recognition techniques [72, 73] employ multi-view gait recognition to overcome the viewing angle transformation problem and to improve the recognition accuracy.

The first step of Gait Recognition is background subtraction. The feature extraction techniques in the literature [74] can be categorized broadly in two classes:

1) Model-Free approaches: In the Model-Free Gait representation [75], the features are composed of a static component, *i.e.*, size and shape of a person, and a dynamic component, which portrays the actual movement. Examples of static features are height, stride length and silhouette bounding box. Whereas dynamic features can include frequency domain parameters like frequency and phase of the movements.
2) Model-Based approaches: In the Model-based Gait representation approaches [71, 76] we need to obtain a series of static or dynamic gait parameters via modeling or tracking the entire body or individual parts such as limbs, legs, and arms. Gait signatures formed using these model parameters are utilized to identify an individual.

Model-free approaches are usually insensitive to the segmentation quality and less computationally expensive compared with the model-based approaches. However, the model-based approaches are usually view-invariant and scale-independent compared with the model-free counterpart.

To obtain the gait signature utilizing a sequence of gait silhouettes, Davis *et.al.* [77] proposed the motion-energy image (MEI) and motion-history image (MHI) which transform the temporal sequence of silhouettes to a 2D template for Gait identification. Later, Han and Bhanu [71] adopted the idea of motion-energy image (MEI) and proposed the Gait Energy Image (GEI) for individual recognition using Gait images. Frequency analysis of spatio-temporal Gait signals is used by researchers to model the periodical gait cycles. Lee *et. al.* [75] proposed a Model-Free approach to first divide the gait silhouette into seven regions and align them with ellipses, later apply Fourier Transform on the fitted ellipses to extract the magnitude and phase components for classification. Goffredo *et.al.* [78] proposed a k-nearest neighbor classifier (k-NN) for front-view Gait recognition where the Gait signature is composed of shape features extracted from sequential silhouettes.

1.1.7 Multimodal Face and Gait Recognition

The fusion of Face and Gait modalities have recently received significant attention [79], mainly motivated by their impact on security related applications. The fusion of the two modalities has been used in the literature to obtain more robust and accurate identification. The fusion can be performed at the feature/ sensor level, the decision level, or the matching score level.

In [80], features from high-resolution profile face images and features from gait energy images are extracted separately and combined at the feature level, later the fused feature vector is normalized and used for multimodal recognition. The experimental results on a database of video sequences for 46 individuals demonstrate that the integrated face and gait features result in a better performance than the performance obtained from the individual modalities. Shakhnarovich *et. al.* [10] proposed a view normalized multimodal face and gait recognition algorithm and evaluated it on a dataset of 26 subjects. First, the face and the gait features are extracted from multiple views and transformed to the canonical pose frontal face and the profile gait view, later the individual face and gait recognition results are combined at the score level. In [11], a score level fusion of face and gait images from a single camera view is proposed and tested on an outdoor gait and face dataset of 30 subjects. The results of a view-invariant gait recognition algorithm, and a face recognition algorithm based on sequential importance sampling are fused in a hierarchical and holistic fashion. Geng *et. al.* [81] proposed a context-aware multi-biometric fusion of gait and face which dynamically adapts the fusion rules to the real-time context and respond to the changes in the environment.

CHAPTER 2

3D Ear Segmentation

Automatic segmentation without user intervention is one of the most challenging problems in image processing and computer vision. The primary problem with ear segmentation is occlusion, which can occur because of long hair, over the ear headphones, ear rings or other objects.

When computing feature vectors to model the ear, if majority of the pixels are from non ear regions, the recognition accuracy is usually affected. Majority of the proposed ear segmentation methods [16, 17, 23] find the smallest possible bounding box that contains the ear. In this paper we aim to segment only the ear region pixels, which is harder than finding the ear bounding box. The reason of applying ear region segmentation, instead of bounding box detection, is to make sure that the extracted features will only represent the ear, and hence minimize the amount of errors in the ear description.

The proposed 3D ear segmentation approach [82] has two basic steps. The first step localizes the ear in the 3D profile facial image and detects landmarks on the curved ear boundary. Using these landmarks, a boundary box around the ear region is obtained on the 3D profile facial image. Later, these salient landmarks are used as

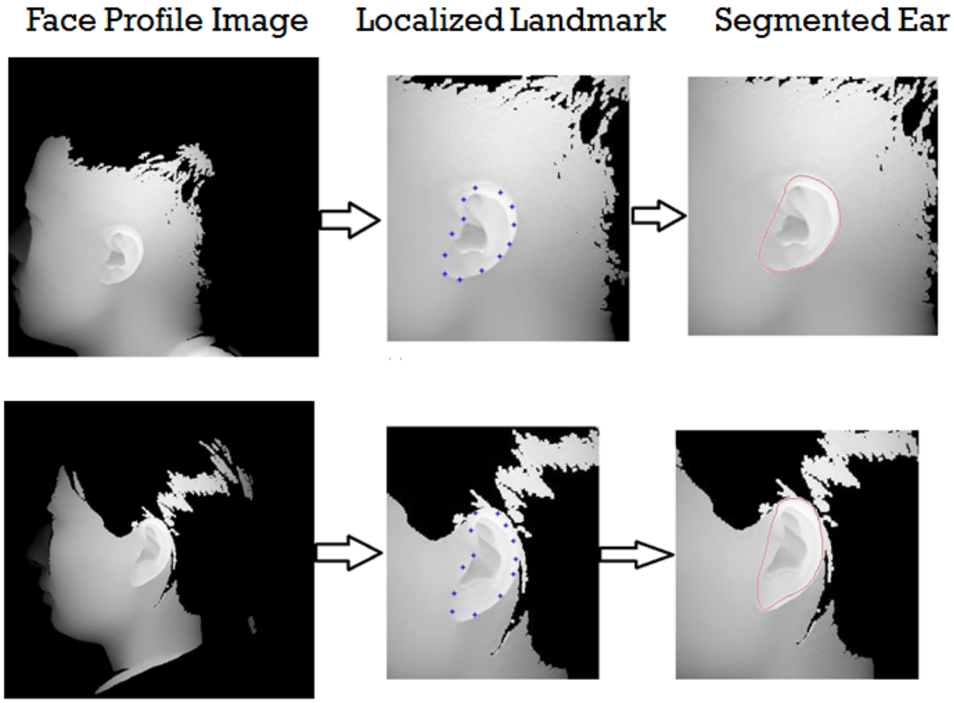


Figure 2.1: Two step 3D ear Segmentation results.

the initial starting points for an active contour model to segment the 3D ear region. The entire pipeline of the two step 3D ear segmentation is shown in Figure 2.1.

The remainder of this chapter is organized as follows: Section 2.1 lists the preprocessing steps performed in the 3D profile face images. Section 2.2 details the landmark localization technique. Section 2.3 describes the automatic segmentation of the 3D ear region from the profile range image. The perfromance evaluation of the proposed segmentation technique analyzed in section 2.4. The procedure of computing the co-occurrence relation and latent SVM modeling technique to detect the landmarks on the ear helix of 3D profile images is explained in section 2.5.

2.1 Preprocessing

Before using the raw depth scans of the profile face, some preprocessing steps are performed. The 3D data contains depth discontinuities which include spikes and holes. To remove spikes from the depth scans we apply median filter and to fill the holes we use cubic interpolation. Finally, to reduce noise and to smooth the 3D data, Gaussian Filter is used.

2.2 Landmark Localization

Utilizing the flexible mixture model introduced in [83], an elegant approach for ear landmark localization was proposed by Lei et al. [84,85]. In [84,85], the training was performed on the 2D color images that are registered to the 3D profile images, then the 2D ear detection results were applied to the 3D images. For the landmark localization step in our proposed ear segmentation technique, the idea in [84] is adapted with significant modifications. In our proposed approach, landmark localization is performed on the 3D profile images, which makes our approach applicable to 3D ear galleries that are collected without corresponding 2D images.

In [83], Yang et al. proposed a tree structured graph to represent the human body structure using each body part as a rigid component. Later a flexible mixture model is trained using latent Support Vector Machine (SVM) to capture the contextual co-occurrence relations among the body parts, preserving the local rigidity. The procedure of computing the cooccurrence relation and latent SVM modeling technique to detect the landmarks on the ear helix of 3D profile images is explained in the later scetions of this chapter.

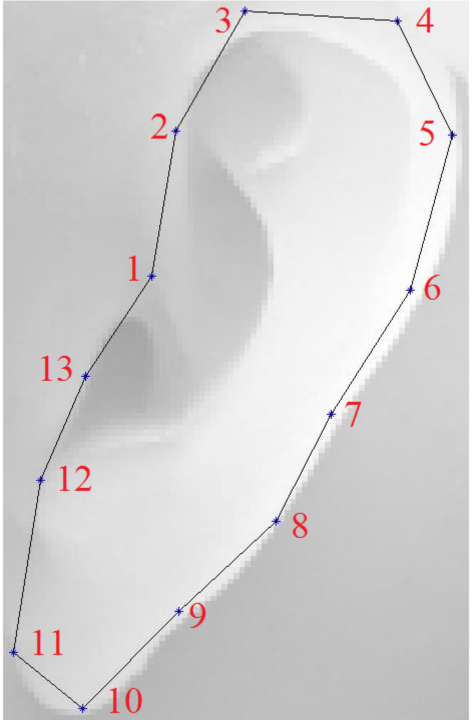


Figure 2.2: Tree Structured Graph

Ear helix along with helix rim and the anti-helix parts are the most easily recognized and distinguishable parts of the ear from different viewing angles. Using this structural advantage of the ear and incorporating the idea in [83], Lei et al. [84] implemented a flexible mixture model of the ear by defining 17 landmark points on the helix and anti-helix portions of the ear to find out the minimum bounding box area that contains the ear. To adapt this technique for our automatic ear-region segmentation, we positioned 13 landmarks only on the ear helix along with helix rim area, shown in Figure 2.2. The chosen landmark positions can be grouped into two groups, the first group (1, 2, 3, 10, 11, 12, 13) has fixed positions, while the second group (4, 5, 6, 7, 8, 9) has flexible positions. The anatomical nomenclature of the fixed position points are 1- Crus of Helix, 2- Triangular Fossa, 3- Crus of AntiHelix, 10- Lobule Tip, 11- Lobule, 12-Incisure Intertragica, 13- Tragus and the flexible landmarks 4-9

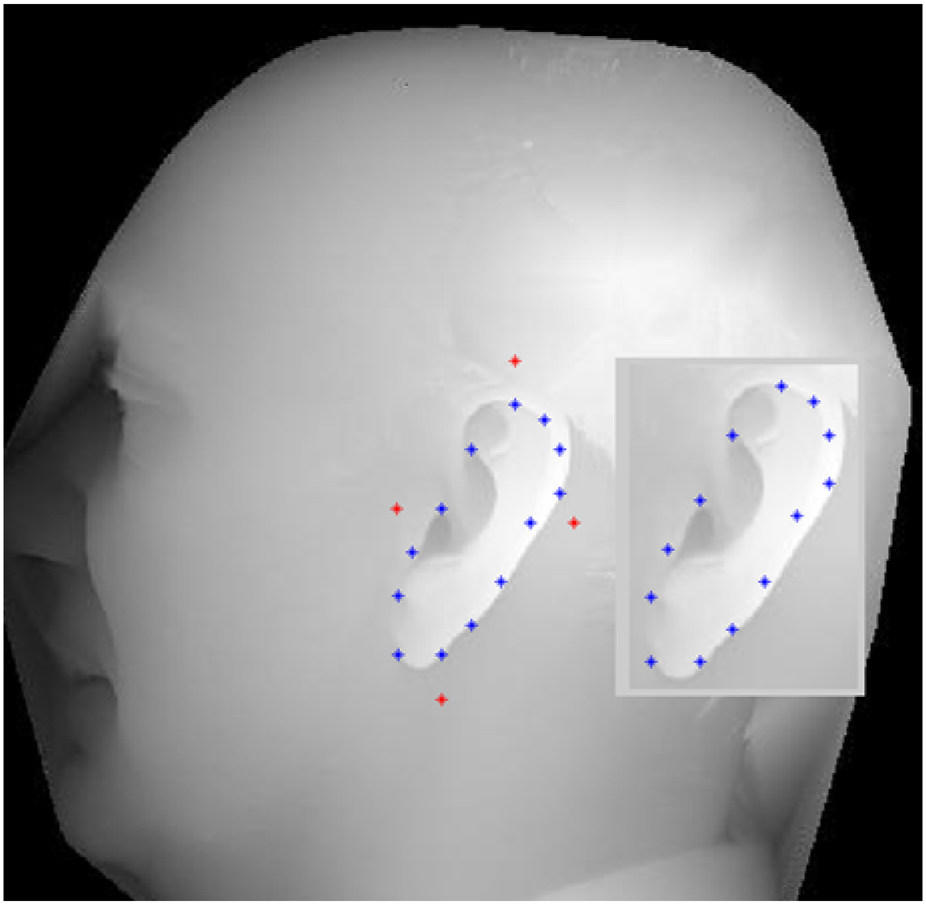


Figure 2.3: Example of Ear Landmark localization

are distributed uniformly on the helix of the ear. The tree structured graph, shown in Figure 2.2, is constructed with the 13 landmarks based on the flexible mixture model [83] to represent the human ear. This tree structured graph consists of the set of nodes that represent the landmarks, and the set of edges that connect them. Figure 2.3 shows an example result of the automatic ear landmark detection.

2.3 Ear-region Segmentation

To segment the ear-region from the 3D profile face image, the well known active contour algorithm, also known as snakes, proposed by Kass et al. [86] is used. Our goal is to find the edge of the ear and segment only the ear region by initializing the snake contour near the boundary of the ear using the landmarks that were localized in the previous step. Snakes [86], is an energy minimizing deformable spline guided by external constraint forces and influenced by image forces that pull it towards the contour of the object. Equation 2.1 represents the energy of a snake. This model was further improved by Cohen et al. [87] by incorporating a balloon model to provide more stable results when the image forces are instable. To apply this method, we need to initialize the snake near the contour of the object in the image, then the energy minimization steps will pull the snake to the contour in the rest of the iterations. The different energy terms, *i.e.*, E_{int}, E_{ext}, are responsible for the expansion and contraction of the spline. $(l(s))$ represents the location of the snake parametrically, *i.e.*, the closed curve from the initialization of the spline till the final spline formation, expressed as $(l(s)) = (x(s), y(s))$.

$$\begin{aligned} E_{snake} &= \int_0^1 E_{snake}(l(s))ds; \\ &= \int_0^1 (E_{int}(l(s)) + E_{ext}(l(s)))ds; \end{aligned} \tag{2.1}$$

where the internal spline energy can be formulated as proposed in [86]:

$$E_{int} = \int_0^1 \frac{1}{2}(\alpha|l'(s)|^2 + \beta|l''(s)|^2); \tag{2.2}$$

where α and β are the weights that control the snake's tension and rigidity, respectively. To attract the snake near the salient features in the image plane and lock on the closest possible edge, the energy functional E_{ext}, adapted from [86], can be expressed as

$$E_{ext} = E_{image} + E_{con}; \tag{2.3}$$

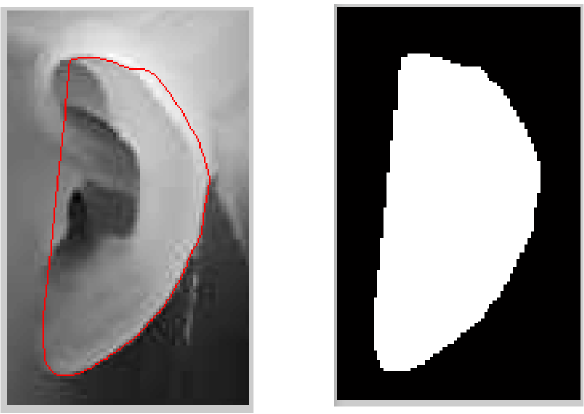


Figure 2.4: Example of Segmentation

where E_{image} is the total image energy and consists of different energy functionals computed from the image as explained in equation 2.4:

$$E_{image} = w_{line} \cdot E_{line} + w_{edge} \cdot E_{edge} + w_{term} \cdot E_{term}; \tag{2.4}$$

where w_{line}, w_{edge}, and w_{term} are the weights corresponding to the energy functionals: image intensity (E_{line}), the edge or image gradient (E_{edge}), and the terminations and corners of line segments (E_{term}), respectively, described in [86] and defined as follows:

$$\begin{aligned} E_{line} &= F(x,y); \\ E_{edge} &= -| \bigtriangledown F(x,y)|^2; \\ E_{term} &= \frac{C_{yy}C_x^2 - 2C_{xy}C_xC_y + C_{xx}C_y^2}{(C_x^2 + C_y^2)^{\frac{3}{2}}}; \end{aligned} \tag{2.5}$$

where $F(x,y)$ is the image intensity in the pixel location (x,y) on the image plane, and $C(x,y) = G_\sigma(x,y) \star F(x,y)$; represents a slightly smoothed version of the image, where G_σ is a Gaussian mask with standard deviation σ. Another component of the external energy functional, E_{con}, is the external constraint energy that generates a pressure force which pushes the curve outward so that it does not shrink to a point or a line, explained in [88], and is described as

$$
\begin{aligned}
E_{con} &= -k\overrightarrow{n}(P);\\
\overrightarrow{n}(P_i) &= \frac{|P_{i-1} - P_{i+1}|}{|Distance(P_{i-1}, P_{i+1})|};
\end{aligned}
\tag{2.6}
$$

where P_i is the ith point on the spline.

The energy functionals computed from the 3D scans are depth (z-dimensional) value (E_{depth}), shape index values (E_{S_I}) [15], and curvature values (E_{curv}) [15]. To achieve the best possible 3D ear segmentation result, experiments are conducted using these energy functionals in all possible combinations. The equal combination of the three energy functionals provided the best possible segmentation compared to using them separately or in any other combination. Thus in our application, E_{image} in equation 2.3 is replaced by E_{3D_image}, which can be represent as follows:

$$E_{3D\_image} = w_{depth} \cdot E_{depth} + w_{S_I} \cdot E_{S_I} + w_{curv} \cdot E_{curv}; \tag{2.7}$$

2.4 Evaluation of the Segmentation Results

To evaluate the performance of the proposed ear segmentation from 3D profile images, manual segmentation was performed for every 3D ear image, as shown in Figure 2.5. The pixels contained in the manual segmentation were used as reference pixels. If the number of ear pixels obtained by automatic segmentation include 5% or more pixels that are not in the reference pixel set, we label the image as over segmented. Similarly, if the number of pixels obtained by automatic segmentation exclude 5% or more pixels from those in the reference pixel set, we label the image

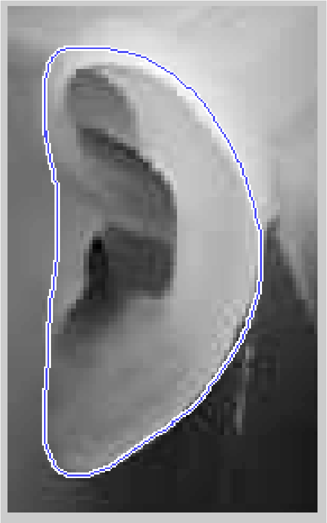


Figure 2.5: Example of manual segmentation

as under segmented. The performance of automatic segmentation using the proposed method on the 3D ear database of University of Notre Dame Collection J2 is shown in the Table 2.1. We achieve a 96.44% segmentation accuracy with a 5% pixel error rate. If we relax the pixel error rate up to 10%, the segmentation accuracy becomes 98.91%.

Comparison of the detection/segmentation accuracy of the proposed approach with the state-of-the-art techniques is shown in Table 2.2. In [26] authors detected a boundary box around the ear region with a detection accuracy of 99.9% using co-registered 2D and 3D profile face images, and in [27] the accuracy of boundary box detection is 99.38% using only 3D profile face images. Ear region segmenation is performed in [7] using co-registered 2D and 3D profile face images but no segmentation accuracy is reported. We should note that detecting a bounding box around the ear is an easier task than segmenting the ear region.

Table 2.1: Segmentation Accuracy

Total Dataset Images	Well Segmented	Over Segmented	Under Segmented
1800	1745/1800 (96.44%)	30/1800 (1.67%)	25/1800 (1.39%)

Table 2.2: Comparison of Detection/Segmentation Accuracy

Method	Mode of Detection	Modality of Images Used	Accuracy
Yan and Bowyer [7]	Segmentation	Co-registered 2D+3D	Not reported
Prakash and Gupta [27]	Boundary Box detection	Only 3D	99.38%
Islam et al. [26]	Boundary Box detection	Co-registered 2D+3D	99.9%
This work	Boundary Box detection detection based on the landmark detection	Only 3D	**100%**
This work	Segmentation	Only 3D	96.44% (5% pixel error rate) **98.91%** (10% pixel error rate)

2.5 Computing the Co-occurrence Relation and Latent SVM Modeling Technique to Detect the Landmarks on the Ear of 3D Profile Images

The mixture model described in [83] was used by Lei et al. [84] to capture contextual co-occurrence relations between the appearance of landmarks on ear images. Let T be the 3D ear image template, in which the pixel locations of the landmarks are denoted as p$_i$=T(x$_i$,y$_i$), where i=1,...,13. Based on the relative orientation of a landmark to its parent, it can be of different types, represented as or_i, where $i = 1, ..., K$. The score of a specific configuration of the flexible mixture model of the ear, utilizing the landmark locations and their relative orientation types, is computed using equation 2.8.

$$S(T, L, R) = \sum_{i \epsilon V} \lambda_i^{or_i} \cdot f(T, p_i) + \sum_{i,j \epsilon E} \mu_{ij}^{or_i or_j} \cdot \varphi(p_i, p_j) + \sum_{i,j \epsilon E} c_{ij}^{or_i or_j}; \tag{2.8}$$

The first term in equation 2.8 is an appearance model that controls image appearances at the landmarks, where $f(T, p_i)$ represent the feature extracted at the center of landmark location p_i, and $\lambda_i^{or_i}$ is a unary template for landmark p$_i$ adjusted for orientation type or_i. The second term in equation 2.8 is a deformation model that evaluates the relative position of each landmark pairs, where $\varphi(p_i, p_j)$ stands for the squared offset of the Histogram of gradients (HOG) feature between two landmark locations p_i, p_j can be represented as $\varphi(p_i, p_j) = [dx\, dx^2\, dy\, dy^2]$, $dx = x_i - x_j$, $dy = y_i - y_j$, and the weight parameter $\mu_{ij}^{or_i or_j}$ favors certain spatial placements. The third term in equation 2.8 is a co-occurrence model which favors certain pairs of orientations between landmarks, where $c_{ij}^{or_i or_j}$ is the pairwise co-occurrence prior between landmark p_i with orientation type or_i and landmark p_j with orientation type or_j. The algorithms applied for inference and learning the parameters from training data [83] are explained in the next two subsections.

2.5.1 Inference

Given the profile face image template T, the best scoring configuration of landmarks is found using dynamic programming by maximizing *S(T, L, R)* in equation 2.8 over landmark locations L, and orientation types R. The score of a specific landmark configuration, the first term in equation 2.8, is computed using message passing technique by iterating over all landmarks starting from the leaf nodes and moving upstream to the root node can be computed as mentioned in [83]:

$$score_i(p_i, or_i) = \lambda_i^{or_i} \cdot f(T, p_i) + \sum_{k \epsilon kids(i)} m_k(p_i, or_i); \tag{2.9}$$

where

$$m_k(p_i, or_i) = \underset{or_i}{max}\, c_{ij}^{or_i or_j} + \underset{p_i}{max}\, score(p_i, or_i) + \mu_{ij}^{or_i or_j} \cdot \varphi(p_i, p_j); \tag{2.10}$$

Here *kids(i)* are the set of children of landmark p_i in the tree structured graph G, where any landmark p_i can have only one parent. The message passed by any landmark p_i to its parent landmark p_j can be compute by using equation 2.9 and 2.10. After all messages passed to the root landmark p_1, $score_1(p_1, or_1)$ demonstrate the best configuration for landmark locations and orientation types. By using a predefined threshold score we can select the best landmark configuration results. Non-maximal suppression method is used to fuse overlapping decisions. By keeping track of the $argmax$ indices, the final location of the landmarks on the ear contour is detected.

2.5.2 Learning

In order to train the model, we use a fully supervised training dataset, where we are given $\{T_e, L_e, R_e\}$, labeled positive images with landmark locations, and orientation types, and negative images $\{T_e\}$. Since the scoring function is linear in its parameters, *i.e.*, $S(T_e, \gamma) = \omega \cdot \Psi(T_e, \gamma)$, where $\omega = (\lambda, \mu, c)$, defined in equation 2.8, the model can be learned using a structural SVM solver represented in equation 2.11 following the optimization procedure proposed in [83].

$$
\begin{aligned}
arg \min_{\omega,\xi_i \geq 0} \frac{1}{2}\|\omega\| + C\sum_{e} \xi_e; \\
s.t. \forall\ n \epsilon\, pos\ ,\ \omega\ \cdot \Psi(T_e, \gamma_e) \geq 1 - \xi_e \\
\forall\ n \epsilon\, neg\ , \forall \gamma\ ,\ \omega\ \cdot \Psi(T_e, \gamma_e) \leq -1 + \xi_e
\end{aligned}
\tag{2.11}
$$

CHAPTER 3

3D Ear Classification Through Indexing

Majority of the available ear recognition techniques in the literature use sequential search to find a match for the query template among all the gallery templates. Performance of sequential search is satisfactory when the gallery contains a small number of subjects, but it is not adequate for large galleries due to its inefficiency. To perform time efficient recognition, we propose a hierarchical categorization of the gallery [82]. During recognition, we first determine the sub-category in which the query image belongs and then only search the gallery templates in that subset. Hence, the search space for a query image is reduced resulting in time efficient matching.

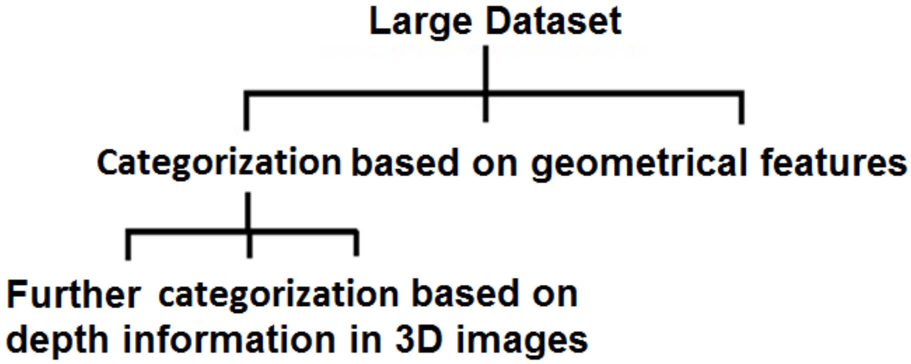

Large Dataset
Categorization based on geometrical features
Further categorization based on depth information in 3D images

Figure 3.1: Proposed categorical hierarchy

In this chapter we present a categorization framework that consists of a hierarchy as shown in Figure 3.1. Inspired by Iannarellis, ear shape classification [2], we classify ears into four basic geometric shape categories, *i.e.*, round, rectangular, triangular and oval, using geometric shape index values of the segmented ear. These four categories are further sub-categorized using depth features from the 3D ear images. While performing categorization based on the depth information, the feature space is partitioned using tree-based indexing techniques. During identification, only a small list of the gallery images that belong to the same class as the query image are retrieved. The block diagram of our approach is shown in Figure 3.2.

The remainder of this chapter is organized as follows: Section 3.1 describes the categorization of the 3D ear gallery based on shape. Section 3.2 presents the 3D feature extraction technique. Categorization of the 3D ear gallery through indexing is explained in Section 3.3. Section 3.4 provides the experimental results to demonstrate the performance of the proposed framework in terms of search space reduction and retrieval time compared with the traditional sequential search.

3.1 Shape Based Categorization

In addition to Iannarellis work on ear classification [2], Choars et al. [89,90] categorized ears based on their shape. We adapt the categories proposed by Iannarelli [2], *i.e.*, round, rectangular, triangular, and oval. To categorize 3D ears into the different shape categories shown in Figure 3.3, moments are computed from the binary mask of the ear obtained after ear-region segmentation, explained in the previous chapter.

Shape analysis is a complex problem due to presence of noise, and in certain cases variations between shapes result in insignificant changes in the measured fea-

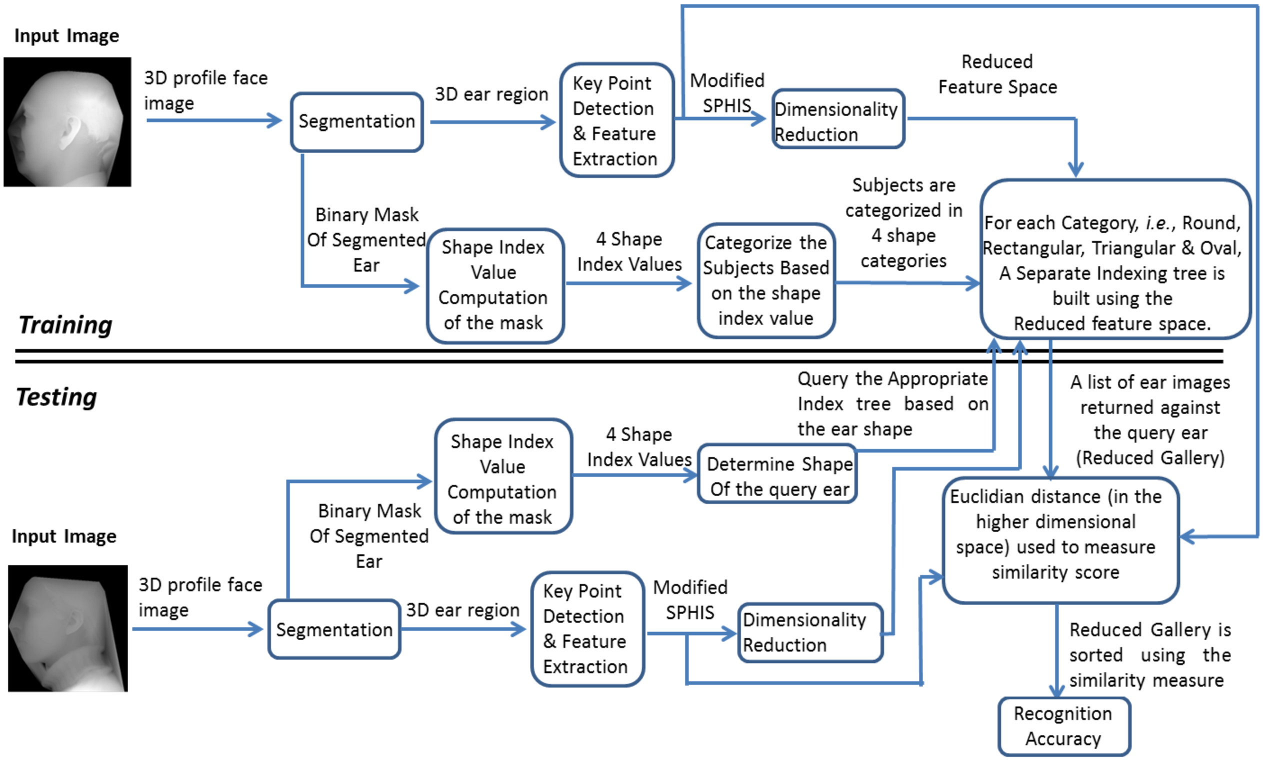

Input Image
3D profile face image
Segmentation
3D ear region
Key Point Detection & Feature Extraction
Modified SPHIS
Dimensionality Reduction
Reduced Feature Space
Binary Mask Of Segmented Ear
Shape Index Value Computation of the mask
4 Shape Index Values
Categorize the Subjects Based on the shape index value
Subjects are categorized in 4 shape categories
For each Category, i.e., Round, Rectangular, Triangular & Oval, A Separate Indexing tree is built using the Reduced feature space.
Training
Testing
Query the Appropriate Index tree based on the ear shape
A list of ear images returned against the query ear (Reduced Gallery)
Shape Index Value Computation of the mask
4 Shape Index Values
Determine Shape Of the query ear
Binary Mask Of Segmented Ear
Input Image
Euclidian distance (in the higher dimensional space) used to measure similarity score
3D profile face image
Segmentation
3D ear region
Key Point Detection & Feature Extraction
Modified SPHIS
Dimensionality Reduction
Reduced Gallery is sorted using the similarity measure
Recognition Accuracy

Figure 3.2: System block diagram: 3D ear classification

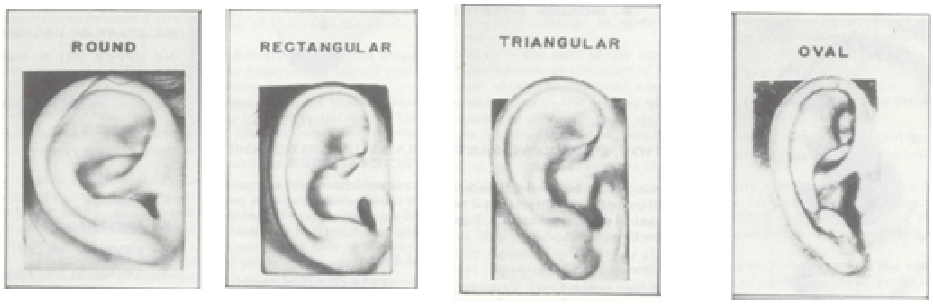


Figure 3.3: Iannarellis classification based on the shape of ear helix

ture values. To recognize objects from their shape, features such as eccentricity, Euler number, compactness and convexity are used in the literature [91]. Moments or central moments are used as quantitative measures for shape description [92, 93]. The translation, rotation, and scale invariant shape indexes such as circularity, rectangularity, triangularity, and ellipticity [94–96] are computed from the moments and their values lie within the range (0,1]. We used these shape descriptors to find the similarity of the ear shape to the geometrical figures: round, rectangular, triangular, and oval, respectively.

3.1.1 Circularity

For a given shape, S, the circularity measure [94] is defined as follows:

$$Cir_S = \frac{1}{2\pi} \frac{\mu_{0,0}(S)^2}{\mu_{2,0}(S) + \mu_{0,2}(S)}; \tag{3.1}$$

where μ refers to the general two-dimensional $(p+q)^{th}$ order moments of a density distribution function $\rho(x, y)$ defined as

$$\mu_{p,q} = \int \int_S x^p y^q dx dy; \quad p, q = (0, 1, 2); \tag{3.2}$$

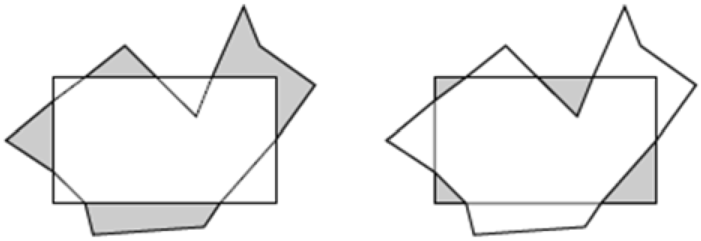


Figure 3.4: Shadowy region represent the calculation of 'R' on left and of 'D' on the right

where (x, y) represent the coordinates of the binary object $\rho(x, y)$, and Cir_S ranges over [0, 1]. For a circle, the ratio $\frac{\mu_{2,0}(S)+\mu_{0,2}(S)}{\mu_{0,0}(S)^2}$ results into $\frac{1}{2\pi}$. Hence, for a perfect circle the value Cir_S will peak at 1.

3.1.2 Rectangularity

The standard approach to measure rectangularity is to use the ratio of the area of the region of interest to the area of its minimum bounding rectangle (MBR) [97]. To overcome the sensitivity of the orientation between the rectangle and region of interest, a normalized rectangularity measurement proposed by Rosin [96] is defined as follows:

$$Rect_D = 1 - \frac{R + D}{B}; \tag{3.3}$$

where R is the difference between the rectangle and the region of interest, D is the difference between the region of interest and the rectangle, B is the area of the rectangle, and $Rect_D$ ranges over [0, 1]. The calculation of R and D is shown in Figure 3.4.

3.1.3 Ellipticity and Triangularity

The measures of ellipticity and triangularity are proposed by Rosin in [95]. Affine transformation of a circle result into an ellipse, and the affine moment invariants of a circle can be represented as:

$$I_1 = \frac{\mu_{2,0}(S)\mu_{0,2}(S) - \mu_{1,1}(S)^2}{\mu_{0,0}(S)^4}; \tag{3.4}$$

where μ is described in equation 5.19, and $\rho(x, y)$ is the characteristics function of the binary object whose ellipticity to be measured. For a unit radius circle, the value of I_1 is $\frac{1}{16\pi^2}$, the measure of ellipticity [95] is defined as follows:

$$Elp_I = \begin{cases} 16\pi^2 I_1, & \text{if } I_1 \leq \frac{1}{16\pi^2}. \\ \frac{1}{16\pi^2 I_1}, & \text{otherwise.} \end{cases}; \tag{3.5}$$

which ranges over [0, 1] and peaks at 1 for a perfect ellipse. A similar approach can be used to characterize triangles by moment invariants. Any triangle can be considered as a simple right triangle aligned with the axes after applying an affine transformation [95]. The computation of the affine moment invariants for a right triangle results in $I_1 = \frac{1}{108}$. The triangularity measure proposed in [95] is as follows:

$$Tri_I = \begin{cases} 108 I_1, & \text{if } I_1 \leq \frac{1}{108}. \\ \frac{1}{108 I_1}, & \text{otherwise.} \end{cases}; \tag{3.6}$$

Based on the maximum value among the four shape measures, the shape of a particular ear is determined. The 3D ear database of University of Notre Dame, Collection J2, contains multiple images of the same subject taken at different points in time under different illumination conditions. Table 3.1 contains the evaluation of the categorization of 3D ear database based on the above measures.

Table 3.1: Categorization evaluation

Shape Category	No. of Subjects with all training ears in same shape category	No. of Subjects having ears in different shape categories
Oval	179	7 (Rectangular,Round,Triangular)
Rectangular	118	3 (Oval,Round,Triangular)
Traingular	106	6 (Round)
Round	12	4 (Triangular)

If training ear images of a single subject are not classified in the same shape category, we label it as overlap between the shape categories in which the ears are distributed. The most overlap is found between rectangular and oval shaped ears and between triangular and round shaped ears. The goal of the proposed approach is to categorize the 3D ear gallery for quick retrieval. Thus, for a specific subject if there is such an overlap during enrollment, we store the data of that subject in each of the overlapped shape categories. This ensures that during recognition we do not misclassify a subject that have a shape measure that falls between two categories. For twenty subjects, out of a total of 415 subjects, their ear images were categorized in more than one shape category. For each one of these subjects, we store all the training ear images of that subject in each of the overlapping shape categories. Based on this approach the number of subjects in each category turns out to be 186 oval, 121 rectangular, 112 triangular and 16 round.

3.2 3D Feature Extraction

In this section we explain the 3D feature extraction technique. At first, key points which contain salient surface information are localized, then the Surface Patch

Histogram of Indexed Shape (SPHIS) descriptors [23,24] of these key points are computed to build the feature space. To reduce the redundancy and the dimensionality of the feature vectors, we applied principal component analysis (PCA) [98]. Finally, we apply indexing in the lower dimensional feature space using KD tree and pyramid technique, separately. In the following subsections we explain the extracted 3D features from the 3D ear images and the feature extraction technique.

3.2.1 Definition of Features

Surface curvature information is utilized as a prominent feature in 3D object recognition. Some of the important features include: 1) maximum (k_{max}) and minimum (k_{min}) principal curvatures, 2) Gaussian curvature (K) and 3) mean curvature (H) proposed by Besl et al. [99]. Using these curvature measures, two of the most prominent features in 3D object recognition, *i.e.*, Shape Index and Curvedness [15], are computed. Shape Index (S_I) is a quantitative measure of the shape of a surface at a vertex p and its value lies within the interval [0,1]. S_I at every point on the 3D surface can be computed as mentioned in [15]:

$$S_I(p) = \frac{1}{2} - \frac{1}{\pi} arctan(\frac{k_{max}(p) + k_{min}(p)}{k_{max}(p) - k_{min}(p)}); \tag{3.7}$$

where k_{max} and k_{min} are the principle curvatures of the surface point p defined as

$$\begin{aligned} k_{max}(p) &= H(p) + \sqrt{H^2(p) - K(p)}; \\ k_{min}(p) &= H(p) - \sqrt{H^2(p) - K(p)}; \end{aligned} \tag{3.8}$$

where $k_{max}(p) > k_{min}(p) \forall p$, and $H(p)$ and $K(p)$ are, respectively, the mean and Gaussian curvature at surface point p. The curvedness value [15] at a surface vertex is both rotation and translation invariant. Curvedness is a measure of the intensity of the surface curvature, which describe how gently or strongly the surface is curved. Mathematically, the curvedness at surface vertex p can be modeled as [15]:

$$C_v(p) = \sqrt{\frac{k^2_{max}(p) + k^2_{min}(p)}{2}}; \tag{3.9}$$

where k_{max} and k_{min} are the maximum and minimum principal curvature, respectively, described in equation 4.9.

3.2.2 Key Point Selection

Inspired by [26], Zhou et al. [24] proposed a sophisticated key point detection technique to select points with higher curvedness values compared with other points within a small neighborhood; those points are highly distinctive and contain salient surface information. A window of $1mm$ x $1mm$ is used to scan the segmented ear region, then, the point with the highest curvedness measure value within each small neighborhood region is selected as a key point. Local surface data around each key point is cropped using a sphere centered at the key point. To discard the insignificant key points chosen in the previous step, the data of the cropped neighborhood surface of the selected key points are examined. If the neighborhood of the selected key points contain any boundary points, it is discarded. To further reject the less discriminative key points, PCA [98] is applied on the cropped neighborhood surface data, and the eigenvalues and eigenvectors are computed to determine the discrimination power associated with each key point. The key points are only retained if the largest and the smallest eigen values satisfy the predefined thresholds in [24]. We adopt this key point selection approach.

3.2.3 Feature Extraction

The Surface Patch Histogram of Indexed Shape [24] (SPHIS) descriptor is calculated for each of the selected key points. SPHIS descriptor is designed to encode the shape information of the surface vertices based on the surrounding surface patch. SPHIS descriptor computation technique is adopted from [24] with few refinements.

To compute the rotation and translation invariant SPHIS descriptor for a key point on the surface, a surface patch is cropped using a sphere of radius r centered at the key point, we use $r = 14mm$ as proposed in [24]. Later, the points contained in the sphere are further divided into four subsets using equally spaced concentric spheres with radius $r_i = \frac{i \times r}{4}; i = 1, 2, 3, 4$ centered at the key point. After forming the four sub-surface patches, histograms are computed using the shape index and curvedness values of the points within each sub-surface patch resulting in 64 dimensional descriptors. Histograms computed for each sub-surface patch are then concatenated to generate a 256 dimensional SPHIS descriptor. To increase the discrimination potential, we add both shape index (S_I) and curvedness ($curv$) value of each key point along with the SPHIS descriptor proposed in [24] resulting in a 258 dimensional descriptor.

The final key points chosen after the key point selection step are used to compute the feature vector of the segmented 3D ear. Let kp be the number of selected key points. Each of the key points is represented by a descriptor of 258 elements which results in a $258Xkp$ dimensional feature space. Finally, the feature space dimension is reduced using PCA [98]. Indexing techniques are then applied on this reduced feature space.

3.3 Categorization Through Indexing

In our approach, categorization through indexing is performed in the second hierarchy level using the depth features, as shown in Figure 3.1. In this section we describe the indexing techniques that we used for categorization.

3.3.1 Indexing Techniques

Indexing is extensively used in the literature [100–102] for efficient content based retrieval from large databases. However, as the dimensionality of the feature space and the size of the database increase, the query response time along with the retrieval accuracy decrease. B-trees [103] and R-trees [104] are efficient for relatively low dimensions. In Biometrics, the feature space can consist of hundreds of features. To adopt an indexing technique that is suitable for indexing a large biometrics database, it should be able to index a high dimensional feature space and can handle range queries efficiently. Since we might need to enroll new subjects and if needed to delete old subjects from the biometrics database, the indexing technique must be dynamic and scalable. The indexing technique should be roughly height balanced or non-skewed to ensure that the tree traversal time is approximately the same for different queries.

The effect of indexing is to split the database to create an abstract ordering of the data points. There are two different paradigms to split a feature space, one is balanced and the other is unbalanced. Both methods have their advantages and disadvantages. The indexing algorithms that use balanced split data structures are computationally inexpensive compared to indexing algorithms that use unbalanced split data structures. In a balanced split data structure, the path from the root to any leaf node has the same length. However, when using an unbalanced split data structure to develop an efficient indexing technique, we need to use another data structure to achieve a height balanced or non-skewed indexing tree. In case of indexing algorithm that use balanced split structures, to solve the *curse of dimensionality* [105] problem, the feature space is recursively split into regions containing equal number of data points in $log_2 n$ dimensions where n is the number of data points. The pyramid indexing algorithm, which uses unbalanced split structure, partitions the D dimensional feature space into $2D$ hyperpyramids, i.e., the number of the hyperpyramids is double the dimension of the feature space. For comparison, we selected one indexing tree technique which uses balanced split data structure and another one which uses unbalanced split data structure and compared their matching accuracy and retrieval performance.

3.3.2 KD Tree

We selected the KD tree [106], which is a balanced split data structure to index a database. KD trees are abstractions of binary search trees for higher dimensional databases. Unlike R-trees [104], R*-trees [107] and X-trees [108], KD trees [106] have no overlap between nodes. A KD tree is formed by a recursive sub-division of the feature space using a $(D - 1)$ dimensional hyper-plane at every node, where D is the dimension of the feature space. After performing the splitting operation at every node, the points to the left of this hyperplane are represented by the left subtree of that node and the points to the right of the hyperplane are represented by the right subtree.

Before indexing we reduce the dimensionality of the feature space using PCA [98]. While indexing, at first, the feature space is recursively split across the top $log_2 n$ dimensions. This results in a tree structure of depth $log_2 n$. After the formation of the KD tree, range search is performed using the feature vector computed from the query 3D ear image. The range query retrieves the list of the gallery images which belong to the hyper rectangle that includes the query. This list of gallery images are labeled as reduced gallery. The average complexity to perform a range search in a KD tree consisting of N nodes in a D dimensional feature space is $O(D \cdot N^{1-\frac{1}{D}})$. However, if we need to add new data points in the KD tree when enrolling a new subject in the database we need to traverse the tree, starting from the root and moving to either the left or the right child depending on whether the point to be inserted is on the 'left' or 'right' side of the splitting plane.

3.3.3 Pyramid Technique

The pyramid technique was proposed by Berchtold et al. [109] to overcome the limitations of high dimensional database indexing using unbalanced split data structures. Compared with indexing algorithms that use balanced split data structures, the number of gallery images retrieved against a query image is much less when using the pyramid technique. To split the feature space, the pyramid technique [109] incorporates a spatial hashing which maps every data point in the original D dimensional feature space to a single dimension key. Later, using these single dimension keys, the data points are indexed in a height balanced $B+$ tree data structure resulting in efficient insert, delete and query operations. Before indexing with the pyramid technique, we first, reduce the dimensionality of the feature space using PCA [98],

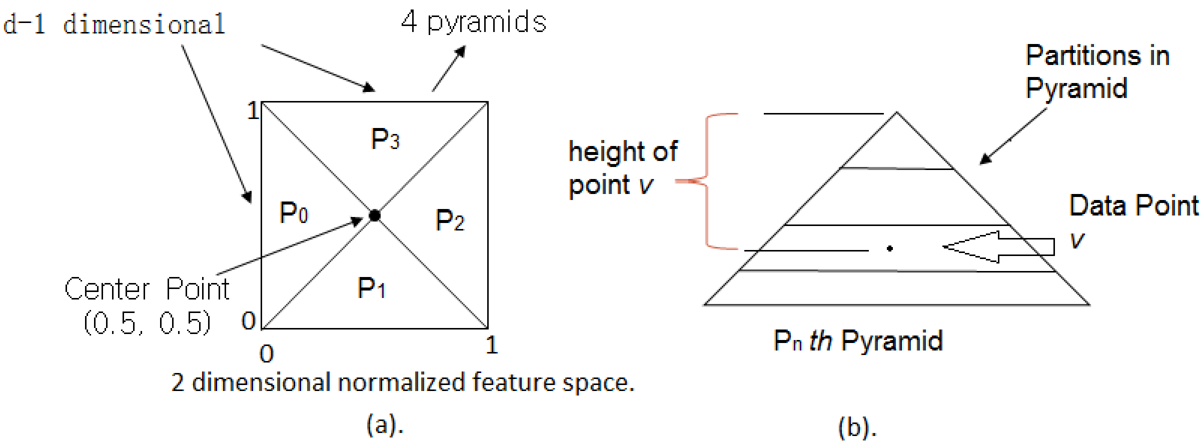


Figure 3.5: Partitioning a two dimensional feature space in pyramids. (a) 2 dimensional normalized feature space. (b) P_nth Pyramid

then normalize the feature space. The D dimensional feature space is partitioned into $2D$ hyperpyramids [109], each having a base of $(D-1)$ dimension, with the apex of each hyperpyramid meeting at the center of the normalized feature space. An example of partitioning a two dimensional feature space is shown in Figure 3.5 (a). where the normalized feature space is split into 4 pyramids p_0, p_1, p_2, and p_3. In the following subsections the key generation technique, and the technique for performing range queries on the Pyramid tree are explained.

3.3.3.1 Key Generation

The key associated with each data point in the multidimensional feature space is computed as the pyramid value of the point. The pyramid value consist of two parameters, *i.e.*, $pyramid_{value} = pyramid_{number} + height$. Computation of pyramid value is explained in Algorithm 1, adopted from [109]. The $pyramid_{number}$ is the number of the pyramid in which the data point belongs and the height is the distance of the data point from the apex of the pyramid. In Figure 3.5 (b). the data point

$'v'$ belongs to pyramid $'n'$ and the height of the data point is shown at the left of the figure. The $pyramid_{number}$ in which a data point belongs can be computed as [109]:

$$pyramid_{number} = \begin{cases} dim_{max}, & \text{if } dim_{max} < 0.5 \\ dim_{max} + D, & \text{if } dim_{max} \geq 0.5 \end{cases} \tag{3.10}$$

$dim_{max} = dim_i | (\forall dim_j, 0 \leq (dim_i, dim_j) < D,$

$dim_i \neq dim_j : |0.5 - dim_i| \geq |0.5 - dim_j|).$

This transformation is not injective, *i.e.*, two points v^1 and v^2 may have the same $pyramid_{value}$, which does not turn out to be an obstacle as we do not need the inverse transformation. Using the $pyramid_{value}$ as the key, the D dimensional data point is inserted into the B+ tree data structure.

Algorithm 1 Calculate the pyramid value, *i.e.*, the key associated to each data point

```
Pyramid Value or Key (Point $v$)
$dim_{max} = 0$
$height = |0.5 - v[0]|$
for $i = 1 \to D - 1$ do
  if $height < |0.5 - v[i]|$ then
    $dim_{max} = i$
    $height = |0.5 - v[i]|$
  end if
end for
if $v[dim_{max}] < 0.5$ then
  $j = dim_{max}$
else
  $j = dim_{max} + D$
end if
$pyramid_{value} = j + height$
return $pyramid_{value}$
```

3.3.3.2 Processing Range Queries

Pyramid trees respond to all possible types of queries such as point queries, range queries, and kNN queries. When querying, first the SPHIS descriptor is computed for the query 3D ear image, then the feature dimension is reduced appropriately using PCA [98]. The D dimensional interval used for the range query $[qr_{0_{min}}, qr_{0_{max}}], ..., [qr_{(D-1)_{min}}, qr_{(D-1)_{max}}]$ defines the hyper rectangular neighborhood of the query image. The query will return the list of the gallery images within this hyper rectangle. To perform the range query, first, the pyramids that intersect with the query hyper rectangle need to be determined. Later, the search interval $[h_{high}\ to\ h_{low}]$, *i.e.*, the range to be traversed in the $(D-1)$ dimensions inside the intersected pyramids, needs to be computed, shown in Figure 3.6. Hence, the D dimensional range query is transformed into D one dimensional range queries, containing two parameters, intersected $pyramid_{value}$ and the $[h_{high}\ to\ h_{low}]$ search intervals within the intersected pyramids. An example of the range query transformation for a two dimensional feature space is shown in Figure 3.6. The gray rectangle, in the two dimensional feature space, represents a range query that only intersects with pyramid p_0 and p_1, shown in Figure 3.6 (a). Hence, other pyramids, p_2 and p_3, do not need to be traversed while performing the range query. In Figure 3.6 (b)., the shaded area represents the query traversal region obtained based on the search ranges for the intersected pyramids.

3.3.4 Extended Pyramid Technique

In the Pyramid technique [109], the data partitioning approach is based on the assumption that the data points are uniformly distributed in the feature space. Usu-

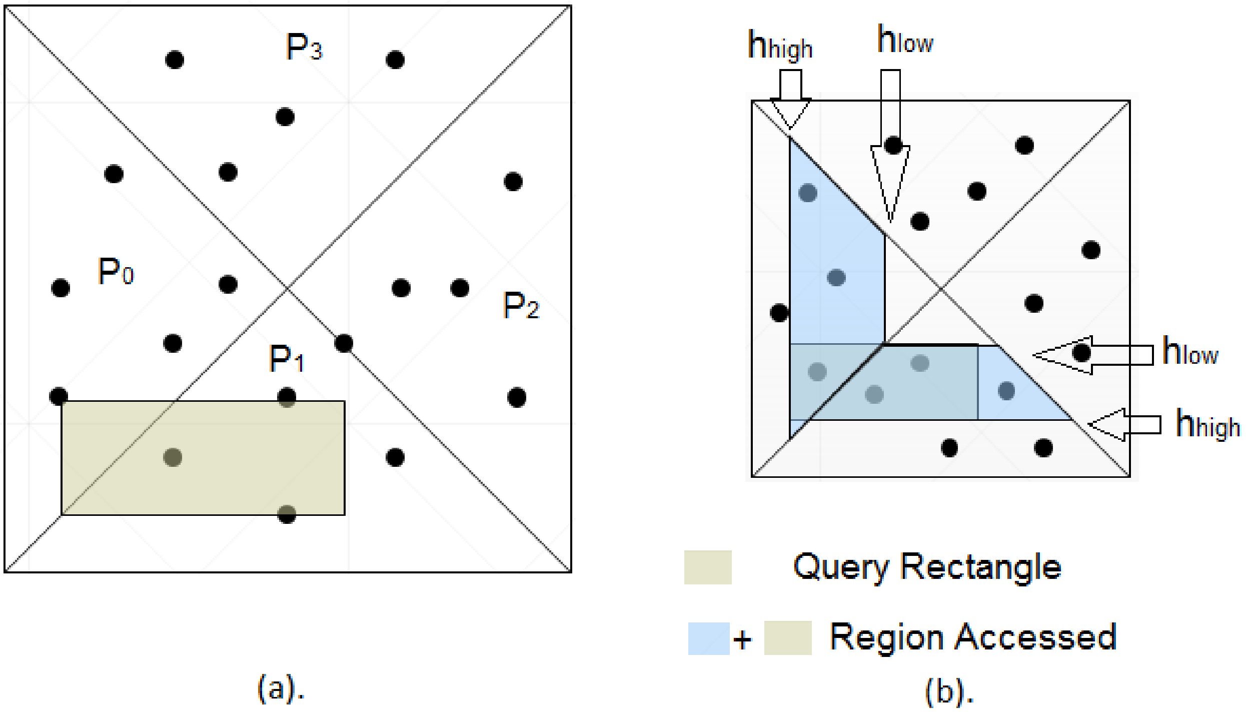


Figure 3.6: Two dimensional range query transformation. (a) Search Range. (b) Query traversal area.

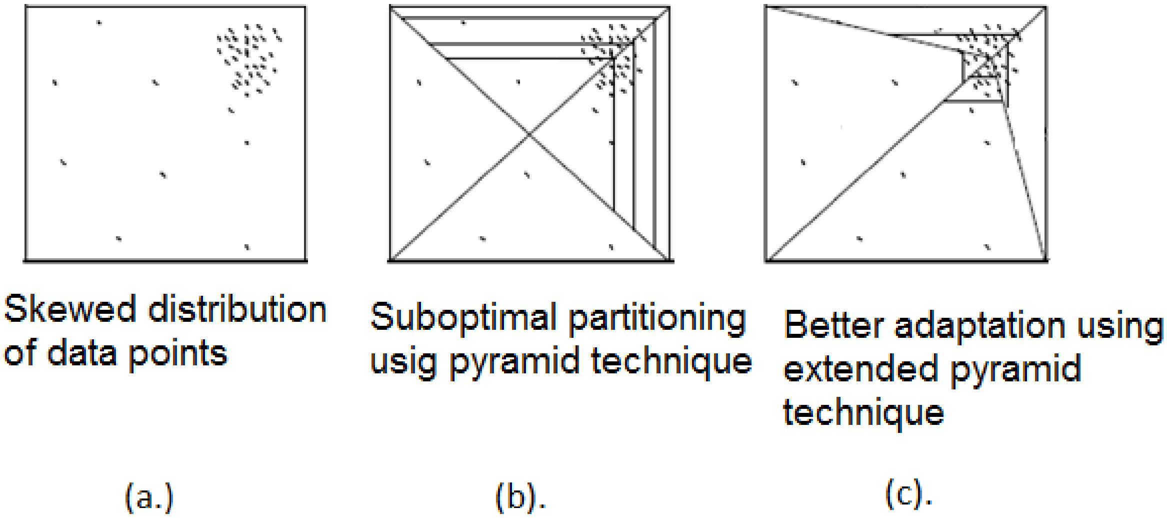

Skewed distribution of data points
Suboptimal partitioning usig pyramid technique
Better adaptation using extended pyramid technique
(a.)
(b).
(c).

Figure 3.7: Extended pyramid technique.(a) Skewed distribution of data points. (b) Suboptimal partitioning using pyramid technique. (c) Better adaptation using extended pyramid technique.

ally in real applications, the data distribution in the Euclidean space is not uniform and might be skewed, as shown in Figure 3.7 (a). Therefore, partitioning the feature space using the traditional pyramid technique will result into suboptimal partitioning as shown in the Figure 3.7 (b). To overcome this shortcoming, in the extended pyramid technique [109], the center point of the normalized feature space is readjusted. Shifting the center point from $(0.5, 0.5, \ldots, 0.5)_D$ to the median of the D dimensional feature space, the partitioning of the data points among $2D$ hyperpyramids will be more uniform, as shown in the Figure 3.7 (c). To find the median of D dimensional feature space we adapted the technique proposed in [110], which has linear complexity, $O(n)$, where n is the number of data points in the distribution.

3.4 Experimental Results

Experiments are conducted on the University of Notre Dame Collection J2 dataset with 1800 images of 415 subjects. Some of the subjects have only two face profile images in the database. Therefore, to include all the available subjects in the 3D ear database, we randomly select one image for training and another for testing. In the training phase, the binary ear masks obtained after segmentation are used to categorize the 3D ear gallery into four shape categories, results are shown in Table 3.2. Then, for each of these categories, separately, we build the feature space using the SPHIS descriptor illustrated in section 3.2.3. Using the 3D key point selection technique, mentioned in section 3.2.2, we computed the minimum number of robust key points on the segmented ears for the entire gallery, which turns out to be 35 key points. As explained in section V each of the key points is represented by a descriptor of 258 elements. Thus, the feature vector for every 3D ear image contains $258 \times 35 = 9030$ elements. To reduce the redundancy and the dimensionality of the feature vector, we applied PCA [98] to reduce the feature vector of each ear image to a 500 dimensional feature vector.

Table 3.2: Recognition accuracy at different search space reductions

Indexing Algorithm	10% reduction	20% reduction	30% reduction	40% reduction	50% reduction
KD-tree	89.26%	86.74%	85.25%	82.71%	80.81%
Pyramid technique	**93.97%**	92.77%	92.19%	91.5%	90.78%
Extended Pyramid	91.1%	90.62%	90.00%	88.25%	87.91%

To evaluate the performance of indexing the biometrics database, using algorithms based on balanced split and unbalanced split data structures, we used KD tree and the

Table 3.3: Average Computation Time / Query (seconds) at different search space reductions

Indexing Algorithm	10% reduction	20% reduction	30% reduction	40% reduction	50% reduction
KD-tree	0.0171	0.0165	0.0158	0.0142	0.0119
Pyramid technique	0.0089	0.0087	0.0082	0.0072	0.0054
Extended Pyramid	0.0065	0.0058	0.0051	0.0045	**0.0039**

Table 3.4: Comparison of time needed in recognition phase

Method	Average computation time / Query (seconds)	Ear Detection
Jindan et al. [24]	.019 s	Automatic
This Work	**.0039 s**	Automatic

Pyramid technique, separately, to index the 3D ear scans in the four shape categories. The query image is first segmented, then the shape index values are calculated using the image moments mentioned in the previous chapter. Based on the maximum shape index value, the shape category is determined and the test image is used to perform a range query on the index tree built for the same shape category. The indexing tree returns a small list of 3D ear images in response to the query image, that we label as reduced gallery. Euclidean distance in higher dimensional feature space (SPHIS descriptors without dimensionality reduction) is used to perform a sequential search on the reduced gallery of 3D ear images to find the best possible match. If the best possible matched ear is of the same subject we consider it as a rank-1 recognition.

To evaluate the robustness of the retrieval performance using the proposed approach, we conducted the recognition without categorizing the database. The average computation time to perform the recognition through sequential search of the entire gallery templates is around 0.023 seconds. The results of recognition accuracy and computation time when performing recognition after indexing the database at different amounts of search space reduction are given in Table 3.2 and 3.3 respectively.

The results show that the performance of the indexing technique with unbalanced split data structure, *i.e.*, Pyramid tree, is much better than that of the indexing technique with balanced split data structure, *i.e.*, KD tree, in both the recognition accuracy and the computation time. The extended pyramid technique proves to be superior over the traditional pyramid technique in time efficient search due to its adaptability to any arbitrary distribution. In the recognition phase the average computation time per query in the proposed approach is only 0.0039 seconds with a 50% search space reduction on a Windows® 7 operating system with Intel® Core™ i5 processor running a Matlab® implementation. In [24], the comparison of 3D ear probe-gallery matching time shows that the key point based matching technique [24] is faster than the ICP-based shape registration and matching technique [7, 17]. To compare the average computation time per query in our approach with the most efficient technique in the state-of-the-art [24], we run the 3D ear recognition approach proposed in [24] on the same platform. In Table 3.4 we compare the running time of our approach with the running time needed for recognition in the state-of-the-art 3D ear biometric system based on the UND 3D ear database. The results in the table demonstrate the superiority and robustness of our approach compared to the state-of-the-art technique.

3.4.1 Comparison with Other Methods

Table VII shows a comparison between the identification performance achieved by the proposed approach without any search space reduction and the recognition accuracy of the the state-of-the-art techniques on the UND, collection J2 database. In [17] the authors experimented with the UND database Collection F, a subset of Collection J2. The probe and gallery ear images used in [17] consist of a single 3D ear model for each of the 302 subjects. Other works in Table VII employed UND, collection J2 database to report their rank-one recognition accuracy. The proposed 3D ear biometric system achieves a rank-one recognition rate of 98.5% on the 415 subjects of UND collection J2 database.

Table 3.5: Comparison of Rank One Recognition accuracy

Method	Rank One recognition accuracy	Modality of Images Used
Chen and Bhanu. [17]	96.4%	Co-registered 2D +3D
Yan and Bowyer. [7]	97.6%	Co-registered 2D +3D
Jindan et al. [24]	98.0%	Only 3D
Prakash and Gupta. [28]	98.30%	Co-registered 2D +3D
This Work	**98.5%**	Only 3D

3.4.2 Recognition Accuracy Using Accurately Segmented Ears

The results in sections 3.2, 3.3, 3.4, and 3.5 use all the images of the 415 subjects in the UND, collection J2, 3D ear database without any post segmentation processing. Later, we performed an experiment to compute the recognition accuracy through indexing by first manually correcting the over and under segmented 3D ear regions, as defined in the previous chapter. By using the accurately segmented ear regions of all the 415 subjects, those who have at least two ear images, the recognition results obtained through indexing improved compared to using the segmented ears, without correction, as shown in Table 3.6. We used both Pyramid tree and KD tree to compute the rank-one recognition accuracy with a 10% to 50% search space reduction with a 10% step size. The best rank-one recognition accuracy increased to 96.87% when using the Pyramid tree indexing technique with a 10% search space reduction.

Table 3.6: Recognition Accuracy Using Accurately Segmented Ears

Indexing Algorithm	10% reduction	20% reduction	30% reduction	40% reduction	50% reduction
KD-tree	91.32%	90.60%	89.64%	88.43%	86.74%
Pyramid technique	**96.87%**	96.14%	95.18%	94.12%	93.49%

CHAPTER 4

Multimodal Biometrics Recognition from Facial Video via Deep Learning

Several factors, e.g., changes in illumination and viewing direction, affect the accuracy and robustness of unimodal face biometrics [3–6]. To overcome these limitations, fusion of different modalities has been used in the literature to obtain robust and accurate recognition results.

There are several motivations for building multimodal biometric systems that work on facial video clips where some of the modalities are missing. Firstly, acquiring video clips of facial data is straight forward using conventional video cameras, which are ubiquitous. Secondly, the nature of data collection is non-intrusive and the ear, frontal, and profile face can appear in the same video. Thirdly, in a multi-modal biometric identification system, it is expected to encounter missing modalities when working with video data. Different modalities, *e.g.*, left ear, right ear, left profile face, right profile face, and frontal face might exist in the training video clips. If the test data does not contain all the modalities during the classification, we should be able to perform multi-modal classification based on the available modalities.

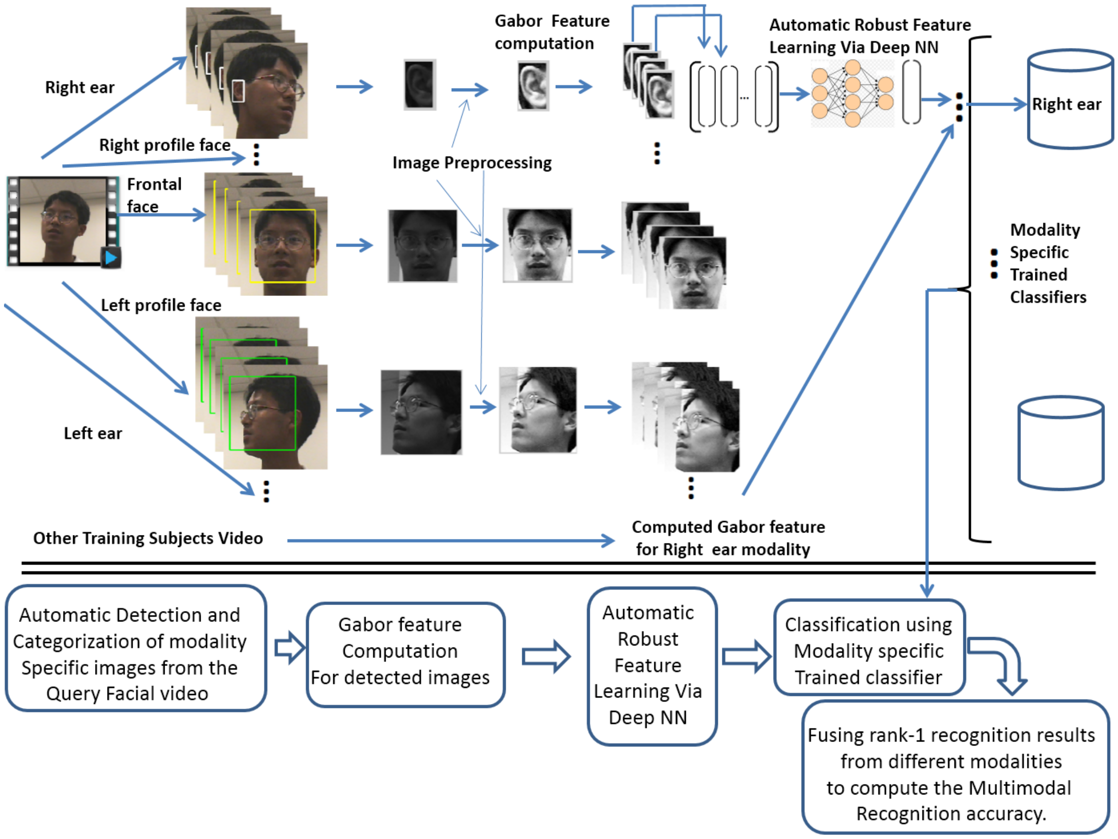

Right ear
Right profile face
Frontal face
Left profile face
Left ear
Gabor Feature computation
Image Preprocessing
Automatic Robust Feature Learning Via Deep NN
Right ear
Modality Specific Trained Classifiers
Other Training Subjects Video
Computed Gabor feature for Right ear modality
Automatic Detection and Categorization of modality Specific images from the Query Facial video
Gabor feature Computation For detected images
Automatic Robust Feature Learning Via Deep NN
Classification using Modality specific Trained classifier
Fusing rank-1 recognition results from different modalities to compute the Multimodal Recognition accuracy.

Figure 4.1: System Block Diagram: Multimodal Biometrics Recognition from Facial Video

In this chapter [111, 112], we present a novel multimodal biometrics approach to efficiently recognize subjects from facial video surveillance data irrespective of the illumination, non-planar movement, and pose variations present in the face video clips. Unlike facial videos recorded under a constrained environment, facial video clips collected in unconstrained environments contain significant head pose variations due to non-planar movements. Moreover, detected frames of the same modality from unconstrained facial video clips contain a high degree of non-planar rotation variabilities compared with the constrained counterpart. This makes unconstrained facial video clips more challenging to adequately extract information for efficient recognition.

The remainder of this chapter is organized as follows: Section 4.1 details the modality specific frame detection from the facial video clips. Section 4.2 describes the automatic feature learning using supervised denoising sparse autoencoder (deep-learning). Section 4.3 presents the modality specific classification using sparse representation and multimodal fusion. Section 4.4 provides the experimental results on the constrained facial video dataset (WVU [113]) and the unconstrained facial video dataset (HONDA/UCSD [114]) to demonstrate the performance of the proposed framework.

4.1 Modality Specific Image Frame Detection

To perform multimodal biometric recognition, we first need to detect the images of the different modalities from the facial video. The facial video clips in the constrained dataset are collected in a controlled environment, where the camera rotates around the subject's head. The video sequences start with the left profile of each subject (0 degrees) and proceed to the right profile (180 degrees). Each of these video sequences contains image frames of different modalities, *e.g.*, left ear, left profile face, frontal face, right profile face, and right ear, respectively. The video sequences in the unconstrained dataset contains uncontrolled and nonuniform head rotations and changing facial expressions. Thus, the appearance of a specific modality in a certain frame of the unconstrained video clip is random compared with the constrained video clips.

The algorithm was trained to detect the different modalities that appear in the facial video clips. To automate the detection process of the modality specific image frames, we adopt the Adaboost object detection technique, proposed by Viola and

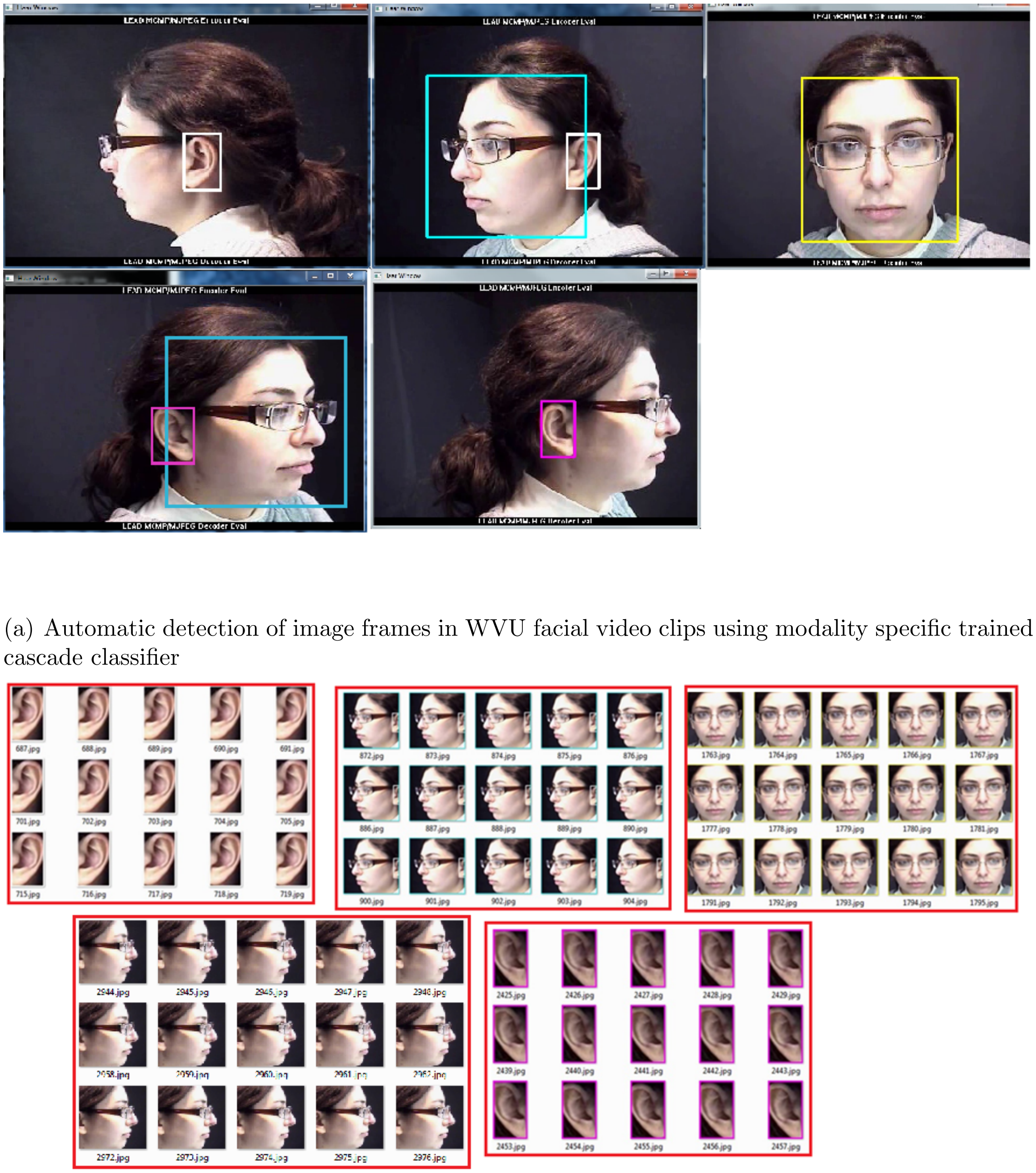
Table 4.1: Detection Accuracy for Unconstrained Video Clips

Modality	Detection Accuracy (%)
Frontal Face	97.55
Left Profile Face	93.42
Right Profile Face	92.21
Left Ear	98.77
Right Ear	98.84

Jones [8]. The algorithm is trained to detect frontal and profile faces in the video frames, respectively, using manually cropped frontal face images from color FERET [115] database, and profile face images from the University of Notre Dame Collection J2 database. Moreover, it is trained using cropped ear images from UND [116] color ear database to detect ear images in the video frames. By using these modality specific trained detectors, we can detect faces and ears in the video frames. The modality specific trained detectors are applied to the entire video sequence to detect the face and the ear regions in the video frames. Examples of detection results from the constrained and unconstrained dataset are shown in Figure 4.2 and Figure 4.3.

The results of the modality specific detection for the constrained face video clip is accurate. However, due to the uncontrolled head movements and non-planar rotation present in the unconstrained dataset, the detection results are not as accurate and there are few false positives. Table 5.1 shows the detection accuracies for the unconstrained dataset.

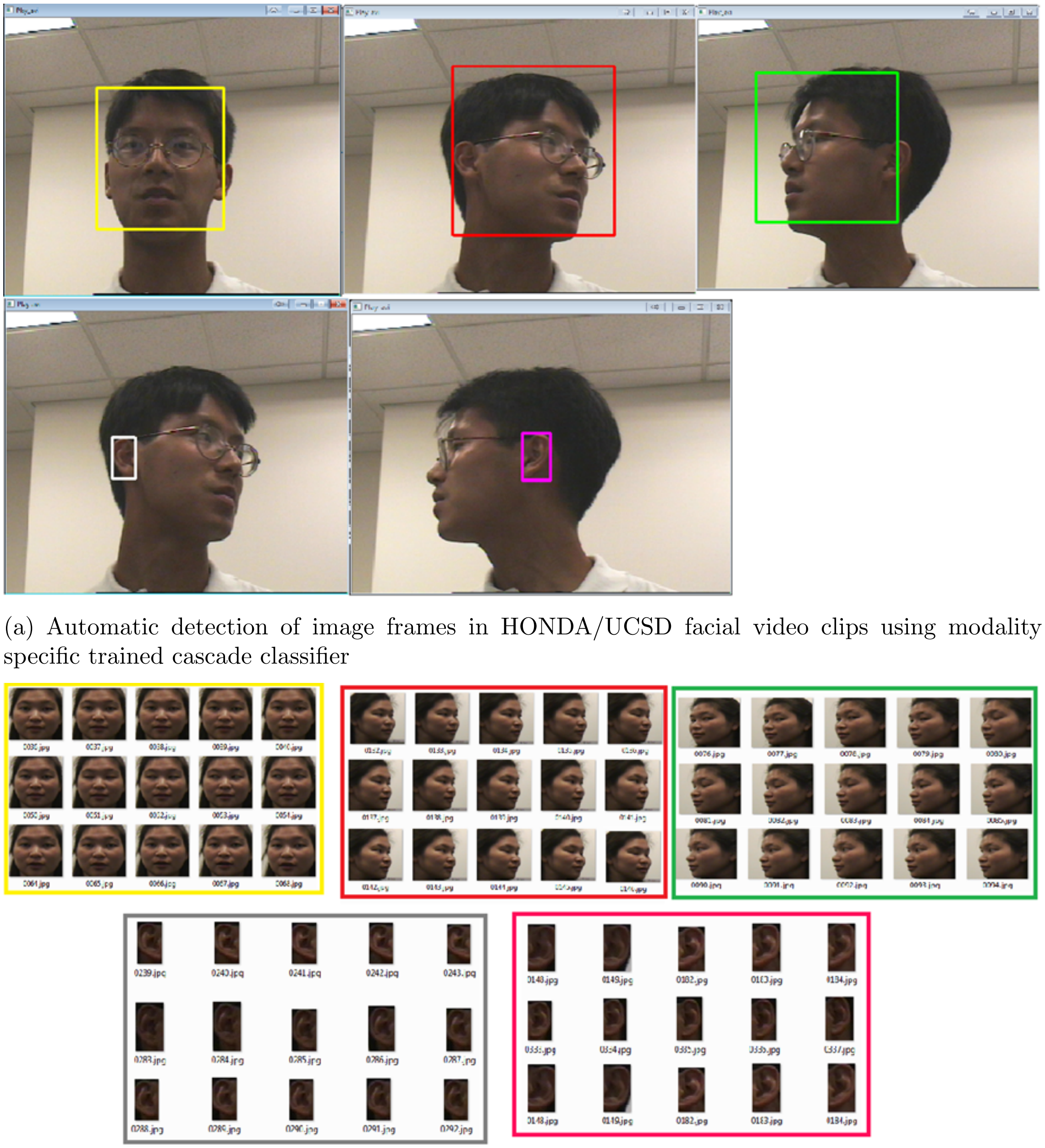
Before using the detected modality specific regions from the video frames for extracting features, some preprocessing steps are performed. The facial video clips recorded in the unconstrained environment contain variations in illumination and low contrast. Histogram equalization is performed to enhance the contrast of the images.



(a) Automatic detection of image frames in WVU facial video clips using modality specific trained cascade classifier

(b) Categorized detected regions from WVU facial video clips into modality specific groups from a video sequence

Figure 4.2: Modality Specific Image Frame Detection for Constrained Face Video Clips



(a) Automatic detection of image frames in HONDA/UCSD facial video clips using modality specific trained cascade classifier

(b) Categorized detected regions from HONDA/UCSD facial video clips into modality specific groups from a video sequence

Figure 4.3: Modality Specific Image Frame Detection for Unconstrained Face Video Clips

Finally, all detected modality specific regions from the facial video clips were resized; ear images were resized to 110 X 70 pixels and faces images (frontal and profile) were resized to 128 X 128 pixels.

4.2 Automatic Feature Learning Using Deep Neural Network

Even though the modalitiy specific sparse classifiers result in relatively significant recognition accuracy on the constrained face video clips, the accuracy suffers in case of unconstrained video because the classifier is vulnerable to the bias in the number of training images from different subjects. For example, subjects in the HONDA/UCSD dataset [114] randomly change their head pose. This results in a nonuniform number of detected modality specific video frames across different video clips, which is not ideal to perform classification through sparse representation.

In the subsequent sections we first describe the gabor feature extraction technique. Then, we describe the supervised denoising sparse autoencoders, which we use to automatically learn equal number of feature vectors for each subject from the uneven number of modality specific detected regions.

4.2.1 Feature Extraction

2D Gabor filters [117] are used in broad range of applications [118, 119] to extract scale and rotation invariant feature vectors. In our feature extraction step, uniform down-sampled Gabor wavelets are computed for the detected regions using equation 5.18, as proposed in [120]:

$$\psi_{\mu,\nu}(z) = \frac{||k_{\mu,\nu}||^2}{s^2} e^{(\frac{-||k_{\mu,\nu}||^2||z||^2}{2s^2})}[e^{ik_{\mu,\nu}z} - e^{\frac{-s^2}{2}}], \tag{4.1}$$

where $z = (x, y)$ represents each pixel in the 2D image, $k_{\mu,\nu}$ is the wave vector, which can be defined as $k_{\mu,\nu} = k_\nu e^{i\phi_\mu}$, $k_\nu = \frac{k_{max}}{f^\nu}$, k_{max} is the maximum frequency, and f is the spacing factor between kernels in the frequency domain, $\phi_\mu = \frac{\pi\mu}{2}$, and the value of s determines the ratio of the Gaussian window width to wavelength. Using equation 5.18, Gabor kernels can be generated from one filter using different scaling and rotation factors. In this paper, we used five scales, $\nu \in 0, ..., 4$ and eight orientations $\mu \in 0, ..., 7$. The other parameter values used are $s = 2\pi$,$k_{max} = \frac{\pi}{2}$, and $f = \sqrt{2}$ as considered for other studies in same application.

Before computing the Gabor features, all detected ear regions are resized to the average size of all the ear images, *i.e.*, 110 × 70 pixels, and all face images (frontal and profile) are resized to the average size of all the face images, i.e., 128 × 128 pixels. Gabor features are computed by convolving each Gabor wavelet with the detected 2D region, as follows:

$$C_{\mu,\nu}(z) = T(z) * \psi_{\mu,\nu}(z), \tag{4.2}$$

where $T(z)$ is the detected 2D region, and $z = (x, y)$ represents the pixel location. The feature vector is constructed out of $C_{\mu,\nu}$ by concatenating its rows.

4.2.2 Classical Sparse Autoencoder

Deep Learning is a class of machine learning techniques, where multiple layers of information processing stages in hierarchical architectures are utilized for pattern analysis/classification. There are different Deep Learning architectures available in the literature. The available Deep Learning architectures can be categorized broadly into three major classes: Convolution Neural Network (CNN), Recurrent Neural Network (RNN), and Deep auto-encoder (DNN). CNNs are Neural network with local

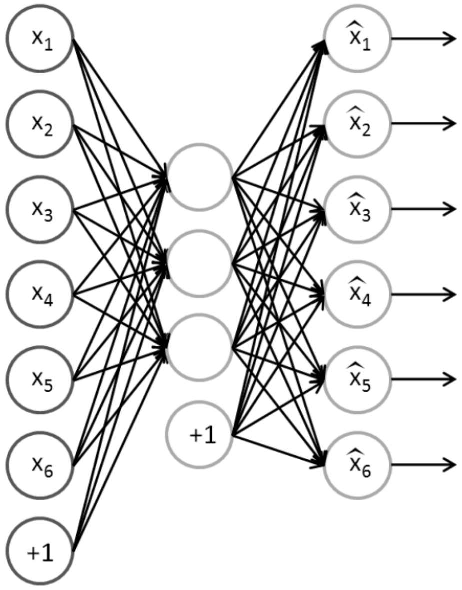


Figure 4.4: Structure of an autoencoder

and global connectivity structure consist of multiple stages of feature extractors. Convolutional neural networks are used in various image/scene recognition, video content analysis, Natural Language Processing applications etc. Recurrent Neural Network contains feed-back connection, so the activations can flow round in a loop. That enables the networks to do temporal processing and learn sequences, e.g., perform sequence recognition/reproduction or temporal association/prediction. Recurrent Neural Networks are used in speech recognition, video captioning, word prediction, translation applications etc. Thus we can see none of the Convolution Neural Network (CNN) or Recurrent Neural Network (RNN) architectures are suitable for the automatic feature extraction. However, in the Deep auto-encoder architecture the output target is the data input itself, often pre-trained with Deep belief network or using distorted training data to regularize the learning. In this subsection we describe the sparse autoencoder [121] learning algorithm, which is one approach to automatically learn features from unlabeled data.

The application of neural networks to supervised learning [122] is well proven in different applications including computer vision and speech recognition. An autoencoder neural network is an unsupervised learning algorithm, one of the commonly used building blocks in deep neural networks, that applies backpropagation to set the target values to be equal to the inputs. The reconstruction error between the input and the output of the network is used to adjust the weights of each layer. As shown in Figure 4.4, an autoencoder tries to learn a function $x_i = \hat{x}_i$, where x_i belongs to unlabeled training examples set $\{x_{(1)}, x_{(2)}, x_{(3)}, ..., x_{(n)}\}$, and $x_i \in \mathbb{R}^n$. In other words, it is trying to learn an approximation to the identity function, to produce an output $\hat{x}$ that is similar to x, in two subsequent stages: (i) An encoder that maps the input x to the hidden nodes through some deterministic mapping function $f : h = f(x)$, then (ii) A decoder that maps the hidden nodes back to the original input space through another deterministic mapping function $g : \hat{x} = g(h)$. For real-valued input, by minimizing the reconstruction error $||x - g(f(x))||_2^2$, the parameters of encoder and decoder can be learned. This simple autoencoder often resembles learning a low-dimensional representation similar to PCA [98]. However, it has been proven in [123] that such a nonlinear auto-encoder is different from PCA, also training an autoencoder results in minimizing the reconstruction error and maximizing a lower bound on the mutual information between the input and the learned representation.

In Figure 4.4, the number of hidden units can be increased, *i.e.*, the number of hidden nodes can be made even greater than the number of input nodes. In this case, we can learn some inherent structure of the data by imposing a sparsity constraint on the network. In other words, if we think of a neuron as being "active" if its output value is close to 1, or as being "inactive" if its output value is close to 0, we would like to constrain the neurons to be inactive most of the time. Recent research progress in biology reveals that the percentage of the activated neurons of human brain at a specific time is around 1% to 4% [124]. Therefore the sparsity constraint on the activation of the hidden layer is commonly used in the autoencoder based neural networks. Results show that sparse autoencoder often achieves better performance than that trained without the sparsity constraint [121].

4.2.3 Denoising Auto-encoder

Denoising Auto-encoder (DAE) [125], is a more generalized and robust version of the classical autoencoder. Since it assumes that the input data contain noise, it is suitable for learning features from noisy data. In other words, DAE is trained to reconstruct a clean or repaired version of the input from a corrupted or noisy one. It is proven that compared to ordinary autoencoders, denoising autoencoders are able to learn Gabor-like edge detectors from natural image patches.

In [125], DAE is designed and effectively tested to address different real world scenario where noise can corrupt the input data. The original input data $x \in \mathbb{R}^n$ can be affected by; a) Additive isotropic Gaussian noise ($\widetilde{x}|x \sim \mathcal{N}(x, \sigma^2 I)$, b) Masking noise, *i.e.*, a fraction of randomly chosen x is forced to 0, and c) Salt-and-pepper noise, *i.e.*, a fraction of randomly chosen x is forced to 0 or 1. The corruped data $\widetilde{x}$ is used as the input of the encoder, *i.e.*, the encoding of DAE is obtained by a nonlinear transformation function:

$$h = f_e(\widetilde{x}) = f_e(W\widetilde{x} + b_e) \tag{4.3}$$

where $h \in \mathbb{R}^y$ denotes the output of the hidden layer and can also be called feature representation or code, y is the number of units in the hidden layer, $W \in \mathbb{R}^{y \times n}$ is the input-to-hidden weights, b_e denotes the bias, $W\widetilde{x} + b_e$ stands for the input of the hidden layer, and f_e is the activation function of the hidden layer. The decoding or reconstruction of DAE is obtained by using a mapping function g_d:

$$\hat{x} = g_d(h) = g_d(W'h + b_d) \tag{4.4}$$

where $\hat{x} \in \mathbb{R}^z$ is the output of DAE, which is also the robust reconstruction of original corrupted data $\widetilde{x}$. The output layer has the same number of nodes as the input layer. $W' = W^T$ is referred to as tied weights. DAE aims to train the network by requiring the output data $\hat{x}$ to reconstruct the noisy input data $\widetilde{x}$, which is also called reconstruction-oriented training. Therefore, the reconstruction error should be used as the objective function or cost function as follows:

$$\underset{W,W',b_e,b_d}{min} \sum_{x \epsilon X} L(x, \hat{x}); \tag{4.5}$$

where L is the reconstruction error, typically squared error $L(x, \hat{x}) = ||x - \hat{x}||^2$ for real-valued inputs, cross-entropy function is used when the values of input x range from 0 to 1. Quantitative experiments show that even when the fraction of corrupted pixels ,*e.g.*, as corrupted by zero masking noises, reaches 55%, the recognition accuracy is still better or comparable with that of a network trained without corruptions.

4.2.4 Supervised Stacked Denoising Auto-encoder

To learn features from modality specific image regions which are robust to illumination, viewing angle, pose etc., we adopted the supervised autoencoder [61]. The supervised autoencoder is trained with the gallery image features (normal illumination

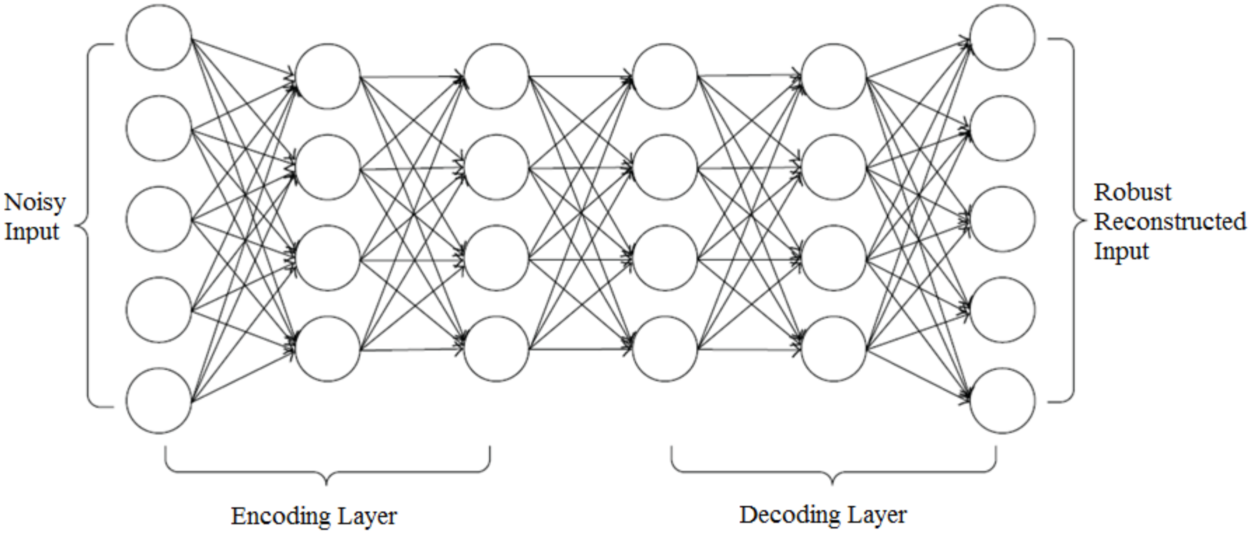


Figure 4.5: Stacked Denoising Auto-encoder network; stacked by two layers of Denoising Auto-encoders

and direct pose) represented as x_i, and the labelled probe image features (containing varying illumination, viewing angle and pose) represented as $\hat{x}_i$. By minimizing the objective criterion given in equation 4.6, *s.t.*, modality-specific features corresponding to the same person to be similar, the supervised autoencoders learn to capture the modality specific robust representation.

$$\underset{W,b_e,b_d}{min} \frac{1}{N} \sum_i \left(\|(x_i - g(f(\hat{x}_i))\|_2^2 + \lambda \|(f(x_i) - f(\hat{x}_i)\|_2^2 \right); \tag{4.6}$$

where $h = f(x) = tanh(Wx + b_e), g(h) = tanh(W^T h + b_d)$, N is the total number of training samples, and λ is the weight preservation term. The first term in equation 4.6 minimize the the reconstruction error, *i.e.*, after passing through the encoder and the decoder the variances of the unconstrained image regions will be repaired. The second term in equation 4.6 enforce simillarity preservation, leads to learn similar modality specific features corresponding to the same person.

Stacking denoising autoencoders to initialize a deep network follows the procedure of stacking Restricted Boltzman Machines (RBMs) in deep belief networks [126–128]. It is worth noting that the corrupted/ noisy input is only used for the initial denoising-training of each individual layer so that it may learn useful feature extractors. After training a first level denoising autoencoder, the learned encoding function f_{e1} can be used on clean input for reconstruction. The resulting representation is used to train a second level denoising autoencoder to learn a second level encoding function f_{e2}. This procedure can be repeated to stack the trained Denoising Auto-encoder layer by layer to form a Stacked Denoising Auto-encoder (SDAE). Figure 4.5 shows a typical instance of SDAE structure, which includes two encoding layers and two decoding layers. In the encoding part, the output of the first encoding layer acts as the input data of the second encoding layer.

After training a stack of encoders as explained in the previous section, its highest level output representation can be used as input to a stand-alone supervised learning algorithm. A logistic regression (LR) layer was added on top of the encoders as the final output layer [129], which enable the deep neural network to perform supervised learning. By performing gradient descent on a supervised cost function, the SDAE automatically learned fine-tuned network weights. Thus, the parameters of the entire SDAE network are fine-tuned to minimize the error in predicting the supervised target (*e.g.*, class labels). It is worth noting that SDAE is unsupervised while LR is supervised and only the data with labeled information can be used in LR stage. The Supervised Stacked Denoising Auto-encoder network is illustrated in Figure 4.6,

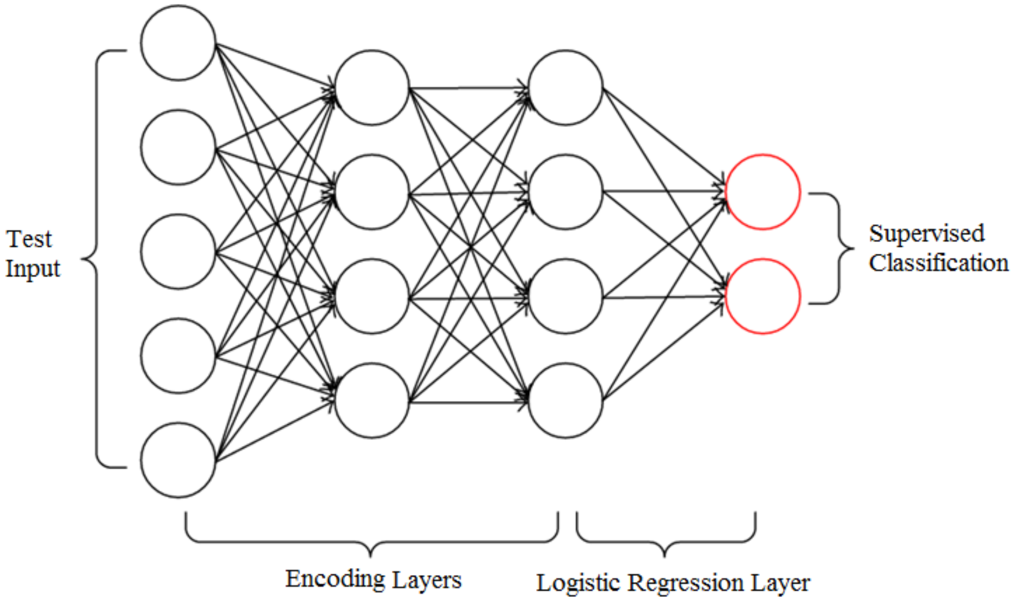


Figure 4.6: Supervised Stacked Denoising Auto-encoder

which shows a two-category classification problem. As per [129], we can see that the decoding part of SDAE is removed and the encoding part of SDAE is retained to produce the initial features. In addition, the output layer of the entire network (LR layer), is added.

4.2.5 Training the Deep Learning Network

In this subsection we will describe the constraints we faced while training the SDAE using the Layer-wise Greedy learning algorithm, and application of the supervised fine tuning to minimize the error of predicting the supervised target.

Empirically, deep networks were generally found to be not better, and often worse, than neural networks with more than one or two hidden layers [125]. A reasonable explanation is that gradient-based optimization often get stuck near poor solutions.

An approach that has been explored and proved successful to train deep networks with more than two hidden layers, is based on constructively adding layers [130], using a supervised criterion at each stage. However, it requires having an extensive training dataset to achieve generalization and avoid overfitting. In our application, the technique of constructively adding layers did not perform well because of the relatively small number of training samples. Moreover, we need to initialize the weights in a region near a good local minima, to better generalize the internal representations of the data.

Thus, we adopt the two stage training of the Deep Learning Network, where we have a better initialization to begin with and a fine tuned network weights that lead us to a more accurate high-level representation of the dataset. The steps of two stage Deep Learning Network training are as follows:

*Step*1. Stacked Denoising Autoencoders are used to train the initial network weights one layer at a time in a greedy fashion using Deep Belief Network (DBN).

*Step*2. The weights of the Deep learning network are initialized using the learned parameters from DBN.

*Step*3. Labelled training data are used as input, and their predicted classification labels obtained using the Logistic regression [125] layer along with the initial weights of the network used as an objective function to fine tune the entire network .

*Step*4. Back propagation is applied on the network to optimize the objective function (given in equation 4.5) , results in fine tune the weights and bias for the entire network.

*Step*5. Finally, the learned network weights and bias are used to extract image features to train the sparse classifier.

In the next two subsections the detailed explanation of the two stage training of the Deep Learning Network is provided.

4.2.5.1 Layer-wise Greedy Learning

Deep multi-layer artificial neural networks have multiple levels of non-linearities associated with them for effectively represent the highly non-linear and highly-varying functions in a compact higher level representation. However, until recently it was not obvious how to efficiently train such deep networks since gradient-based optimization starting from random initialization usually get stuck in local optima resulting in poor solutions. Hinton et al. [131], recently introduced a greedy layer-wise unsupervised learning algorithm for Deep Belief Networks (DBN), a generative model with many layers of hidden causal variables. Later on, in [129], a variant of the greedy layer-wise unsupervised learning is proposed to extend it to cases where the inputs are continuous.

In a DBN, let x be the input, and g_i be the hidden variables at layer i, then the computation of probability and sampling can be represented by the joint distribution:

$$P(x, g^1, g^2, ..., g^l) = P(x|g^1)P(g^1|g^2)...\alpha P(g^l - 2|g^l - 1)P(g^l - 1, g^l); \tag{4.7}$$

where all the conditional layers $P(g^i|g^{i+1})$ are factorized conditional distributions. In Hinton et al. [131] the hidden layer g^i is used as a binary random vector with n^i elements of g^i_j

$$P(g^i|g^{i+1}) = \prod_{j=1}^{n^i} P(g^i_j|g^{i+1}); \tag{4.8}$$

where

$$P(g_j^i = 1|g^{i+1}) = sigm(b_j^i + \sum_{k=1}^{n^{i+1}} W_{kj}^i g_k^{i+1}); \tag{4.9}$$

where $sigm(t) = \frac{1}{1+e^{-t}}$, the b_j^i are biases for unit j of layer i, and W^i is the weight matrix for layer i. If we set $g^0 = x$, the generative model for the first layer $P(x|g^1)$ will follow the equation 4.7.

A Deep Belief Network (DBN) can be used for generatively pre-training a Deep Neural Network (DNN) by using the learned weights as the initial weights [131]. A DBN can be efficiently trained in an unsupervised, layer-by-layer manner where the layers are typically made of restricted Boltzmann machines (RBM) [132]. A RBM is a generative stochastic artificial neural network that can learn a probability distribution over the set of inputs.

It should be noted that 1-level DBN is equivalent to an RBM. The greedy layer-wise strategy to add multiple layers in the DBN follows this same methodology. Train the first layer as an RBM that models the raw input $x = g^0$ as its visible layer. Then use the first layer to obtain the mean activations $P(g^1 = 1|g^0)$ of the input, which will be used as input data for the second layer. Train the second layer as an RBM $P(g^0, g^1)$, taking the transformed data (mean activations) as input to the visible layer of that RBM. Iterate the same steps to add the $(l+1)$th level, after training the top-level RBM of a l level DBN, such that, the distribution $P(g^{l-1}, g^l)$ from the RBM associated with layers $(l-1)$ and l is kept as part of the DBN generative model. In training a single RBM, weight updates are performed with gradient ascent via the following equation:

$$\Delta w_{ij}(t+1) = w_{ij}(t) + \eta \frac{\delta log(p(v))}{\delta w_{ij}}; \tag{4.10}$$

where η is the learning rate and $p(v)$ is the probability of a visible vector, which is given by:

$$p(v) = \frac{1}{Z} \sum_{h} e^{-E(v,h)}; \tag{4.11}$$

In equation 4.11, Z is the partition function (used for normalizing) and $E(v,h) = -h'Wv - b'v - c'h$ is the energy function assigned to the state of the network. Here, v stands for visible units and hidden layer activations h stands for hidden units. Computation of stepwise weight updates is explained in Algorithm 1, where b is the vector of biases for visible units and c is the vector of biases for the hidden units.

Algorithm 2 Stepwise weight update of the DBN

```
1. Initialize the visible units to a training vector.
2. Update the hidden units in parallel given the visible units: $p(h_j = 1|V) = sigm(b^j + \sum_i v_i W_{ij})$
3. Update the visible units in parallel given the hidden units: $p(v_i = 1|H) = sigm(c^i + \sum_j h_j W_{ij})$ ("Reconstriction" step.)
4. Reupdate the hidden units in parallel given the reconstructed visible units following the same equation as step 2.
5. Perform weight update by following: $\Delta w_{ij} \propto \langle v_j h_j \rangle_{data} - \langle v_j h_j \rangle_{reconstriction}$
```

4.2.5.2 Supervise Fine Tuning

Once all layers are pre-trained, the network goes through the second stage of training called fine-tuning. This supervised fine-tuning is performed to minimize the overall prediction error of the entire Deep Learning Network. To achieve this, a logistic regression layer (or in generic scenario a soft-max regression classifier) is added on top of the network. We then train the entire network as we would train a multilayer perceptron, where the encoding parts of each auto-encoder are used. This stage is supervised since now we use the target class during training.

The network is illustrated in Figure 4.6, which shows a two-category classification problem (there are two output values). We can see that the decoding part of SDAE is removed and the encoding part of SDAE is retained to produce the initial features. In addition, the output layer of the whole network, which is also called logistic regression layer, is added. The following sigmoid function is used as activation function of the logistic regression layer:

$$h(x) = \frac{1}{e^{-Wx-b}}; \tag{4.12}$$

where x is the output of the last encoding layer y^l, in other words the deep features that are pretrained by SDAE network. The output of the sigmoid function is between 0 and 1, which denotes the classification results in case of two class classification problem. Therefore, we can use the errors between the predicted classification results and the true labels associated with the training data points to fine-tune the whole network weights.The cost function is defined as the following cross-entropy function:

$$Cost = -\frac{1}{m}\left[\sum_{i=1}^{m} l^{(i)} log(h(x^{(i)})) + (1-l^{(i)}) log(1-h(x^{(i)}))\right]; \tag{4.13}$$

where $l^{(i)}$ denotes the label of the sample $x^{(i)}$. By minimizing the cost function, we update the network weights.

4.3 Modality Specific and Multimodal Recognition

The modality specific sub-dictionaries are formed utilizing the feature vectors generated by Deep Learning Network using the modality specific detected regions of the training video sequence. Where, *modality specific sub-dictionary* is the learned dictionary using the modality specific training data of an individual subject.

First the modality specific sub-dictionary d, and the coefficient matrix X_0 are obtained by linearly representing the training data Y; *i.e.*, $Y = dX_0$. Then we concatenate the modality specific learned sub-dictionaries d_j^i to build the modality specific dictionary D_i, shown in equation 4.14.

$$D_i = [d_1^i; d_2^i; ...; d_j^i]; \tag{4.14}$$

where, i represents the modality used in the multimodal recognition, $i \in 1, 2, ..., 5$; and j stands for the total number of training video sequences. Where, *modality specific dictionary* is the learned dictionary using the modality specific (such as left ear, left profile face, frontal face, right profile face, and right ear) training data of all the subjects in the entire database.

4.3.1 Sparse Representation For Classification

For each training video sequence, the modality specific sub-dictionaries $d_l^i \in R^p$, are formed using the feature vectors genertaed by Deep Learning Network utilizing the modality specific detected regions of the lth training video sequence, and p is the length of the feature vectors learned by Deep Learning Network. Similarly, the feature vector learned by Deep Learning Network, $y^i \in R^p$, using modality specific detected regions in the test video, is then represented as a linear combination of the feature vectors learned from the training video sequences:

$$y^i = d_1^i * \alpha_1^i + d_2^i * \alpha_2^i + ... + dj^i * \alpha_j^i. \tag{4.15}$$

where α_j^i's are the coefficients corresponding to the training data of ith modality in the jth training video sequence. Equation 4.15 can be represented by using the concatenated modality specific dictionary D_i, defined in equation 4.14, as:

$$y^i = D_i x \in R^p, \tag{4.16}$$

where x is the coefficient vector, and the test data y^i belongs to ith modality. In our approach we used Smoothed l^0 (SL0) [133] norm to solve equation 4.16. SL0 algorithm is utilized to obtain the sparsest solution of under determined systems of linear equations by directly minimizing the l^0 norm. SL0 has proven to be more efficient than l^0 and l^1 in space and time complexity [133].

Using majority voting on the sparse classification coefficients obtained from the individual sub-dictionaries for all the modality specific regions detected from a specific test video, the modality specific classification decisions are made. Later, the final classification of the subject present in the video sequence is made based upon the score level fusion of the modality specific classification. Some of the modalities may not be available in the video used for recognition, in these cases we make the final decision based on the available modalities. In the experiments section, we tested the algorithm when all the modalities are available during the recognition phase and also all possible combinations of missing modalities, i.e., 1, 2 or 3 modalities are absent.

4.3.2 Multimodal Recognition

The five modalities- left ear, left profile face, frontal face, right profile face, and right ear- are combined at the score level. While performing fusion at the score level, we have the flexibility to fuse the match scores from various modalities upon their availability. The *tanh* [134] was used to transform the matching scores obtained from the different matchers into a common domain. Later, the weighted sum technique was used to fuse the results at the score level. The *tanh*-estimators score normalization technique, used to make the match score from individual matchers comparable, is defined as follows:

$$s_j^n = \frac{1}{2}\left\{tanh(0.01(\frac{s_j - \mu_{GH}}{\sigma_{GH}})) + 1\right\}, \tag{4.17}$$

where s_j and s_j^n are the match scores before and after normalization, respectively. μ_{GH} and σ_{GH} are the mean and standard deviation estimates of the genuine score distribution given by Hampel estimators [135], respectively. Hampel's estimators are based on the influence functions ψ which are odd functions and can be defined for any x as follows:

$$\psi(x) = \begin{cases} x, & 0 \leq |x| < a, \\ a\ sign(x), & a \leq |x| \leq b, \\ \frac{a(r-|x|)}{r-b}\ sign(x), & b \leq |x| \leq r, \\ 0, & r \leq x, \end{cases} \tag{4.18}$$

where

$$sign(x) = \begin{cases} +1, & \text{if } x \geq 0, \\ -1, & \text{otherwise}, \end{cases} \tag{4.19}$$

In equation 5.21, the value of a, b, and r in ψ, reduces the influence of the scores at the tails of the distribution during the estimation of the location and scale parameters. The normalized match scores are then fused using the weighted sum technique:

$$S_p = \sum_{i=1}^{M} w_i * s_i^n; \tag{4.20}$$

where w_i and s_i^n are the weight and normalized match score of the i^{th} modality specific classifier, respectively, such that $\sum_{i=1}^{M} w_i = 1$. In this study, the weights $w_i, i = 1, 2, 3, 4, 5$; stands for the left ear, left profile face, frontal face, right profile face, and right ear modalities, respectively. These weights can be obtained by exhaustive search or based on the individual performance of the classifiers [134]. In our work, we empirically chose the weights for modality specific classifiers to maximize the fused multimodal recognition accuracy.

4.4 Experimental Results

In this section we describe the constrained, WVU dataset [113] and the unconstrained, HONDA/UCSD [114] dataset contents. Then, we demonstrate the results of the set of modality specific and multi-modal recognition experiments on both datasets.

4.4.1 WVU Dataset

The WVU data set [113] contains video sequences obtained by a camera moving in a semicircle around the face, starting from the extreme left profile of each subject (0 degree) up-to the extreme right profile (180 degree), for a total of 402 subjects. Video clips in the WVU database are collected at different times under the same environmental constraints, *e.g.*, illumination and distance from the camera. Three of the subjects had their left and right ears fully occluded, and therefore, they were removed from the dataset. Fifty nine subjects have two or more video sequences with widely varied appearance with and without facial hair, glasses, caps, and long hair, which partially occluded the ear, while the remaining 340 subjects only have one video sequence.

To perform the multimodal recognition, we trained the modality specific dictionaries for all the five modalities using the training video sequences. Despite the situation of missing modalities in the test video, we are able to perform the multimodal recognition using the available modalities because of obtaining the modality specific dictionaries using the training data. In order to evaluate our algorithm, we prepared two instances of datasets from the available video sequences in the WVU dataset.

4.4.1.1 Dataset-1

In dataset 1, we use one video sequence for each subject, which results into a total of 399 video sequences. The detected modality specific regions from the video sequence of each subject are separated for training and testing in a non-overlapping fashion. Detection of the left ear and the left profile face is performed between 0 to 30 degree rotation of the camera in the video. The detected regions in the first 100 frames were used for training and the detected regions in the next 100 frames were used for testing. Detection of frontal face is performed on frames between 75 to 105 degrees, where the detected regions in the first 100 frames were used for training and the detected regions in the next 100 frames for testing. Detection of right ear and right profile face performed between 150 to 180 degrees, where the detected regions in the last 100 frames were used for training and the preceding 100 frames for testing.

4.4.1.2 Dataset-2

In dataset 2, we use only those subjects who have more than one video sequence, which results into a total of 121 video clips, with one subject having three video sequences, one subject having four sequences, and the rest of the 57 subjects having two video sequences. The detected different modality specific regions from one video were used for training and from the others were used for testing in cross fold fashion. Detection of the left ear and the left profile face is performed between 0 to 30 degree where detected regions in the first 200 frames are used. Detection of the frontal face is performed between 75 to 105 degree where detected regions in the first 200 frames are used. Detection of the right ear and the right profile face is performed between 150 to 180 degree where detected regions in the last 200 frames are used.

To compute the multimodal rank-1 recognition result, score level fusion is performed using majority voting of rank-1 recognition rate from the five different modalities. The multimodal recognition accuracy of our approach is as follows: for dataset-1 we obtained 99.17% average rank-1 recognition rate, and for dataset-2, we obtained 96.49% average rank-1 recognition rate. The best rank-1 recognition rates, using ear, frontal and profile face modalities for multimodal recognition, compared with the results reported in [53–55] is shown in Table 5.3. All the modality specific recognition rates and the multimodal recognition rate of the proposed approach outperforms the other multimodal recognition techniques that uses ear, frontal face and profile face.

Table 4.2: Comparison of 2D multimodal (frontal face, profile face and ear) rank-1 recognition accuracy with the state-of-the-art techniques

Approaches	Modalities	Fusion Performed In	Best Reported Rank-1 accuracy
Kisku et al. [54]	Ear and Frontal Face	Decision Level	Ear: 93.53%; Frontal Face: 91.96%; Profile Face: NA; Fusion: 95.53%
Pan et al. [55]	Ear and Profile Face	Feature Level	Ear: 91.77%; Frontal Face: NA; Profile Face: 93.46%; Fusion: 96.84%
Boodoo et al. [53]	Ear and Frontal Face	Decision Level	Ear: 90.7%; Frontal Face: 94.7%; Profile Face: NA; Fusion: 96%
This Work	Ear , Frontal and Profile Face	Score Level	Ear: **95.04%**; Frontal Face: **97.52%**; Profile Face: **93.39%**; Fusion: **99.17%**

Later, we performed experiments when all the modalities were available during the training and only some of the modalities were available during the testing. The accuracy of the recognition results using all possible combinations of the different modalities in the test video, for dataset-1 and dataset-2 are shown in Table 5.4 and 4.4, respectively. The results indicate that, among all possible combinations of different modalities, frontal face with ear, *i.e.* right and left ear modalities, have the best recognition rate.

4.4.2 HONDA/UCSD Dataset

The publicly available HONDA/UCSD dataset [114], contains facial video clips with non-planar head rotations (left-right and up-down directions) as well as various facial expressions. The dataset has two parts, dataset-1 and dataset-2, that consist of separate training and testing facial video clips of 20 and 15 unique subjects, re-

Table 4.3: Recognition Result of Multimodal Recognition with all possible combinations of 2, 3 and 4 modalities using dataset-1(WVU). 'Fr_Fc' stands for frontal face,'Lf_pr' and 'Rt_pr' stands for left and right profile face respectively, and, 'Lf_ear' and 'Rt_ear' stands for left and right ear respectively.

Test Done with	Rank-1 accuracy	Test Done with	Rank-1 accuracy
Combining any two modalities			
Fr_Fc+ Lf_pr/ Rt_pr	97.52%	Fr_Fc+ Lf_ear/ Rt_ear	**98.35%**
Lf_ear/ Rt_ear + Lf_pr/ Rt_pr	**98.35%**		
Combining any three modalities			
Fr_Fc+ Lf_pr+ Lf_ear	97.52%	Fr_Fc+ Lf_pr+ Rt_ear	97.52%
Fr_Fc+ Rt_pr+Rt_ear	97.52%	Lf_pr+ Lf_ear+Rt_pr	94.21%
Fr_Fc+ Lf_pr+ Rt_pr	97.52%	Rt_pr+ Rt_ear+Lf_pr	95.04%
Fr_Fc+ Lf_ear+ Rt_ear	**98.35%**	Rt_pr+ Lf_ear+Rt_ear	**98.35%**
Fr_Fc+ Rt_pr+Lf_ear	97.52%	Lf_pr+ Lf_ear+Rt_ear	**98.35%**
Combining any four modalities			
Fr_Fc+ Rt_ear+ Lf_ear+ Lf_pr	**98.35%**	Fr_Fc+ Rt_ear+ Lf_ear+ Rt_pr	**98.35%**
Fr_Fc+ Lf_pr+ Rt_pr+Lf_ear	97.52%	Fr_Fc+ Lf_pr+ Rt_pr+Rt_ear	**98.35%**
Rt_ear+ Lf_ear+ Lf_pr+ Rt_pr	**98.35%**		

Table 4.4: Recognition Result of Multimodal Recognition with all possible combinations of 2, 3 and 4 modalities using dataset-2 (WVU). ‘Fr_Fc’ stands for frontal face,‘Lf_pr’ and ‘Rt_pr’ stands for left and right profile face respectively, and, ‘Lf_ear’ and ‘Rt_ear’ stands for left and right ear respectively.

Test Done with	Rank-1 accuracy	Test Done with	Rank-1 accuracy
Combining any two modalities			
Fr_Fc+ Lf_pr/ Rt_pr	91.23%	Fr_Fc+ Lf_ear/ Rt_ear	**94.74%**
Lf_ear/ Rt_ear+ Lf_pr/ Rt_pr	**94.74%**		
Combining any three modalities			
Fr_Fc+ Lf_pr+ Lf_ear	92.98%	Fr_Fc+ Lf_pr+ Rt_ear	91.23%
Fr_Fc+ Rt_pr+Rt_ear	91.23%	Lf_pr+ Lf_ear+Rt_pr	89.47%
Fr_Fc+ Lf_pr+ Rt_pr	91.23%	Rt_pr+ Rt_ear+Lf_pr	89.47%
Fr_Fc+ Lf_ear+ Rt_ear	**94.74%**	Rt_pr+ Lf_ear+Rt_ear	**94.74%**
Fr_Fc+ Rt_pr+Lf_ear	92.98%	Lf_pr+ Lf_ear+Rt_ear	**94.74%**
Combining any four modalities			
Fr_Fc+ Rt_ear+ Lf_ear+ Lf_pr	**94.74%**	Fr_Fc+ Rt_ear+ Lf_ear+ Rt_pr	**94.74%**
Fr_Fc+ Lf_pr+ Rt_pr+Lf_ear	92.98%	Fr_Fc+ Lf_pr+ Rt_pr+Rt_ear	91.23%
Rt_ear+ Lf_ear+ Lf_pr+ Rt_pr	**94.74%**		

spectively. HONDA/UCSD dataset contains a total of 89 facial video sequences of 35 unique subjects, where each subject has two or more video clips. In our experiments, we used one facial video sequence for training and rest for testing in cross fold approach.

The Adaboost detector results in a few false modality specific detection for the Left and Right Profile Face, when applied on the HONDA/UCSD unconstrained dataset. Thus, to quantify how the false detection affect the multimodal recognition accuracy we manually selcted only the true positive detections. In Table-4.5 and 4.6 the result of modality specific recognition on the HONDA/UCSD is given respectively for the datset containing false positive dectection and the dataset which include only the true positive detected regions. The multimodal recognition accuracy obtained including false positive dectection is 97.14% (34 true positive out of 35 subjects), and 100% using only true positive detected regions.

The feature vecorts automatically learned using the trained Deep Learning network resulted in legth of 9600 for frontal and profile face; 4160 for ear. In order to decrease the computational complexity and to find out most effective feature vector length to maximize the recognition accuracy, the dimensionality of the feature vector is reduced to a lower dimension using Principal Component Analysis (PCA) [47, 98]. Using PCA, the number of features is reduced to 500 and 1000. In Table- 4.5 and 4.6 the modality specific recognition accuracy obtained for the original feature vector and for the reduced feature vector of 1000, 500 is shown. Feature vectors of 1000 elements obtained the best multimodal recognition accuracy.

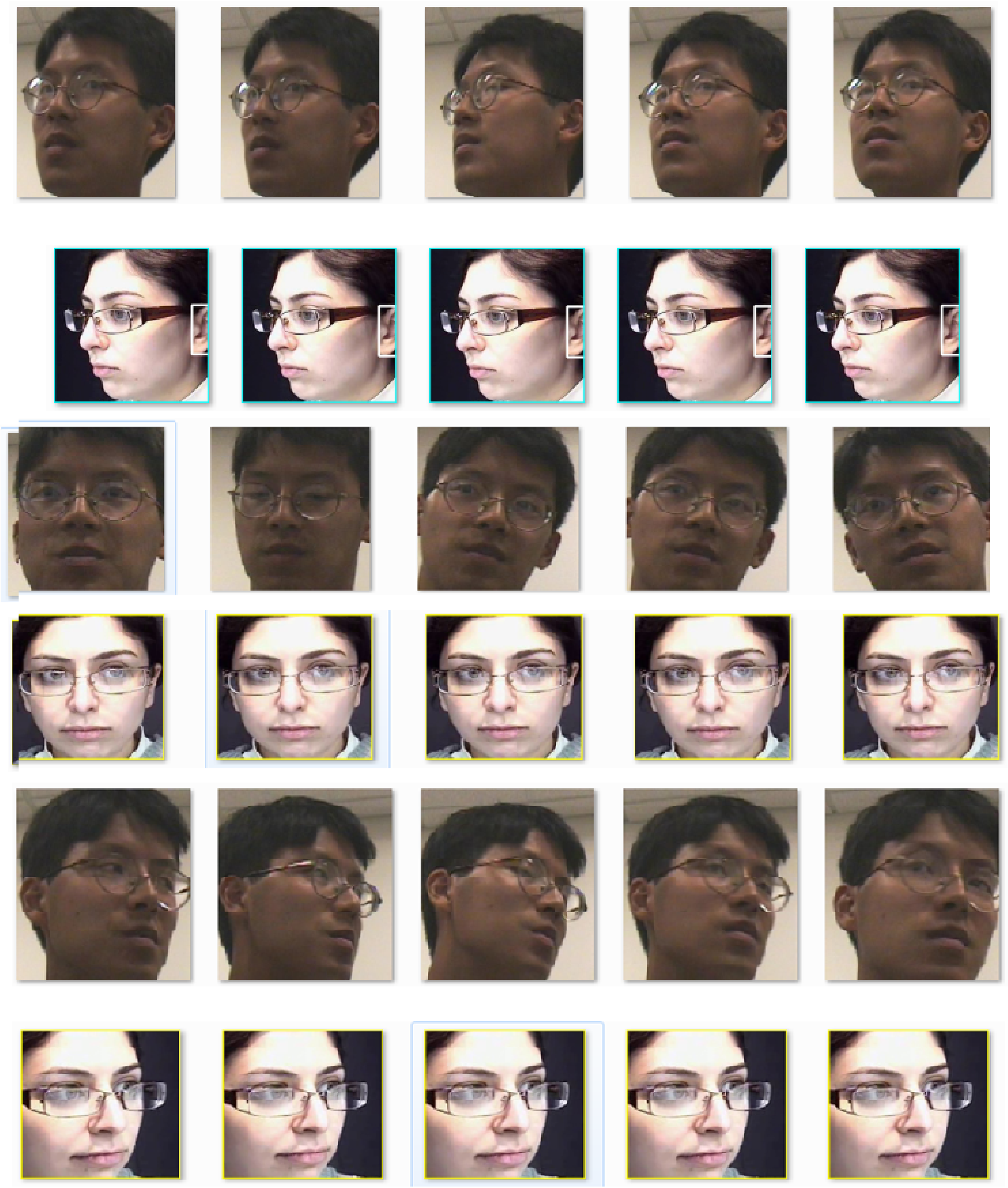


Figure 4.7: Nonplanar movement In HONDA dataset compared with WVU. Left Profile, Frontal and Right Profile

Table 4.5: Modality Specific Recognition Accuracy using all detected regions

Gabor Feature Length	Frontal face	Left profile face	Right profile face	Left ear	Right ear
No feature reduction	91.43%	71.43%	71.43%	85.71%	85.71%
1000	91.43%	71.43%	**74.29%**	**88.57%**	**88.57%**
500	88.57%	68.57%	68.57%	85.71%	82.86%

Table 4.6: Modality Specific Recognition Accuracy using only accurately detected regions

Gabor Feature Length	Frontal face	Left profile face	Right profile face	Left ear	Right ear
No feature reduction	91.43%	80.00%	82.86%	94.29%	91.43%
1000	**97.14%**	**82.86%**	82.86%	94.29%	**94.29%**
500	91.43%	81.19%	80.00%	91.43%	91.43%

4.4.3 Comparison with Baseline Algorithms

Due to the unavailability of proper datasets for multimodal recognition studies [9], often virtual multimodal databases are synthetically obtained by pairing a subject from different databases consists of different modalities. To the best of our knowledge, the proposed approach is the first study where multiple modalities are extracted from a single data source and belongs to the same subject. Thus, we compare the performance of the proposed techique of learning automatic robust features using Deep Learning network and using sparse respresentation for classification with the following Baseline Algorithms due to their close relationships. It is also worth noting that all the comparisons are based on the same training/test set.

1) Sparse Representation for Classification (SRC) [9] with extracted Gabor features.

2) KSVD [136] dictionary learning with extracted Gabor features. K-SVD is a dictionary learning algorithm for creating a dictionary for sparse representations by iteratively alternating between sparse coding the input data based on the current dictionary, and updating the atoms in the dictionary to better fit the data. Therefore, K-SVD is better suited than Sparse classifier to train models with varying number of training samples for different classes.

In Table-4.7 the comparisons of multimodal recogntion accuracy of the baseline techniques and the proposed method are provided for both WVU and HONDA/UCSD datasets. The comparison shows that the proposed technique perform better on both the constrained and unconstrained datasets compared with the other Baseline Algorithms. However, we can see that performance of the two Baseline Algorithms are relatively satisfactory while applying on the constrained (WVU) dataset, but while applying on the unconstrained (HONDA/UCSD) dataset the performance of the two Baseline Algorithms are very poor. The Sparse Classifier (SRC) is failing due to a biased number of training samples for different subjects. Although KSVD is performing better than SRC, due to the presence of nonplanar movements (shown in Figure 4.7) in the unconstrained (HONDA/UCSD) dataset it cannot achieve a satisfactory recognition accuracy.

Table 4.7: Comparison of Multimodal Recognition with the Baseline Algorithms

Method	WVU	Honda/UCSD
Gabor+SRC	95.04%	45.71%
Gabor+KSVD	97.52%	68.57%
(Gabor+Deep Learning)+SRC	**99.17%**	**97.14%**

4.4.4 Parameter Selection for the Deep Neural Network

We have tested the performance of the proposed multimodal recognition framework against different parameters of the Deep Neural Network. We varied the number of hidden layers from three to seven. By using five hidden layers we achieved the best performance. To incorporate the sparsity in the hidden layers, we also conducted experiments by changing the number of hidden nodes from two to five times of the input nodes. By using twice the hidden nodes of the input nodes in the five hidden layers we obtain the best accuracy of the multimodal recognition system. The pre-training learning rate of the DBN is used as 0.001 and the fine tuning learning rate of the SADE is used as 0.1 to achieve the optimal performance. While training the SADE network in a Core i7-2600K CPU clocked at 3.40GHz Windows® PC using Theano Library (Python Programming Language) pre-training of the DBN takes approximately 600 minutes and the fine-tuning of the SADE network converged within 48 epochs in 560.2 minutes.

CHAPTER 5

Multimodal Low Resolution Face and Frontal Gait Recognition from Surveillance Video

The importance of identifying and monitoring the activity of registered offenders using video surveillance footage has been proven effective on several occasions, *e.g.*, identifying the Boston bombing suspects, to lead the detectives in the right direction. However, the quality of the video data acquired by the surveillance system poses challenges. The primary causes of poor image quality recorded in most digital video surveillance systems are low resolution, excessive quantization, and low frame rate. Moreover, high-resolution video surveillance systems require excess storage space. These factors result in low-resolution biometric data, *e.g.*, face images, obtained from the video surveillance clips collected using the existing video surveillance systems.

In this chapter [137] we propose a solution for accurate human identification from low-resolution video surveillance footages by combining gait recognition and low resolution (LR) face recognition. The proposed system, shown in Figure 5.1, is a fully automatic platform which first extracts the frontal gait silhouettes and low resolution face images from the frontal walking video surveillance clips. Then obtains the feature vectors from the preprocessed frontal gait silhouettes, and the low resolution face

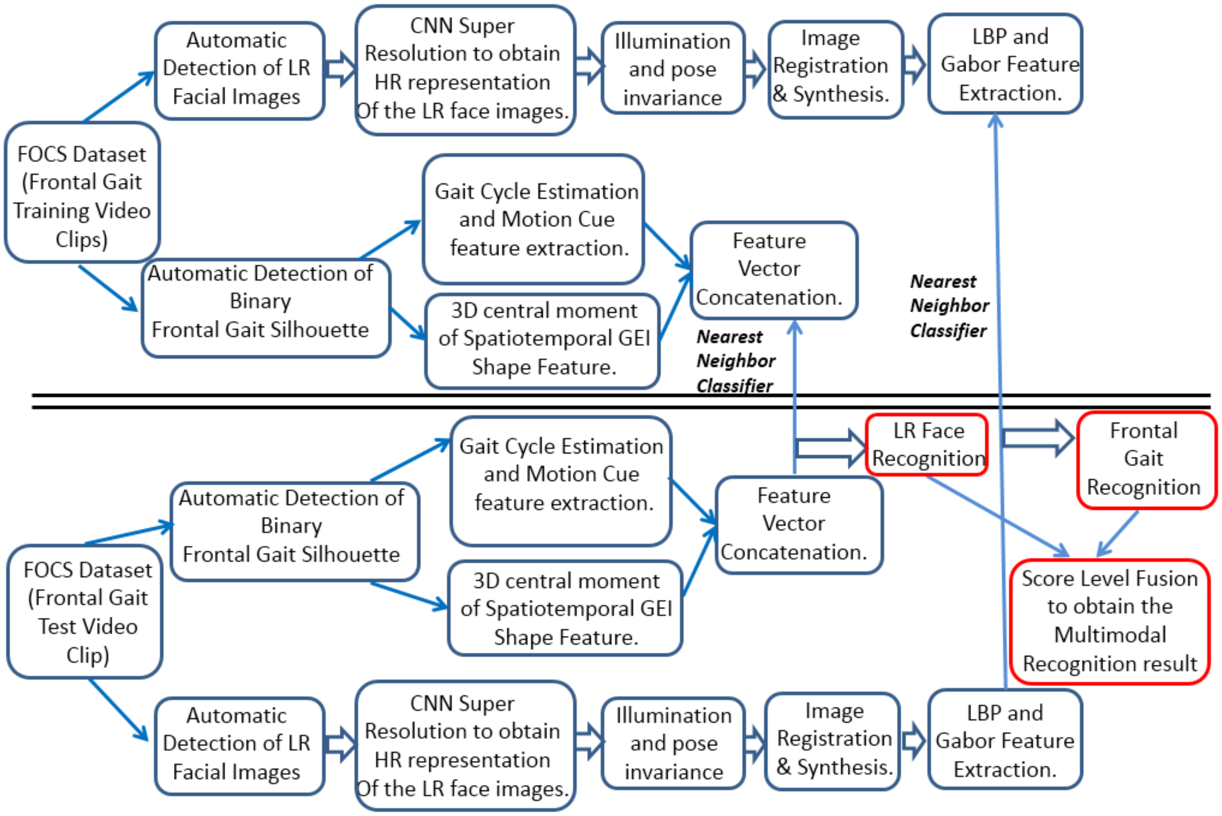

Automatic Detection of LR Facial Images
CNN Super Resolution to obtain HR representation Of the LR face images.
Illumination and pose invariance
Image Registration & Synthesis.
LBP and Gabor Feature Extraction.
FOCS Dataset (Frontal Gait Training Video Clips)
Gait Cycle Estimation and Motion Cue feature extraction.
Feature Vector Concatenation.
Automatic Detection of Binary Frontal Gait Silhouette
3D central moment of Spatiotemporal GEI Shape Feature.
Nearest Neighbor Classifier
Nearest Neighbor Classifier
LR Face Recognition
Frontal Gait Recognition
Automatic Detection of Binary Frontal Gait Silhouette
Gait Cycle Estimation and Motion Cue feature extraction.
Feature Vector Concatenation.
FOCS Dataset (Frontal Gait Test Video Clip)
3D central moment of Spatiotemporal GEI Shape Feature.
Score Level Fusion to obtain the Multimodal Recognition result
Automatic Detection of LR Facial Images
CNN Super Resolution to obtain HR representation Of the LR face images.
Illumination and pose invariance
Image Registration & Synthesis.
LBP and Gabor Feature Extraction.

Figure 5.1: System Block Diagram: Multimodal Biometrics Recognition from Video Survillience Data

images. Later the feature vectors are used to train two separate classifiers to perform the frontal gait recognition, and low resolution face recognition. Finally, the individual recognition results are fused through score level fusion. Given a test surveillance video clip of a subject walking towards the camera, first the gait features and LR face image features are extracted, later Nearest neighbor classifiers are used to separately obtain the Rank-1 frontal gait recognition and LR face recognition results. Finally, score level fusion is performed to fuse the individual recognition results.

The remainder of this chapter is organized as follows: Section 5.1 details the segmentation of the frontal Gait silhouettes from the background, and detection of the low resolution face images from the frontal walking video sequences. Section 5.2

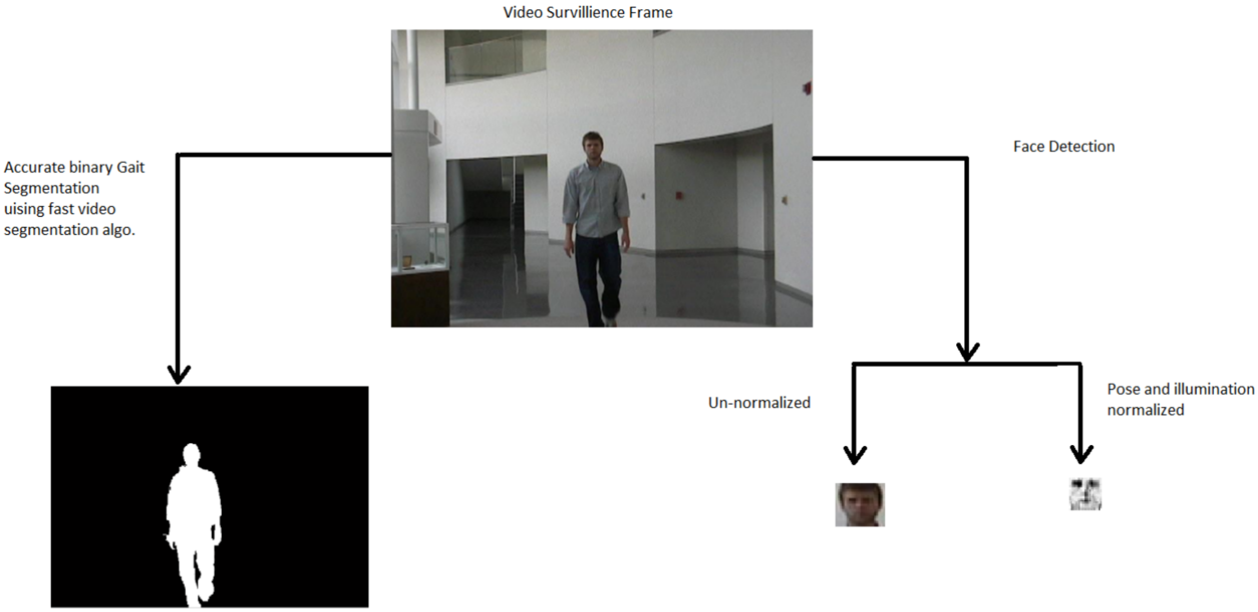

Video Survillience Frame
Accurate binary Gait Segmentation using fast video segmentation algo.
Face Detection
Un-normalized
Pose and illumination normalized

Figure 5.2: Gait and Low Resolution Face Extraction

describes the feature representations of the frontal Gait binary silhouettes. Section 5.3 presents the proposed low resolution face recognition technique. Section 5.4 provides the experimental setup and results on the frontal Gait data and LR face images obtained from the frontal walking video sequences to demonstrate the performance of the proposed framework.

5.1 Gait Silhouette and Low Resolution Face Extraction

To perform multimodal biometric recognition, we need to detect the face and the gait silhouette from the surveillance video clips. The surveillance video clips are captured by a static video camera which records the frontal view of a walking person. The subjects start walking form a distance directly approaching the camera. We extract both the frontal gait silhouettes from the sequence of video frames and the low resolution frontal face images, as explained below.

We adopted the fast object segmentation method proposed by Papazoglou *et.al.* [138] for segmenting the foreground silhouette from the background. The fast object segmentation is fast, fully automatic, and have minimal assumptions about the motion of the foreground. Therefore, it performs efficiently in cases of unconstrained settings, presence of rapidly moving objects, arbitrary object motion and appearance change, and non-rigid deformations and articulations. The fast object segmentation technique first produces a rough estimate of the pixels that are inside the object, based on motion boundaries using optical flow obtained from pairs of subsequent frames [139, 140]. In the second step, a spatiotemporal extension of GrabCut [141, 142] technique is used to bootstrap an appearance model based on the initial foreground estimate, and refine it by integrating information over the entire video sequence. An example of segmented silhouettes from different frames, using fast object segmentation [138], is shown in Fig. 5.3, which shows accurate segmentation by isolating the silhouette from its reflection on the shiny floor.

To automatically detect the low resolution frontal face images in the surveillance video clips, we adopt the Adaboost object detection technique, proposed by Viola and Jones [8]. The algorithm is trained to detect low resolution frontal faces using manually cropped frontal face images from the color FERET [115] database. By using the trained detector, we can detect low resolution faces in the video frames. The trained detector is applied to the entire video sequence to detect the LR frontal faces. An example of the detection results from a surveillance video clip is shown in Figure 5.2.

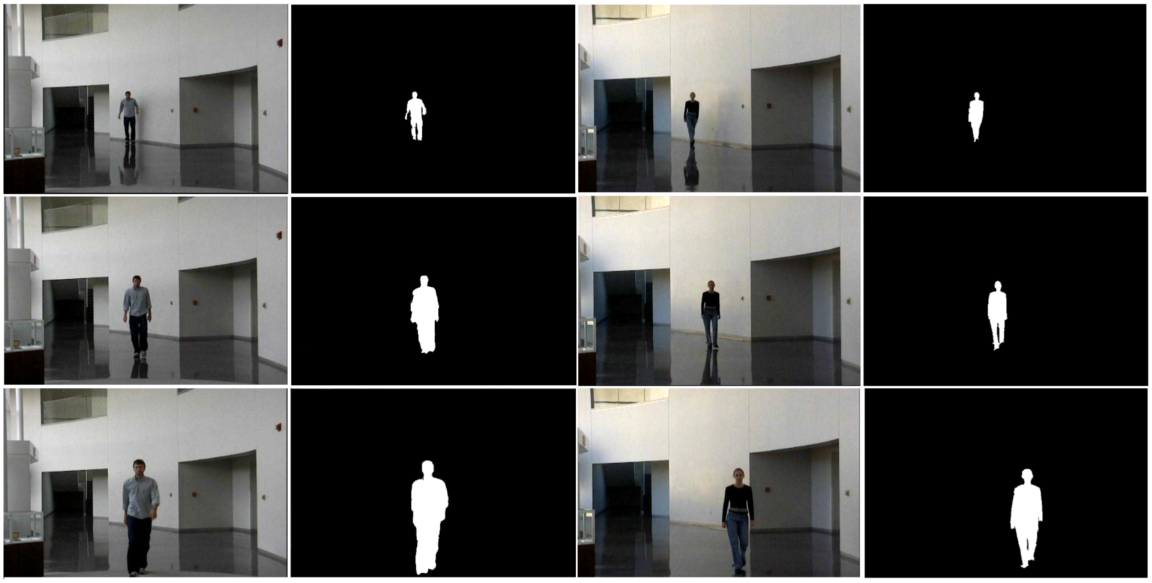


Figure 5.3: Frontal Gait Silhouette Segmentation

5.2 Gait Feature Representation

Existing studies in the literature [143,144] suggests that human periodic movement speeds and patterns are similar in repeated trials of the same subject. We have incorporated both Model-free and Model-based feature representation of the segmented silhouettes to obtain accurate and efficient gait recognition. Identification of the gait cycle, using the frontal gait video, is proposed to compute average movement speed for efficient model-free gait recognition. Moreover, model-based Gait energy image (GEI) [71] features are also extracted to perform view-invariant and scale-independent gait recognition.

In the following subsections, we described the proposed method of gait cycle identification to compute average movement speed and 3D moments from the Spatio-temporal GEI shape feature, using the segmented silhouettes.

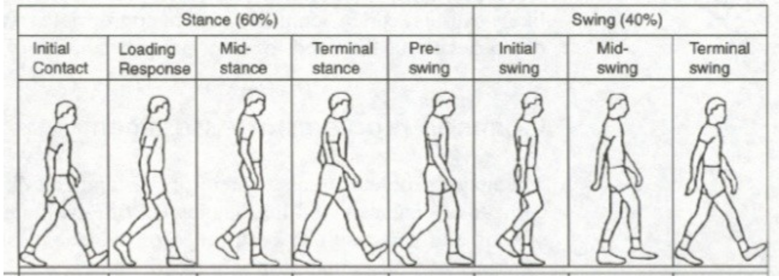


Figure 5.4: Gait Cycle Definition

5.2.1 Gait Cycle Indentification using Frontal Gait

In this section, we first define the Gait Cycle and then describe the proposed approach to identify Gait Cycle using only Frontal Gait information.

The Gait Cycle [143] can be defined as the time interval between two successive occurrences of the repetitive phases while walking. The gait cycle involves two principal stages: the stance phase and the swing phase. The stance phase occupies 60% of the gait cycle, while the swing phase occupies only 40%, as explained in Figure 5.4. The stance phase consists of Initial Contact, Loading Response, Midstance, Terminal Stance, and Pre-swing. Whereas, the swing phase is composed of Initial Swing, Mid Swing, and Terminal Swing. Stance phase begins with the heel strike – this is the moment when the heel begins to touch the ground, but the toes did not yet touch. We can see from Figure 5.4, during the Stance phase, in the Midstance position, the difference between the lower points (or pixel locations) of the two limbs is maximized. Similarly, in the Midswing position of the swing phase, in-between the Initial Swing and the Heel Strike, the distance between the lower points of the two limbs is maximized. Whereas, during the Terminal-swing through Loading response stages, the distance between the lowest white pixel of the two limbs is minimized.

Following this specific attribute of the gait cycle, we can analyze gait cycle from the frontal silhouette. In Figure 5.6, we can see that in the silhouette bounding box of frame 138 and 152 the difference between the lowest white pixel of the two limbs is maximum which indicates the successive events of the Midstance through Midswing phase. Moreover, in the silhouettes of frame 144 and frame 158, the difference between the lowest white pixel of the two limbs is minimum, which signifies the successive events of Pre-swing through Terminal swing. Therefore, we can identify the entire gait cycle from the sequence of frontal gait silhouettes starting from the Initial Contact (frame 135) through Terminal swing (frame 158) in Figure 5.6.

Identifying the gait cycles from the gait video is usually the initial step in gait analysis for separating the periodic occurrences of the walking sequence. Majority of the techniques [71, 145] in the literature perform the detection of the gait cycle using profile gait view or multiple gait views due to the ease of discrimination of different gait phases as described earlier. As per biological studies [146, 147] of the human gait cyclic phases during walking, the body pose changes periodically and the upper and lower limbs move symmetrically. Since the width and height of the bounding box of the binary silhouette directly depends on the limbs fluctuation, we represent the Gait fluctuation as a periodic function which depends on the silhouette's width and height over time. In a frontal gait video, as the subject is moving towards the surveillance camera, the silhouettes height and width will be increasing in the later frames compared with the earlier ones. To compensate for these scale variations, we normalized [148] the width and height of the silhouette bounding box.

Based on the theoretical premise of the gait cycle and the experimental observations using the frontal gait video clips, we propose a gait cycle identifier which represents the periodic motion and cyclic phases as follows:

$$\begin{split} GC_{identifier}(f_t) = 0.5 * [H_{norm}(f_t) * W_{norm}(f_t) + \{||lowest\ left\ limb\ pixel \\ -lowest\ right\ limb\ pixel||\}/H(f_t)]; \end{split} \tag{5.1}$$

where, $GC_{identifier}(f_t)$ is the variable that represents the gait cycle phase for the t-th frame (f_t), $H_{norm}(f_t)$ and $W_{norm}(f_t)$ are the silhouettes bounding box height and width for the t-th frame after normalization to compensate for the scale variations. The second term in equation 5.1 is the normalized difference between the lowest white pixels of the two limbs. The multiplier 0.5 is used to normalize the value of the Gait Cycle identifier variable. The plot of the $GC_{identifier}(t_f)$ against the sequence of frames is shown in Figure 5.5.

5.2.2 Three Dimensional Moments from the Spatio Temporal Gait Energy Image

After the silhouettes are segmented from each of the video frames, their heights are first normalized with respect to the frame height. The average silhouette image or the Gait Energy Image (GEI) [71] represents the principal shape of the human silhouette and its change over a sequence of frames in a gait cycle. A pixel with higher intensity value in the GEI indicates that the human body was present more frequently at this specific position. Equation 5.2 is used for obtaining the pixel values of the GEI:

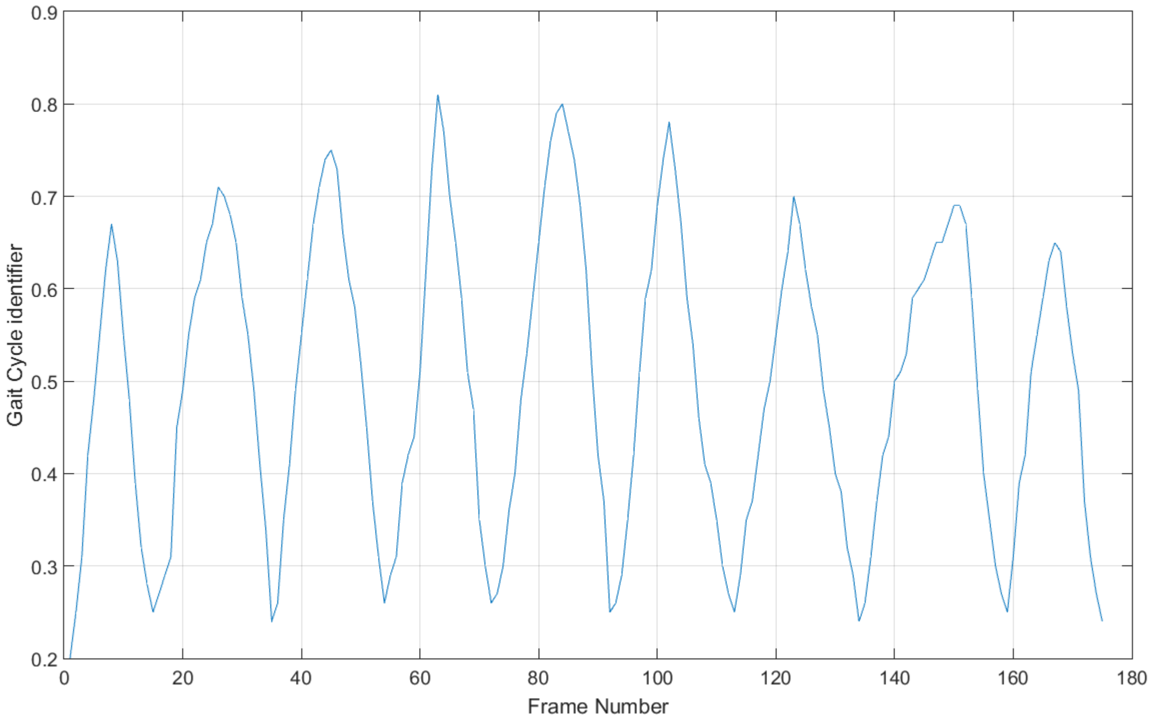


Figure 5.5: Gait Cycle Plot Using GC identifier

$$G(x, y) = \frac{1}{F} \sum_{t=1}^{F} B_t(x, y), \tag{5.2}$$

where, t stands for the temporal frame number from which the silhouette is obtained, F is the total number of frames in a complete gait cycle, $B_t(x, y)$ stands for the binary silhouette. Spatio-temporal GEI or the periodic gait volume $V(x, y, n)$ is obtained from the GEIs computed using the gait cycles in a gait video clip, where n represents the gait cycle number.

Even though, GEI suffers from some information loss of the details, it has numerous benefits compared with the representation of binary silhouettes as a temporal sequence. Since GEI is the average of a sequence of silhouettes, it is not very sensitive to errors in the silhouette segmentation in the individual frames. The robustness of the GEI is improved by discarding pixels with the energy values lower than a predefined threshold.

Shape analysis is a complex problem due to the presence of noise, and in certain cases, variations between shapes result in significant changes in the measured feature values. To recognize objects from their shape, features such as eccentricity, Moments, Euler number, compactness, and convexity are widely used in the literature [92]. Moments or central moments are used as quantitative measures for shape description [93]. Hu *et. al.* [93] derived a set of moment invariants for various geometric shapes. Moments are widely used in various complex shape based object recognition [82] due to the fact that they are invariant to orientation.

Three-dimensional raw moments for the Spatio-temporal GEI or periodic gait volume for each gait cycle can be represented as:

$$3DMoment_{p_1p_2p_3} = \sum_{x\in\mathbf{x}}\sum_{y\in\mathbf{y}}\sum_{n\in\mathbf{n}} x^{p_1}\cdot y^{p_2}\cdot n^{p_3}\cdot V(x,y,n), \tag{5.3}$$

where, $O = p_1 \cdot p_2 \cdot p_3$ is the 3D moment's order. For any translation, *e.g.*, (a,b,c), of the 3D coordinates of the center of mass of the object, the change in the three dimensional moments $3DMoment_{p_1p_2p_3}$ can be represented as:

$$\overline{3DMoment}_{p_1p_2p_3} = \sum_{x\in\mathbf{x}}\sum_{y\in\mathbf{y}}\sum_{n\in\mathbf{n}} (x+a)^{p_1}\cdot (y+b)^{p_2}\cdot (n+c)^{p_3}\cdot V(x,y,n), \tag{5.4}$$

When the center of mass $(\overline{x},\overline{y},\overline{n})$ is at the origin, the raw moments and the central moments are the same. Thus, the central moment $\mu_{p_1p_2p_3}$ can be represented by replacing a, b, c with the mean value of x, y, n respectively:

$$\begin{aligned}\mu_{p_1p_2p_3} = \sum_{x\in\mathbf{x}}\sum_{y\in\mathbf{y}}\sum_{n\in\mathbf{n}} &(x-\overline{x})^{p_1}\cdot (y-\overline{y})^{p_2} \\ &\cdot (n-\overline{n})^{p_3}\cdot V(x,y,n).\end{aligned} \tag{5.5}$$

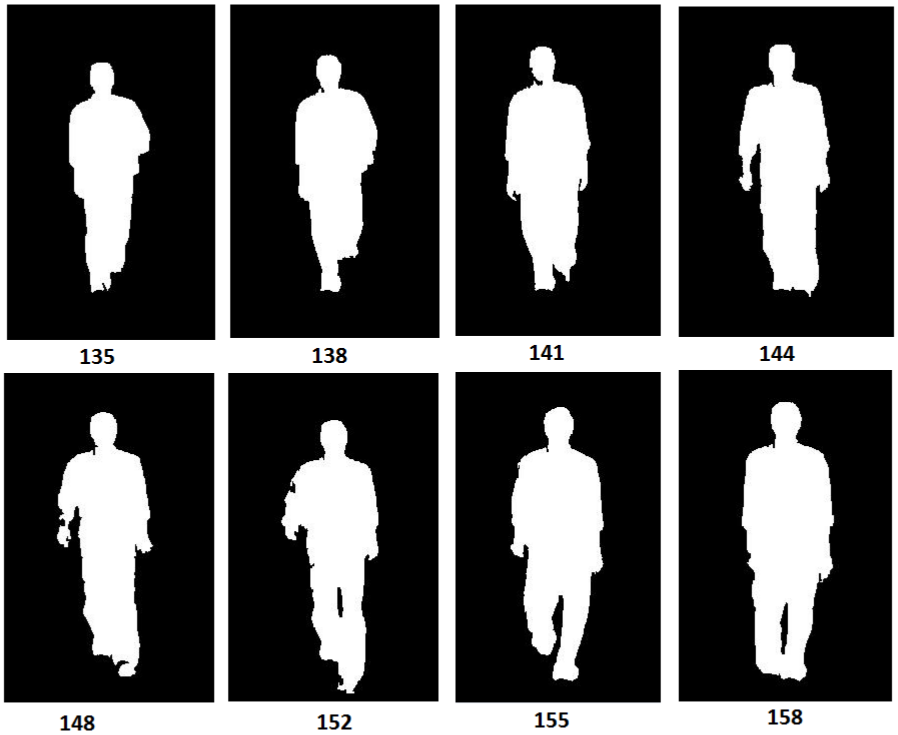


Figure 5.6: Gait Cycle Estimation

Here,

$$\overline{x} = \frac{m_{100}}{m_{000}}; \quad \overline{y} = \frac{m_{010}}{m_{000}}; \quad \overline{n} = \frac{m_{001}}{m_{000}}, \tag{5.6}$$

where, m_{000} is the zeroth spatial moment, and m_{100}, m_{010}, and m_{001} are the x, y, and n componets of the first spatial moment, respectively. The pixel on the pereodic gait volume, *e.g.*, $[x_j(n), y_j(n)]$, of $V(x, y, n)$ is the jth point that belongs to the n-th gait cylce. Hence, the 3D central moment of the Spatio-temporal GEI or the pereodic gait volume can be represented as:

$$\mu_{p_1 p_2 p_3}^{GEI_{vol}} = \sum_{n \in \mathbf{n}} \sum_{j=1}^{P(n)} (x_j(n) - \overline{x})^{p_1} \cdot (y_j(n) - \overline{y})^{p_2} \cdot (n - \overline{n})^{p_3}, \tag{5.7}$$

where $P(n)$ is the total number of pixels on the pereodic gait volume for gait cycle n.

Following the method mentioned in the previous two sections, we obtained the scale and translation invariant three-dimensional moments of the periodic gait volume ($\mu_{p_1p_2p_3}^{GEI_{vol}}$). Additionally, the average number of frames in the gait cycles identified using the frontal walking video clip is used as the average movement speed of the subject. We used these two components together to obtain the gait signature used for classifying the subjects through Gait recognition.

5.3 Low Resolution Face Feature Representation

In this section, we describe the proposed algorithm for Low Resolution face recognition from surveillance video clips. The description of the components used in the algorithm are detailed in the subsequent sections.

Algorithm 3 Low Resolution Face Recognition

```
1. Detect faces in the video surveillance frames.
2. Use a Super-resolution technique to obtain High-resolution from the Low-resolution detected face images.
3. Perform illumination and pose normalization.
4. Register the preprocessed and normalized face regions, followed by synthesizing them using Curvelet and Inverse Curvelet transformations.
5. Extract Local Binary Pattern (LBP) and Gabor Features from the synthesized image.
6. Perform face Recognition using the extracted features.
```

5.3.1 Super-resolution

Super-resolution (SR) [149, 150] is a class of image processing algorithms, used to enhance the resolution of low resolution images. SR algorithms can be used to enhance the resolution of an image from single or multiple low resolution images.

Interpolation techniques such as nearest neighbor, bilinear and cubic convolution are widely used for SR processing of the LR images in the literature.

The two key components of a digital imaging system are the sensor and the lens, those introduce two types of image degradation, specifically optical blur and limitation on the highest spatial frequency that can be recorded. The sensor is constructed from a finite number of discrete pixels which results in the presence of so-called aliased components in the sensor output. These correspond to high spatial-frequency components in the scene that are higher than frequencies that the sensor can handle and should not normally be present in the output. These are the key components used by the SR algorithms to obtain the HR representation. The available SR algorithms can be categorized broadly into two major classes: reconstruction-based SR and recognition-based SR. The reconstruction-based methods are suitable for synthesizing local texture resulting in better visualization and do not incorporate any specific prior information. However, recognition-based SR [149, 150] algorithms try to detect or identify certain pre-configured patterns in the low resolution data.

The Recognition based SR algorithms [149] learn a mapping correspondence between low and high resolution image patches from the training LR and HR images, which can be directly applied to a test LR image to construct the HR counterpart. In the training phase densely overlapping patches are cropped from the low-resolution and high-resolution image pair. Followed by jointly training two dictionaries for the low- and high-resolution image patches by enforcing the similarity of sparse representation for each image pair. Given the trained LR and HR dictionaries and a test LR image, the algorithm obtains its HR representation in three steps. First, densely overlapping patches are cropped from the LR input image and pre-processed (*i.e.*, normalization). Second, the sparse coefficients obtained from the LR dictionary for the LR test image patches are passed into the high-resolution dictionary for reconstructing the high-resolution patches. Finally, the overlapped HR reconstructed patches are aggregated (*i.e.*, weighted averaging) to produce the final output.

Convolutional neural networks (CNN) [151] was developed several decades ago and deep Conv Nets [152] have recently been popular among researchers primarily due to its success in image classification. CNN is a specific artificial neural network topology, that is inspired by biological visual cortex, formed by stacking multiple stages of feature extractors. CNN have also been used successfully for other computer vision applications, such as object detection, face recognition, and pedestrian detection.

Dong *et. al.* [150] proposed a CNN based SR algorithm, which directly learns an end-to-end mapping between the low and high-resolution image pair. The three components of the pipeline in the Recognition based SR algorithms are represented as different layers of CNN, which efficiently optimize the entire SR implementation through the CNN. The mapping is represented as a deep convolutional neural network (CNN) that takes the low-resolution image as the input and outputs the high-resolution one. The first step is patch extraction and representation. The Recognition based SR algorithms [149] use the densely extracted patches and then represent them by a set of pre-trained bases such as PCA, DCT and Haar. This is equivalent of convolving the image by a set of filters, each of which is a basis. Thus, the first layer of the CNN can be expressed as:

$$F_1(Y) = max(0, W_1 * Y + B_1), \tag{5.8}$$

where W_1 and B_1 represent the filter weights and biases respectively, '$*$' denotes the convolution operation. W_1 is of size $c \times f_1 \times f_1 \times n_1$, corresponds to n_1 filters of spatial size $f_1 \times f_1$ and c stands for the number of channels in the image, that applies n_1 convolutions on the image. The output is composed of n_1 feature maps. B_1 is an n_1-dimensional bias vector, whose each element is associated with a filter. The second component of the Recognition based SR algorithm pipeline can be represented using the Non-linear mapping step of CNN. As shown in Equation 5.8, the first layer extracts an n_1-dimensional feature vector for each patch. In the second operation, each of these n_1-dimensional vectors is mapped into an n_2-dimensional vetor. The operation of the second layer can be represented as:

$$F_2(Y) = max(0, W_2 * F_1(Y) + B_2), \tag{5.9}$$

here W_2 is of size $n_1 \times f_2 \times f_2 \times n_2$, corresponds to n_2 filters of spatial size $n_1 \times f_2 \times f_2$, and B_2 is n_2-dimensional bias. Each of the output n_2-dimensional vectors is a representation of a high-resolution patch that will be used for SR reconstruction. Finally, the reconstruction step in the Recognition based SR algorithm pipeline produces the final HR image by averaging the overlapping high-resolution patches. The averaging can be considered as a pre-defined filter on a set of feature maps, where each position is the flattened vector form of a high-resolution patch.

$$F(Y) = W_3 * F_2(Y) + B_3, \tag{5.10}$$

where W_3 is of size $n_2 \times f_3 \times f_3 \times c$, corresponds to c filters of a spatial size $n_2 \times f_3 \times f_3$, and B_3 is a c-dimensional bias vector. The values of the parameters n_1, n_2, n_3, f_1, f_2, and f_3 used in the experiements are detailed in the Experimental Result section 5.4.4.

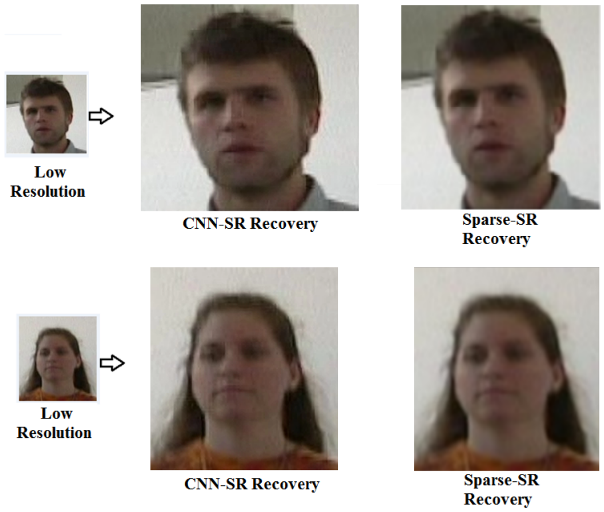


Figure 5.7: Super-resolution recovery of the LR face images

The Super-resolution pre-processing technique is used to obtain High-resolution representation of the Low-resolution face images as shown in Figure 5.7. We can see that the performance of the CNN based Super resolution recovery method face is better than the performance of Sparse based Super resolution technique.

5.3.2 Illumination and Pose Invariance

In this section, we explain the preprocessing steps for normalizing the low resolution images with respect to illumination and pose variations.

5.3.2.1 Illumination Normalization

It has been proven in the literature, that illumination variations are among the primary problems in biometric authentication. We adopted the $Self-quotient\ image(SQI)$ [153] to normalize the illumination variations in the low resolution facial images. SQI incorporates an edge-preserving filtering technique to minimize the spectral variations present in the illumination.

The Lambertian model can be factorized into two parts, the intrinsic part, and the extrinsic part:

$$I(x,y) = \rho(x,y)\ n(x,y)^T \cdot\ s = F(x,y) \cdot\ s, \tag{5.11}$$

where ρ is the albedo and n is the surface normals. $F = \rho n^T$depends on the albedo and surface normal of an object and hence is an intrinsic factor, where F represents the identity of a face. However, s is the illumination and is an extrinsic factor. Separating the two factors and removing the extrinsic component is a key to achieve a robust face recognition by normalizing the effect of varying illumination.

The SQI image Q of an image I can be represented as:

$$Q = \frac{I}{\widehat{I}} = \frac{I}{P * I}, \tag{5.12}$$

where $\widehat{I}$ is the smoothed version of I, P is the smoothing kernel, and the division is pixel-wise. SQI [153] achieves the removal of extrinsic component s in Eqn. 5.11 through a two-step process. First, an illumination estimation step: the extrinsic factor is estimated to generate a synthesized smooth image, which has same illumination and shape as the input but a different albedo. Second, an illumination effect subtraction step: the illumination is normalized by computing the difference between the logarithms of the albedo maps of the input and the synthesized images, $(log\rho_0 - log\rho_1)$.

5.3.2.2 Pose Correction

Pose variations present a major problem in real-world face recognition applications. Since the human face is approximately symmetric, if it is in the frontal pose with no rotations, the matrix containing the face image (F) will have the lowest rank. Employing the above-stated principle, Zhang *et. al.* [154] proposed *transform invariant low-rank textures (TILT)* to normalize the pose of a rotated frontal face and remove minor occlusions.

$TILT$ [154] tries to find a transformation (Euclidean, affine, or projective) matrix τ,through modelling the face rotation using an error matrix E, *s.t.* $\widehat{F} \, * \, \tau = F + E$, where $\widehat{F}$ represents the deformed and corrupted face and F is the corrected low-rank face image, by optimizing the following equation:

$$\underset{F,E,\tau}{min} \; rank(F) + \gamma ||E||_o \quad s.t. \quad \widehat{F} \, * \, \tau = F + E \tag{5.13}$$

where $||E||_o$ is the l_0-norm of the error matrix, *i.e.*, number of non-zero elements. It actually finds the corrected low-rank face image (F) with the lowest possible rank and the error with the lowest number of non-zero elements, which satisfy the above condition. γ trades off the rank of the matrix and the sparsity of the error.

Optimizing the rank function and the l_0-norm in the above equation is very challenging. Therefore, they are substituted by their convex surrogates. Since the rank of a matrix is equivalent to the number of its non-zero singular values, we can substitute the rank(F) by its nuclear norm $||F||_*$, which is the sum of its singular values. Moreover, l_0-norm is substituted by l_1-norm, which is the sum of the absolute values of the elements of the matrix. Additionally, the constraint $\widehat{F} \, * \, \tau = F + E$ is non-linear, by linearizing the constraint around its current estimate through an iterative process, the optimization problem becomes as follows:

$$\underset{F,E,\Delta\tau}{min} \; ||F||_* + \gamma ||E||_1 \quad s.t. \quad \widehat{F} \, * \, \tau + \nabla\widehat{F}\Delta\tau = F + E, \tag{5.14}$$

where ∇ represents the Jacobian. Finally, we train a binary classifier using local features (Local Binary Pattern) to remove the false positive frames detected by the Adaboost face detector.

5.3.3 Registration and Synthesizing Low Resolution Face Images

In this section we describe the image registration of the preprocessed and normalized face regions, and synthesizing them using Curvelet and Inverse Curvelet transformation.

5.3.3.1 Registration

We adopted the subspace-based holistic registration (SHR) method [155], which was proposed to perform registration on low-resolution face images. The majority of the automatic landmark-based registration methods can only perform accurate registration on high resolution images. However, SHR is able to obtain a user independent face model using Procrustes transformation by incorporating the image edges as feature vectors to register low-resolution face images. The best registration parameters are iteratively obtained through the downhill simplex optimization technique by maximizing the similarity score between the probe and the gallery image. The registration similarity is calculated using the probability that the probe and gallery face images are correctly aligned in a face subspace by computing the residual error in the dimensions perpendicular to the face subspace.

The first step of obtaining the subject independent face model to perform the registration is to compute the edges in the low resolution facial image. Gaussian kernel derivatives of the LR face images are calculated in the x and y directions respectively using G_x and G_y as follows:

$$\begin{aligned} G_x(x,y) &= \frac{-x}{2\pi\sigma^4} exp(-\frac{x^2+y^2}{2\sigma^2}), \\ G_y(x,y) &= \frac{-y}{2\pi\sigma^4} exp(-\frac{x^2+y^2}{2\sigma^2}), \end{aligned} \tag{5.15}$$

The derivatives H_x and H_y of the images are obtained by convoluting the LR face image with G_x and G_y resulting in the "*edge images*" used for the registration purpose. Procrustes transformation is used to align the probe image to the gallery image by correcting the variations of scale by a factor f, rotatation with an angle α, and translation of $\mathbf{u}$, while preserving the distance ratios. Given a pixel location $\mathbf{p} = (x,y)^T$, the transformation $U_\theta \mathbf{p}$ on a pixel location can be represented as:

$$U_\theta \mathbf{p} = f R(\alpha)\mathbf{p} + \mathbf{u}, \tag{5.16}$$

where, $\theta = \{\mathbf{u}, \alpha, f\}$ represent the registration parameters, and $R(\alpha)$ is the rotation matrix. The transformation of the entire probe image to perform the registration operation is obtained by applying U_θ on the computed "*edge images*" as follows:

$$T_\theta H(\mathbf{p}) = H(U_\theta^{-1}\mathbf{p}). \tag{5.17}$$

where, $H = \sqrt{H_x^2 + H_y^2}$. Thus, a registered and aligned image, $T_\theta H(\mathbf{p})$, is obtained through backward mapping and interpolation by utilizing the optimal registration parameter θ found using simplex optimization technique.

5.3.3.2 Synthesizing

To enhance the spectral features for face recognition, image synthesizing methods [156] are very popular in the literature. The synthesizing methods available in the literature can be broadly categorized into two classes, one performs the syhtnesis in the spatial domain and the other in the frequency domain. In this paper, we adopted the Curvelet-based image synthesis [157] which uses the Curvelet coefficients [158] to represent the face.

Curvelet transform has improved directional capability, better ability to represent edges and other singularities along curves as compared to other traditional multi-scale transforms, *e.g.* wavelet transform. First, curvelet transforms are applied to the sequence of registered face images. The smallest low-frequency components are represented by the coarse Curvelet coefficients and the largest high-frequency components are represented by the fine Curvelet coefficients. For the image sequence $I_1, I_2, ..., I_n$ the Curvelet coefficients can be represented as $C_{I_i}\{j\}\{l\}$, where $i = 1, 2, ..., n$ represent the image image sequence to be synthesized, and j, l, is the scale and direction parameters, respectively. The components of the first scale where $j = 1$ represent the low-frequency parts of the face images, and the components associated to other scales ($j > 1$) represent the high-frequency parts. The minimum components between each $C_{I_i}\{1\}\{l\}$, where scale $j = 1$, and ($i = 1, 2, ..., n$), and the maximum components between each $C_{I_i}\{j\}\{l\}$, where ($j = 2, ..., 5$), and $i = 1, 2, ..., n$ are retained for the synthesized Curvet coefficients. Inverse Curvelet transformation of the synthesized Curvelet feature vector generates the synthesized image used for feature extraction.

5.3.4 Feature Extraction

We obtain LBP and Gabor features from the fused image and compare their performance for recognition. In the subsequent sections, we describe the LBP and Gabor feature extraction techniques.

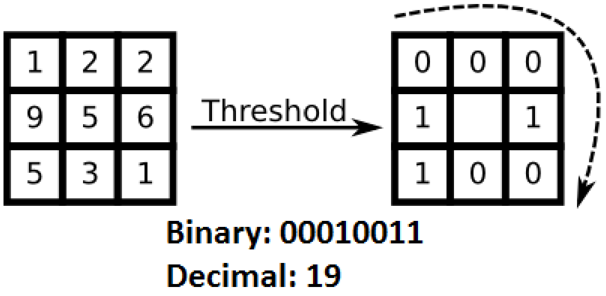


Figure 5.8: LBP feature, circular (8,1) neighborhood

5.3.4.1 LBP

The original LBP operator, introduced by Ojala *et. al.* [159], is a powerful method for texture description. The operator labels the pixels of an image by thresholding the 3x3-neighbourhood of each pixel with the center value and considering the result as a binary number. Then, the histogram of the labels can be used as a texture descriptor. See Figure 5.8 for an illustration of the basic LBP operator.

Later the operator was extended to use neighbourhoods of different sizes. Using circular neighborhoods and bilinearly interpolating the pixel values allow any radius and number of pixels in the neighborhood. For neighborhoods we use the notation (P, R) which means P sampling points on a circle of radius of R. Figure 5.9 shows an example of the circular neighborhood (8,2). Another extension to the original operator uses what is called uniform patterns. A Local Binary Pattern is called uniform if it contains at most two bitwise transitions from 0 to 1 or vice versa when the binary string is considered circular. For example, 00000000, 00011110 and 10000011 are uniform patterns. Ojala *et. al.* [159] noticed that in their experiments with texture images, uniform patterns account for a bit less than 90% of all patterns when using the (8,1) neighborhood and for around 70% in the (16,2) neighborhood.

An extension of LBP-based face description method is proposed by Ahonen *et. al.* [160]. The facial image is divided into local regions ($k \times k$ window) and LBP texture descriptors are extracted from each region independently. The descriptors are then concatenated to form a global description of the face that describes the facial image in a high dimensional feature space. Window sizes used for experiment purposes are $k = 3, 5, 7$.

5.3.4.2 Gabor

2D Gabor filters [117] are used in a broad range of applications [119] to extract scale and rotation invariant feature vectors. In our feature extraction step, uniform down-sampled Gabor wavelets are computed for the detected regions using equation 5.18, as proposed in [120]:

$$\psi_{\mu,\nu}(z) = \frac{||k_{\mu,\nu}||^2}{s^2} e^{(\frac{-||k_{\mu,\nu}||^2||z||^2}{2s^2})}[e^{ik_{\mu,\nu}z} - e^{\frac{-s^2}{2}}], \tag{5.18}$$

where $z = (x, y)$ represents each pixel in the 2D image, $k_{\mu,\nu}$ is the wave vector, which can be defined as $k_{\mu,\nu} = k_\nu e^{i\phi_\mu}$, $k_\nu = \frac{k_{max}}{f^\nu}$, k_{max} is the maximum frequency, and f is the spacing factor between kernels in the frequency domain, $\phi_\mu = \frac{\pi\mu}{2}$, and the value of s determines the ratio of the Gaussian window width to wavelength. Using equation 5.18, Gabor kernels are generated from one filter using different scaling and rotation factors. In this paper, we used five scales, $\nu \in 0, ..., 4$ and eight orientations $\mu \in 0, ..., 7$. The other parameter values used are $s = 2\pi$, $k_{max} = \frac{\pi}{2}$, and $f = \sqrt{2}$.

Gabor features are computed by convolving each Gabor wavelet with the synthesized Super Resolution preprocessed LR face images, as follows:

$$C_{\mu,\nu}(z) = T(z) * \psi_{\mu,\nu}(z), \tag{5.19}$$

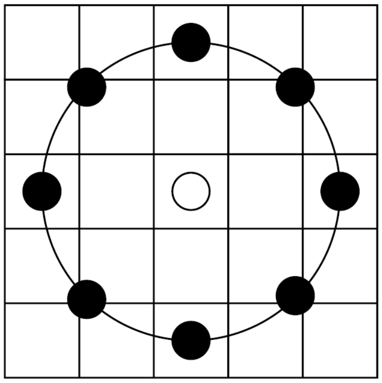


Figure 5.9: Circular (8,2) neigbourhood.

where $T(z)$ is the face image, and $z = (x, y)$ represents the pixel location. The feature vector is constructed out of $C_{\mu,\nu}$ by concatenating its rows.

5.4 Experimental Results

In this section, we first describe the Face and Ocular Challenge Series (FOCS) [161] dataset. Then, we demonstrate the experiments and results of the frontal gait recognition and low resolution face recognition followed by the score level fusion to obtain the multi-modal recognition.

5.4.1 FOCS Dataset

The video challenge dataset, Face and Ocular Challenge Series (FOCS) [161], contains video sequences of individuals, acquired on different days. Students from The University of Texas at Dallas, between the age group of 18 and 25, volunteered for the data collection. The FOCS dataset is collected in two sessions, where in the second duplicate session of data collection the subjects have a different hairstyle, different clothing, and may be otherwise different in appearance.

The FOCS database contains a variety of still images and videos of a large number of individuals taken in a variety of contexts. For our experiments, we used the frontal walking video sequences. In the frontal walking video sequences, the subject walks parallel to the line of sight of the camera, approaching the camera, but veering off to the left while reaching in front of the camera. These frontal walking video sequences capture the subject from the start point until he/she goes out of view. Thus, the time varies somewhat for each subject due to their walking speed, but on average it is approximately 10 seconds. The FOCS frontal walking video sequences contain videos that are acquired from 136 unique subjects. The number of samples per subject varies. Out of 136 subjects, 123 subjects have at least 2 videos. We used data from these 123 subjects for our experiments, where one of the video clips is randomly chosen for training and the other is used for testing.

5.4.2 Experimental Setup

To perform the multimodal recognition, we first segment the frontal Gait silhouette from the background and detect the low resolution face images from the frontal walking video sequences as described in section 5.1. In order to evaluate the proposed algorithm, we perform the frontal gait recognition and the low resolution face recognition experiments as two seperate components. Later, we use the match score level fusion scheme to fuse the individual recognition results.

5.4.2.1 Frontal Gait Recognition

Once the binary gait silhouette is acquired, we obtained the scale and translation invariant 3D moments of the Spatio-temporal GEI or the periodic Gait volume, and the average number of frames in the identified gait cycles in the frontal walking video clips to prepare a high dimensional feature vector as described in section 5.2. The frontal gait features were classified using a k-nearest neighbor classifier (k-NN), where a test gait feature vector belongs to the class that minimizes the similarity distance between the gallery and the probe gait feature vector.

We performed quantitative experiments using different moment orders $O = p_1 \cdot p_2 \cdot p_3$. The best recognition performance was obtained when $p_1 = p_2 = p_3 = 10$. The results of frontal gait recognition are presented in Table 5.1. We can see that the recognition performance when using the 3D moments of periodic gait volume is better than the performance when using the average movement speed feature representation. However, concatenating the feature vectors together improved the gait recognition performance. Table 5.2 shows a comparsion between the frontal gait recognition performance achieved by the proposed approach and the recognition accuracy of the state-of-the techniques on the FOCS dataset. The proposed frontal gait recognition system achieves a rank-one recognition rate of 93.5% on the 123 subjects of FOCS dataset.

5.4.2.2 Low Resolution Face Recognition

The first step of LR face recognition is to detect the low resolution faces using the Adaboost detector from the video surveillance frames as described in section 5.1. The proposed LR face recognition Algorithm 3 is described in section 5.3. After employing the CNN based Super resolution technique to obtain the High Resolution equivalent of the LR faces, we perform the illumination and pose normalization steps. The sizes of the pre-processed face images varies between 40×40 pixels and 180×180 pixels. To effectively leverage the high-frequency information present in the pre-processed face images we separate the face images into two classes. The face images of size less than 96×96 pixels are labeled as $Class-1$ and those that are greater than 96×96 pixels are labeled as $Class-2$. We use the face image of size 72×72 pixel in $Class-1$ as the base or template image to apply the SHR registration technique [155], described in Section 5.3.3.1, to register all the face images that belong to $Class-1$ after rescaling them to 72×72 pixels. Similarly, the face image of size 120×120 pixels in $Class-2$ is used as the base or template image to apply the SHR registration technique [155] to register all the face images that belong to $Class-2$ after rescaling them to 120 × 120 pixel. After performing the image synthesis using the Curvelet coefficients as described in Section 5.3.3.2 of the face images in $Class-1$ and $Class-2$ separately we obtain two synthesized face images for each surveillance video clip. We extract the LBP and Gabor feature vectors, as mentioned in section 5.3.4, from the two synthesized face images and perform feature concatenation to obtain the composed LBP and Gabor features, which represent the LR face in the surveillance video clip. The obtained LBP and Gabor feature vectors are used separately to compare their performance in LR face recognition. For each of the 123 subjects used in the performance evaluation, the feature vector obtained from one randomly chosen surveillance video clip is used to build the model and the one obtained from the other video is used for testing.

We compare the performance of the proposed LR face recognition technique using the CNN based Super resolution [150] with the following baseline algorithms. It is worth noting that all the comparisons are based on the same training/test set.

1) LR face recognition without any Super Resolution preprocessing technique.

2) LR face recognition using Bicubic Interploation Super Resolution preprocessing

technique.

3) LR face recognition using Sparse Super Resolution [149] preprocessing technique. The obtained LR face features were classified using a k-nearest neighbor classifier (k-NN), where a test LR facial feature vector belongs to the class that minimizes the similarity distance between the gallery and the probe feature vector. The result of Low Resolution Face Recognition is presented in Table 5.3. We can see that the performance when using the local feature representation LBP is better than the performance when using global feature representation or Gabor features. Moreover, by employing the CNN based Super resolution technique the LR face recognition performance is increased to 82.91% compared with 72.36% without any SR pre-processing of the LR face images.

5.4.3 Multimodal Recognition Accuracy

Score level fusion techniques are very popular in multimodal biometrics applications specifically in the application of fusing Face and Gait [11,79]. In our experiment, results from the different classifiers were combined directly using the Sum, Max, and Product rules.

To prepare for fusion, the matching scores obtained from the different matchers are transformed into a common domain using a score normalization technique. Later, the score fusion methods are applied. We have adopted the *Tanh* score normalization technique [134], which is both robust and efficient, defined as follows:

$$s_j^n = \frac{1}{2}\left\{tanh(0.01(\frac{s_j - \mu_{GH}}{\sigma_{GH}})) + 1\right\}, \tag{5.20}$$

where s_j and s_j^n are the match scores before and after normalization, respectively. μ_{GH} and σ_{GH} are the mean and standard deviation estimates of the actual score distribution given by Hampel estimators [135], respectively. Hampel's estimators are based on the influence functions ψ which are odd functions and can be defined for any x (matching score, s_j, in this paper) as follows:

$$\psi(x) = \begin{cases} x, & 0 \leq |x| < a, \\ a \ sign(x), & a \leq |x| \leq b, \\ \frac{a(r-|x|)}{r-b} \ sign(x), & b \leq |x| \leq r, \\ 0, & r \leq |x|, \end{cases} \tag{5.21}$$

where

$$sign(x) = \begin{cases} +1, & \text{if } x \geq 0, \\ -1, & \text{otherwise}, \end{cases} \tag{5.22}$$

In equation 5.21, the values of a, b, and r in ψ, reduce the influence of the scores at the tails of the distribution during the estimation of the location and scale parameters , *i.e.*, μ_{GH} and σ_{GH} in equation 5.23. The normalized match scores of synthesized face images of the gallery and probe and the normalized match scores of gaits of the gallery and probe from the same video clips are fused based on different match score fusion techniques. Let, s^n_{jF} and s^n_{jG} be the normalized match scores obtained from a specific video clip for the face and gait, respectively. The unknown test subject is classified to class C if the fused match score corresponding to the class C is maximum compared to all other classes in the gallery:

$$FR\{s^n_{CF}, s^n_{CG}\} = max \ FR\{s^n_{jF}, s^n_{jG}\}; j \in (1, 2, ..., N) \tag{5.23}$$

where, $FR\{,\}$ represents the fusion rule, and N represents the number of enrolled indivuduals in the gallery. In this paper, we use Sum, Max, and Product rules.

Table 5.1: Frontal Gait Recognition

Feature Vectors Used	Rank-1 accuracy
3D Moments	88.62% (109 out of 123)
Average movement speed	69.11% (85 out of 123)
3D Moments and Average movement speed	**93.5%** (115 out of 123)

Table 5.2: Comaprison of Frontal Gait Recognition Accuracy

Method	Rank-1 Frontal Gait Recognition Accuracy
Wang et al. [12]	69.11%
Chen et al. [13]	89.43%
Goffredo et al. [78]	91.06%
This Work	**93.50%**

The result of the fused multimodal recognition are presented in Table 5.4. We can see that the fusion based on the Sum rule of the Frontal Gait and the LR Face results in the best recognition accuracy.

5.4.4 Parameter Selection for the CNN Super Resolution

We tested the performance of the proposed LR face recognition with different parameters of the Convolution Neural Network. The number of layers in the CNN

Table 5.3: Low Resolution Face Recognition

Features Used	Super Resolution Technique	Rank-1 accuracy
LBP	None	72.36% (89 out of 123)
Gabor	None	70.73% (87 out of 123)
LBP	Bicubic	73.98% (91 out of 123)
Gabor	Bicubic	71.54% (88 out of 123)
LBP	Sparse	75.61% (93 out of 123)
Gabor	Sparse	72.36% (89 out of 123)
LBP	SRCNN	**82.92%** (102 out of 123)
Gabor	SRCNN	79.67% (98 out of 123)

Table 5.4: Comparison of Multimodal Recognition Fusion Scheme

Fusion Rule	Rank-1 accuracy
Sum Rule	**95.9%** (118 out of 123)
Max Rule	94.3% (116 out of 123)
Product Rule	93.5% (115 out of 123)

network is varied between 3 and 5, where the best performance was obtained when using 3 layer architecture. The Reognition based Super Resolution algorithm has three distinct steps, which signifies the optimal performnace of the CNN with 3 layers. Experiments are conducted by varying the numbers of filters n_1 and n_2 (refer to equations 5.9 and 5.10) of the CNN architecture. Three sets of network parameters were used for experimental purposes ($n_1 = 32$ and $n_2 = 16$), ($n_1 = 64$ and $n_2 = 32$), and ($n_1 = 128$ and $n_2 = 64$). The best performance was achieved with the parameters ($n_1 = 128$ and $n_2 = 64$). The Super resolution restoration speed decreases with the increase of the size of the filters. To obtain a reasonable trade off we set the number of the filters n_1 and n_2 to 128 and 64, respectively. Moreover, the size of filters f_1, f_2, and f_3 (refer to equations 5.8, 5.9, and 5.10) are varied between $(9, 1, 5)$, $(9, 3, 5)$, and $(9, 5, 5)$. The best accuracy and performance trade off was obtained using the parameter values of $f_1 = 9$, $f_2 = 3$, and $f_3 = 5$. With the above mentioned pararmeter settings 8×10^8 iterations of backpropagations were needed to achieve convergance.

CHAPTER 6

Conclusion and Future Work

In this dissertation, we have presented a series of novel biometric methods for uni-modal 3D ear and multi-modal ear and face recognition using facial video clips. The motivating factors underlying to use of the proposed biometric systems are the high availability of 3D scanners, the nature of data collection is non-intrusive, and the close physical proximity of the different modalities to each other.

In Chapter 2 we presented a fully automated system for 3D ear segmentation. Utilizing the tree-structured graph model and active contour segmentation we proposed the first fully automated 3D ear-region segmentation algorithm from the range scan of the face profile. The uniqueness of this study exists in the fact that instead of only finding the smallest possible bounding box that contains the ear, we perform accurate segmentation of ear through bounding box detection. To demonstrate the potential of this approach and its suitability for the application, we applied our algorithm to the largest available 3D ear database (UND database collection J2). The accuracy of the proposed segmentation approach outperforms the state-of-the-art 3D ear segmentation techniques.

In Chapter 3 we described a fully automatic 3D ear classification through Indexing. The segmented 3D ear region is used for hierarchical categorization of the gallery based on the shape information and surface depth information, respectively. To prove the efficacy of the algorithm for time efficient recognition, we tested the proposed approach on the largest available 3D ear database (UND database collection J2). Compared with the results reported in the literature, the rank-one recognition accuracy obtained by the proposed approach is the highest on the UND database collection J2 and is faster than other automatic 3D ear recognition systems in the literature. The contribution of this study is obvious as the proposed hierarchical categorization of the gallery can be applied to any biometric modality for time efficient recognition. Future extensions of this work may include the use of categorization for every modality in multi-modal biometrics along with fusion at the decision level.

In Chapter 4 we proposed a system for multimodal recognition using a single biometrics data source i.e., facial video clips acquired in constrained or unconstrained environment. Using the Adaboost detector we automatically detect the modality specific regions. We used Gabor filters to extract feature vectors from the detected regions and automatically learn robust and non-redundant features by training a Supervised Stacked Denoising Auto-encoder (Deep Learning) network. The Deep Neural Network with 5 hidden layers and hidden layer neurons double of the input layer results in best recognition performance. Classification through sparse representation is used for modality specific recognition. Then, the multimodal recognition accuracy is obtained through the fusion of the modality specific recognition. We trained the algorithm using all modalities and tested the system when all the modalities are available, and in the presence of missing modalities, *i.e.*, only some of the modalities are available during classification. The results, in this case, indicate that among all possible combinations of different modalities frontal face and ear, *i.e.*, right and left ear modalities, together produce the best recognition rate. Feature vectors of 1000 elements obtained the best recognition accuracy while testing for effective feature vector length.

In Chapter 5 we introduced a system for highly accurate multimodal human identification from low resolution video surveillance footage through LR face and Frontal Gait recognition using a single biometric data source, *i.e.*, frontal walking Surveillance Video. Using the trained Adaboost detector, we automatically detect the LR face images. The frontal gait binary silhouettes are segmented using the Fast Object Segmentation algorithm. We proposed an approach for accurate identification of the gait cycles in the entire gait video clip using only frontal Gait information, then we extract the average movement speed and the shape feature. The detected LR face images are preprocessed using Super Resolution techniques to obtain the high resolution representation. This is followed by illumination and pose normalization, and image synthesis through registration. Finally, Gabor and LBP features are extracted from the synthesized face images. The Nearest neighbor classifier is used to obtain modality specific rank-1 recognition for each modality. Then, the individual recognition results are fused through the score level fusion. The results indicate that combining the LR face and the Frontal Gait modalities produce the best recognition Rank-1 accuracy compared to the performance of each modality.

Bibliography

[1] A. Abaza, A. Ross, C. Hebert, M. A. F. Harrison, and M. S. Nixon, "A survey on ear biometrics," *ACM Computing Surveys*, vol. 45, no. 2, pp. 1–35, 2013.

[2] A. Iannarelli, "Ear identification. forensic identification series," 1989.

[3] M. N. Arab, F. Karray, J. A. Saleh, and M. Alemzadeh, "Multi modal biometric systems: A state of the art survey," in *Computational Intelligence, Robotics and Autonomous Systems*, Palmerston North, New Zealand, pp. 1221–1224.

[4] S. Cadavid, M. Mahoor, and M. Abdel-Mottaleb, "Multi-modal biometric modeling and recognition of the human face and ear," in *IEEE International Workshop on Safety, Security Rescue Robotics (SSRR)*, Nov 2009, pp. 1–6.

[5] M. Mahoor and M. Abdel-Mottaleb, "A multimodal approach for face modeling and recognition," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 431–440, Sept 2008.

[6] A. Ross and A. K. Jain, "Multimodal biometrics: an overview," in *Proceedings of 12th European Signal Processing Conference*, 2004, pp. 1221–1224.

[7] P. Yan and K. Bowyer, "Biometric recognition using 3d ear shape," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 8, pp. 1297–1308, 2007.

[8] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition*, vol. 1, 2001, pp. 511–518 vol.1.

[9] Z. Huang, Y. Liu, C. Li, M. Yang, and L. Chen, "A robust face and ear based multimodal biometric system using sparse representation," *Pattern Recognition*, vol. 46, no. 8, pp. 2156 – 2168, 2013.

[10] G. Shakhnarovich and T. Darrell, "On probabilistic combination of face and gait cues for identification," in *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*, May 2002, pp. 169–174.

[11] A. Kale, A. K. Roychowdhury, and R. Chellappa, “Fusion of gait and face for human identification,” in *Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP ’04). IEEE International Conference on*, vol. 5, May 2004, pp. V–901–4 vol.5.

[12] L. Wang, T. Tan, W. Hu, and H. Ning, “Automatic gait recognition based on statistical shape analysis,” *IEEE Transactions on Image Processing*, vol. 12, no. 9, pp. 1120–1131, Sept 2003.

[13] S. Chen and Y. Gao, “An invariant appearance model for gait recognition,” in *2007 IEEE International Conference on Multimedia and Expo*, July 2007, pp. 1375–1378.

[14] H. Chen and B. Bhanu, “Human ear detection from 3d side face range images,” in *3D Imaging for Safety and Security.* Springer, 2007, vol. 35, pp. 133–155.

[15] C. Dorai and A. Jain, “Cosmos-a representation scheme for 3d free-form objects,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 10, pp. 1115–1130, 1997.

[16] H. Chen and B. Bhanu, “Contour matching for 3d ear recognition,” in *Seventh IEEE Workshops on Application of Computer Vision*, vol. 1, 2005, pp. 123–128.

[17] ——, “Human ear recognition in 3d,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 718–737, April 2007.

[18] P. Yan and K. Bowyer, “Empirical evaluation of advanced ear biometrics,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, June 2005, pp. 41–41.

[19] ——, “Ear biometrics using 2d and 3d images,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, June 2005, pp. 121–121.

[20] ——, “Multi-biometrics 2d and 3d ear recognition,” in *Audio- and Video-Based Biometric Person Authentication*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2005, vol. 3546.

[21] S. Cadavid and M. Abdel-Mottaleb, “Human identification based on 3d ear models,” in *First IEEE International Conference on Biometrics: Theory, Applications, and Systems.*, Sept 2007, pp. 1–6.

[22] ——, “3d ear modeling and recognition from video sequences using shape from shading,” in *19th International Conference on Pattern Recognition.*, Dec 2008, pp. 1–4.

[23] J. Zhou, S. Cadavid, and M. Abdel-Mottaleb, "A computationally efficient approach to 3d ear recognition employing local and holistic features," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2011, pp. 98–105.

[24] ——, "An efficient 3-d ear recognition system employing local and holistic features," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 978–991, 2012.

[25] S. Islam, R. Davies, A. Mian, and M. Bennamoun, "A fast and fully automatic ear recognition approach based on 3d local surface features," in *Advanced Concepts for Intelligent Vision Systems*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2008, vol. 5259, pp. 1081–1092.

[26] S. Islam, R. Davies, M. Bennamoun, and A. Mian, "Efficient detection and recognition of 3d ears," *International Journal of Computer Vision*, vol. 95, no. 1, pp. 52–73, 2011.

[27] S. Prakash and P. Gupta, "An efficient technique for ear detection in 3d: Invariant to rotation and scale," in *5th IAPR International Conference on Biometrics*, March 2012, pp. 97–102.

[28] ——, "Human recognition using 3d ear images," *Neurocomputing*, vol. 140, no. 0, pp. 317 – 325, 2014.

[29] L. Zhang, Z. Ding, H. Li, and Y. Shen, "3d ear identification based on sparse representation," *PLoS ONE*, vol. 9, no. 4, p. e95506, 04 2014.

[30] A. Mian, M. Bennamoun, and R. Owens, "Keypoint detection and local feature matching for textured 3d face recognition," *International Journal of Computer Vision*, vol. 79, no. 1, pp. 1–12, 2008.

[31] P. Verlinde and G. Chollet, "Comparing decision fusion paradigms using k-nn based classifiers, decision trees and logistic regression in a multi-modal identity verification application," in *2nd International conference on Audio- and Video-Based Biometric Person Authentication*, 1999, pp. 188–193.

[32] Y. Ma, B. Cukic, and H. Singh, "A classification approach to multi-biometric score fusion," in *Proceedings of the 5th international conference on Audio- and Video-Based Biometric Person Authentication.* Berlin, Heidelberg: Springer-Verlag, 2005, pp. 484–493.

[33] E. Henry, "Classification and uses of finger prints," 1990.

[34] N. Ratha, K. Karu, S. Chen, and A. Jain, "A real-time matching system for large fingerprint databases," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 799–813, 1996.

[35] J. Wu, A. Narasimhalu, B. Mehtre, C. Lam, and Y. Gao, "Core: a content-based retrieval engine for multimedia information systems," *Multimedia Systems*, vol. 3, no. 1, pp. 25–41, 1995.

[36] J.-K. Wu and A. Narasimhalu, "Identifying faces using multiple retrievals," *IEEE MultiMedia*, vol. 1, no. 2, pp. 27–38, 1994.

[37] A. Mhatre, S. Palla, S. Chikkerur, and V. Govindaraju, "Efficient search and retrieval in biometric databases," in *SPIE Defense and Security Symposium*, 2005, pp. 265–273.

[38] A. Mhatre, S. Chikkerur, and V. Govindaraju, "Indexing biometric databases using pyramid technique," in *Audio-and Video-Based Biometric Person Authentication.* Springer, 2005, pp. 841–849.

[39] U. Jayaraman, S. Prakash, Devdatt, and P. Gupta, "An indexing technique for biometric database," in *International Conference on Wavelet Analysis and Pattern Recognition.*, vol. 2, 2008, pp. 758–763.

[40] H. Chen and B. Bhanu, "Efficient recognition of highly similar 3d objects in range images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 1, pp. 172–179, 2009.

[41] U. Jayaraman, S. Prakash, and P. Gupta, "Indexing multimodal biometric databases using kd-tree with feature level fusion," in *Information Systems Security.* Springer, 2008, pp. 221–234.

[42] P. Gupta, A. Sana, H. Mehrotra, and C. J. Hwang, "An efficient indexing scheme for binary feature based biometric database," pp. 653 909–10, 2007.

[43] A. Bronstein, M. Bronstein, and R. Kimmel, "Three-dimensional face recognition," *International Journal of Computer Vision*, vol. 64, no. 1, pp. 5–30, 2005.

[44] K. Etemad and R. Chellappa, "Discriminant analysis for recognition of human face images," in *Audio- and Video-based Biometric Person Authentication*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 1997, vol. 1206, pp. 125–142.

[45] C. Lu and X. Tang, "Surpassing human-level face verification performance on lfw with gaussianface," *arXiv preprint arXiv:1404.3840*, 2014.

[46] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1701–1708.

[47] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[48] Y.-C. Chen, V. Patel, P. Phillips, and R. Chellappa, "Dictionary-based face recognition from video," in *ECCV*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, vol. 7577, pp. 766–779.

[49] K.-C. Lee, J. Ho, M.-H. Yang, and D. Kriegman, "Video-based face recognition using probabilistic appearance manifolds," in *In Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2003, pp. 313–320.

[50] U. Park and A. Jain, "3d model-based face recognition in video," in *Advances in Biometrics*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, vol. 4642, pp. 1085–1094.

[51] U. Park, A. Jain, and A. Ross, "Face recognition in video: Adaptive fusion of multiple matchers," in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2007, pp. 1–8.

[52] A. Abaza, A. Ross, C. Hebert, M. A. F. Harrison, and M. S. Nixon, "A survey on ear biometrics," *ACM Comput. Surv.*, vol. 45, no. 2, pp. 22:1–22:35, 2013.

[53] N. B. Boodoo and R. K. Subramanian, "Robust multi biometric recognition using face and ear images," *CoRR*, vol. 0912.0955, 2009.

[54] D. R. Kisku, J. K. Sing, and P. Gupta, "Multibiometrics belief fusion," *CoRR*, vol. 1002.2755, 2010.

[55] X. Pan, Y. Cao, X. Xu, Y. Lu, and Y. Zhao, "Ear and face based multimodal recognition based on kfda," in *International Conference on Audio, Language and Image Processing*, July 2008, pp. 965–969.

[56] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, 2011, pp. 689–696.

[57] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.

[58] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, June 2005, pp. 539–546 vol. 1.

[59] G. Goswami, R. Bhardwaj, R. Singh, and M. Vatsa, "Mdlface: Memorability augmented deep learning for video face recognition," in *IEEE International Joint Conference on Biometrics (IJCB)*, Sept 2014, pp. 1–7.

[60] D. Menotti, G. Chiachia, A. Pinto, W. Robson Schwartz, H. Pedrini, A. Xavier Falcao, and A. Rocha, "Deep representations for iris, face, and fingerprint spoofing detection," *Information Forensics and Security, IEEE Transactions on*, vol. 10, no. 4, pp. 864–879, April 2015.

[61] S. Gao, Y. Zhang, K. Jia, J. Lu, and Y. Zhang, "Single sample face recognition via learning deep supervised autoencoders," *Information Forensics and Security, IEEE Transactions on*, vol. 10, no. 10, pp. 2108–2118, Oct 2015.

[62] Z. Wang, Z. Miao, Q. M. Jonathan Wu, Y. Wan, and Z. Tang, "Low-resolution face recognition: a review," *The Visual Computer*, vol. 30, no. 4, pp. 359–386, 2014.

[63] W. W. Z. Wilman and P. C. Yuen, "Very low resolution face recognition problem," in *Biometrics: Theory Applications and Systems (BTAS), 2010 Fourth IEEE International Conference on*, Sept 2010, pp. 1–6.

[64] C. X. Ren, D. Q. Dai, and H. Yan, "Coupled kernel embedding for low-resolution face image recognition," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3770–3783, Aug 2012.

[65] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, Nov 2010.

[66] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, Feb 2016.

[67] S. Shekhar, V. M. Patel, and R. Chellappa, "Synthesis-based recognition of low resolution faces," in *Biometrics (IJCB), 2011 International Joint Conference on*, Oct 2011, pp. 1–6.

[68] J. Yu and B. Bhanu, "Super-resolution restoration of facial images in video," in *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 4, 2006, pp. 342–345.

[69] K. Jia and S. Gong, "Generalized face super-resolution," *IEEE Transactions on Image Processing*, vol. 17, no. 6, pp. 873–886, June 2008.

[70] S. Sarkar, P. J. Phillips, Z. Liu, I. R. Vega, P. Grother, and K. W. Bowyer, "The humanid gait challenge problem: data sets, performance, and analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 2, pp. 162–177, Feb 2005.

[71] J. Man and B. Bhanu, "Individual recognition using gait energy image," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 2, pp. 316–322, Feb 2006.

[72] W. Kusakunniran, Q. Wu, H. Li, and J. Zhang, "Multiple views gait recognition using view transformation model based on optimized gait energy image," in *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, Sept 2009, pp. 1058–1064.

[73] S. Zheng, J. Zhang, K. Huang, R. He, and T. Tan, "Robust view transformation model for gait recognition," in *2011 18th IEEE International Conference on Image Processing*, Sept 2011, pp. 2073–2076.

[74] T. K. M. Lee, M. Belkhatir, and S. Sanei, "A comprehensive review of past and present vision-based techniques for gait recognition," *Multimedia Tools and Applications*, vol. 72, no. 3, pp. 2833–2869, 2014.

[75] L. Lee and W. E. L. Grimson, "Gait analysis for recognition and classification," in *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*, May 2002, pp. 148–155.

[76] L. Wang, T. Tan, H. Ning, and W. Hu, "Silhouette analysis-based gait recognition for human identification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1505–1518, Dec 2003.

[77] J. W. Davis and A. F. Bobick, "The representation and recognition of human movement using temporal templates," in *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, Jun 1997, pp. 928–934.

[78] M. Goffredo, J. N. Carter, and M. S. Nixon, "Front-view gait recognition," in *Biometrics: Theory, Applications and Systems, 2008. BTAS 2008. 2nd IEEE International Conference on*, Sept 2008, pp. 1–6.

[79] H. Lu, J. Wang, and K. N. Plataniotis, *Advanced signal processing: Theory and implementation for sonar, radar, and non-invasive medical diagnostic systems (2nd ed.).* Boca Raton: CRC Press, 2009, ch. A Review on Face and Gait Recognition: System, Data and Algorithms, pp. 303–325.

[80] X. Zhou and B. Bhanu, "Feature fusion of face and gait for human recognition at a distance in video," in *18th International Conference on Pattern Recognition (ICPR'06)*, vol. 4, 2006, pp. 529–532.

[81] "Context-aware fusion: A case study on fusion of gait and face for human identification in video," *Pattern Recognition*, vol. 43, no. 10, pp. 3660 – 3673, 2010.

[82] S. Maity and M. Abdel-Mottaleb, "3d ear segmentation and classification through indexing," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 2, pp. 423–435, Feb 2015.

[83] Y. Yang and D. Ramanan, “Articulated pose estimation with flexible mixtures-of-parts,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1385–1392.

[84] J. Lei, J. Zhou, and M. Abdel-Mottaleb, “Gender classification using automatically detected and aligned 3d ear range data,” in *International Conference on Biometrics*, 2013, pp. 1–7.

[85] J. Lei, J. Zhou, M. Abdel-Mottaleb, and X. You, “Detection, localization and pose classification of ear in 3d face profile images,” in *IEEE International Conference on Image Processing*, 2013, pp. 4200–4204.

[86] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: Active contour models,” *International journal of computer vision*, vol. 1, no. 4, pp. 321–331, 1988.

[87] J. L. Xu, C.and Prince, “Gradient vector flow: A new external force for snakes,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1997, pp. 66–71.

[88] M. Burge and W. Burger, “Ear biometrics in computer vision,” in *15th International Conference on Pattern Recognition*, vol. 2, 2000, pp. 822–826 vol.2.

[89] M. Choras and R. S. Choras, “Geometrical algorithms of ear contour shape representation and feature extraction,” in *Sixth IEEE International Conference on Intelligent Systems Design and Applications.*, vol. 2, 2006, pp. 451–456.

[90] M. Choras, “Further developments in geometrical algorithms for ear biometrics,” in *Articulated Motion and Deformable Objects.* Springer, 2006, pp. 58–67.

[91] M. Sonka, V. Hlavac, R. Boyle *et al.*, “Image processing, analysis, and machine vision,” 1999.

[92] J. Flusser and T. Suk, “Pattern recognition by affine moment invariants,” *Pattern recognition*, vol. 26, no. 1, pp. 167–174, 1993.

[93] M. Hu, “Visual pattern recognition by moment invariants,” *IRE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, 1962.

[94] J. Žunić, K. Hirota, and P. L. Rosin, “A hu moment invariant as a shape circularity measure,” *Pattern Recognition*, vol. 43, no. 1, pp. 47–57, 2010.

[95] P. Rosin, “Measuring shape: ellipticity, rectangularity, and triangularity,” *Machine Vision and Applications*, vol. 14, no. 3, pp. 172–184, 2003.

[96] ——, “Measuring rectangularity,” *Machine Vision and Applications*, vol. 11, no. 4, pp. 191–196, 1999.

[97] H. Freeman and R. Shapira, "Determining the minimum-area encasing rectangle for an arbitrary closed curve," *Communications of the ACM*, vol. 18, no. 7, pp. 409–413, 1975.

[98] I. Jolliffe, *Principal component analysis.* Wiley Online Library, 2005.

[99] P. J. Besl and R. C. Jain, "Invariant surface characteristics for 3d object recognition in range images," *Computer vision, graphics, and image processing*, vol. 33, no. 1, pp. 33–80, 1986.

[100] S. Krishnamachari and M. Abdel-Mottaleb, "Hierarchical clustering algorithm for fast image retrieval," in *Proc. SPIE Conference on Storage and Retrieval for Image and Video databases VII*, 1999, pp. 427–435.

[101] M. Fonseca and J. A. Jorge, "Indexing high-dimensional data for content-based retrieval in large databases," in *Eighth International Conference on Database Systems for Advanced Applications.*, 2003, pp. 267–274.

[102] B. Ooi, K. Tan, C. Yu, and S. Bressan, "Indexing the edges a simple and yet efficient approach to high-dimensional indexing," in *Proceedings of the nineteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems.* ACM, 2000, pp. 166–174.

[103] D. Comer, "Ubiquitous b-tree," *ACM Computing Surveys (CSUR)*, vol. 11, no. 2, pp. 121–137, 1979.

[104] A. Guttman, "R-trees: A dynamic index structure for spatial searching," in *International Conference On Management Of Data.* ACM, 1984, pp. 47–57.

[105] S. Berchtold and D. Keim, "High-dimensional index structures database support for next decade's applications (tutorial)," in *ACM SIGMOD Record*, vol. 27, no. 2. ACM, 1998, p. 501.

[106] J. Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, 1975.

[107] N. Beckmann, H. Kriegel, R. Schneider, and B. Seeger, "The r*-tree: an efficient and robust access method for points and rectangles," in *International Conference On Management Of Data.* ACM, 1990, pp. 322–331.

[108] S. Berchtold, D. A. Keim, and H.-P. Kriegel, "The x-tree: An index structure for high-dimensional data," *Readings in multimedia computing and networking*, p. 451, 2001.

[109] S. Berchtold, C. Böhm, and H. Kriegal, "The pyramid-technique: towards breaking the curse of dimensionality," in *ACM SIGMOD Record*, vol. 27, no. 2. ACM, 1998, pp. 142–153.

[110] S. Battiato, D. Cantone, D. Catalano, G. Cincotti, and M. Hofri, “An efficient algorithm for the approximate median selection problem,” in *Algorithms and Complexity*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2000, vol. 1767, pp. 226–238.

[111] S. Maity, M. Abdel-Mottaleb, and S. A. Shihab, “Multimodal biometrics recognition from facial video via deep learning,” in *Second International Conference on Image and Signal Processing (ISPR -2016)*, vol. 7, Jan. 2017, pp. 67–75.

[112] ——, “Multimodal biometrics recognition from facial video via deep learning with missing modalities,” *Signal & Image Processing : An International Journal (SIPIJ)*, vol. 8, no. 1, Feb. 2017.

[113] G. Fahmy, A. El-sherbeeny, S. M, M. Abdel-mottaleb, and H. Ammar, “The effect of lighting direction/condition on the performance of face recognition algorithms,” in *SPIE Conference on Biometrics for Human Identification*, 2006, pp. 188–200.

[114] K. Lee, J. Ho, M. Yang, and D. Kriegman, “Visual tracking and recognition using probabilistic appearance manifolds,” *Computer Vision and Image Understanding*, 2005.

[115] P. Phillips, H. Moon, S. Rizvi, and P. Rauss, “The feret evaluation methodology for face-recognition algorithms,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090–1104, Oct 2000.

[116] K. Chang, K. Bowyer, S. Sarkar, and B. Victor, “Comparison and combination of ear and face images in appearance-based biometrics,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1160–1165, Sept 2003.

[117] C. Liu and H. Wechsler, “Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition,” *IEEE Transactions on Image Processing*, vol. 11, no. 4, pp. 467–476, Apr 2002.

[118] R. Khorsandi, S. Cadavid, and M. Abdel-Mottaleb, “Ear recognition via sparse representation and gabor filters,” in *IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems*, Sept 2012, pp. 278–282.

[119] S. Urolagin, K. V. Prema, and N. Subba Reddy, “Rotation invariant object recognition using gabor filters,” in *International Conference on Industrial and Information Systems*, July 2010, pp. 404–407.

[120] M. Yang and L. Zhang, “Gabor feature based sparse representation for face recognition with gabor occlusion dictionary,” in *Computer Vision ECCV 2010*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2010, vol. 6316, pp. 448–461.

[121] Y. Bengio, "Learning deep architectures for ai," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, Jan. 2009.

[122] D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, Eds., *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*. Cambridge, MA, USA: MIT Press, 1986.

[123] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contracting auto-encoders: Explicit invariance during feature extraction," in *In Proceedings of the Twenty-eight International Conference on Machine Learning (ICML11*, 2011.

[124] P. Lennie, "The cost of cortical computation," *Current Biology*, vol. 13, no. 6, pp. 493 – 497, 2003.

[125] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," 2008, pp. 1096–1103.

[126] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, Jul. 2006.

[127] Y. Bengio and Y. LeCun, "Scaling learning algorithms towards AI," in *Large Scale Kernel Machines*, L. Bottou, O. Chapelle, D. DeCoste, and J. Weston, Eds. MIT Press, 2007.

[128] M. Ranzato, C. S. Poultney, S. Chopra, and Y. LeCun, in *NIPS*, B. Schlkopf, J. C. Platt, and T. Hoffman, Eds.

[129] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Advances in Neural Information Processing Systems 19*. MIT Press, 2007, pp. 153–160.

[130] R. Lengelle and T. Denux, "Training mlps layer by layer using an objective function for internal representations," *Neural Netw.*, vol. 9, no. 1, pp. 83–97, Jan. 1996.

[131] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[132] P. Smolensky, "Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1," D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, Eds. Cambridge, MA, USA: MIT Press, 1986, ch. Information Processing in Dynamical Systems: Foundations of Harmony Theory, pp. 194–281.

[133] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed l^0 norm," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 289–301, Jan 2009.

[134] A. A. Ross, K. Nandakumar, and A. K. Jain, *Handbook of multibiometrics.* Springer, 2006.

[135] F. R. Hampel, E. M. Ronchetti, P. J. Rousseeuw, and W. A. Stahel, *Robust Statistics: The Approach Based on Influence Functions (Wiley Series in Probability and Statistics).* Wiley-Interscience, Apr. 2005.

[136] M. Aharon, M. Elad, and A. Bruckstein, "k -svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, Nov 2006.

[137] S. Maity, M. Abdel-Mottaleb, and S. A. Shihab, "Multimodal low resolution face and frontal gait recognition from surveillance video." *Transactions on Information Forensics & Security, IEEE*, Under Review.

[138] A. Papazoglou and V. Ferrari, "Fast object segmentation in unconstrained video," in *2013 IEEE International Conference on Computer Vision*, Dec 2013, pp. 1777–1784.

[139] T. Brox and J. Malik, *11th European Conference on Computer Vision.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, ch. Object Segmentation by Long Term Analysis of Point Trajectories, pp. 282–295.

[140] N. Sundaram, T. Brox, and K. Keutzer, "Dense point trajectories by gpu-accelerated large displacement optical flow," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2010-104, Jul 2010.

[141] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut -interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics (SIGGRAPH)*, August 2004.

[142] Y. J. Lee, J. Kim, and K. Grauman, "Key-segments for video object segmentation," in *Proceedings of the 2011 International Conference on Computer Vision*, ser. ICCV '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 1995–2002.

[143] M. P. Murray, A. B. Drought, and R. C. Kory, "Walking patterns of normal men," *The Journal of Bone & Joint Surgery*, vol. 46, no. 2, pp. 335–360, 1964.

[144] J. Perry and J. M. Burnfield, *Gait Analysis: Normal and Pathological Function.* Slack Incorporated, New Jersey, 2010.

[145] J. J. Little and J. E. Boyd, "Recognizing people by their gait: The shape of motion," 1998.

[146] J. L. Stephenson, S. J. D. Serres, and A. Lamontagne, "The effect of arm movements on the lower limb during gait after a stroke," *Gait & Posture*, vol. 31, no. 1, pp. 109–115, 2010.

[147] J. J. K. M. Jackson and S. J. Wyard., “The upper limbs during human walking. part 2: Function.” *Electromyogr Clin Neurophysiol.*, vol. 23, pp. 435–446, 1983.

[148] T. K. M. Lee, M. Belkhatir, P. A. Lee, and S. Sanei, “Fronto-normal gait incorporating accurate practical looming compensation,” in *19th International Conference on Pattern Recognition, 2008.*, Dec 2008, pp. 1–4.

[149] J. Yang, J. Wright, T. S. Huang, and Y. Ma, “Image super-resolution via sparse representation,” *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, Nov 2010.

[150] C. Dong, C. C. Loy, K. He, and X. Tang, “Image super-resolution using deep convolutional networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, Feb 2016.

[151] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, “Backpropagation applied to handwritten zip code recognition,” *Neural Computation*, vol. 1, no. 4, pp. 541–551, Dec 1989.

[152] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems*, 2012.

[153] H. Wang, S. Z. Li, Y. Wang, and J. Zhang, “Self quotient image for face recognition,” in *Image Processing, 2004. ICIP ’04. 2004 International Conference on*, vol. 2, Oct 2004, pp. 1397–1400 Vol.2.

[154] Z. Zhang, A. Ganesh, X. Liang, and Y. Ma, “Tilt: Transform invariant low-rank textures,” *International Journal of Computer Vision*, vol. 99, no. 1, pp. 1–24, 2012.

[155] B. J. Boom, L. J. Speeuwers, and R. N. J. Veldhuis., “Subspace-based holistic registration for low-resolution facial images,” *EURASIP Journal of Advances in Signal Processing*, 2010.

[156] H. B. Mitchell, *Image Fusion: Theories, Techniques and Applications.* Springer-Verlag Berlin Heidelberg, 2010.

[157] X. Xiang, L. Wanquan, and L. Ling, “Low resolution face recognition in surveillance systems,” *Journal of Computer and Communications*, vol. 2, no. 2, pp. 70–77, 2014.

[158] E. Cands, L. Demanet, D. Donoho, and L. Ying, “Fast discrete curvelet transforms,” 2005.

[159] T. Ojala, M. Pietikainen, and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, Jul 2002.

[160] T. Ahonen, A. Hadid, and M. Pietikainen, “Face description with local binary patterns: Application to face recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, Dec 2006.

[161] A. J. O’Toole, J. Harms, S. L. Snow, D. R. Hurst, M. R. Pappas, J. H. Ayyad, and H. Abdi, “A video database of moving faces and people,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 812–816, May 2005.