Master of Science in Computer Science and Engineering

Optical Flow Based Facial Expression Recognition from Video Sequences

by

Md Sirajus Salekin

Department of Computer Science and Engineering (CSE)
Islamic University of Technology (IUT)
A subsidiary organ of the Organization of Islamic Cooperation (OIC)
Gazipur-1704, Dhaka, Bangladesh

ISLAMIC UNIVERSITY OF TECHNOLOGY

Optical Flow Based Facial Expression Recognition from Video Sequences

Author:

Md Sirajus Salekin

Student ID: 134602

Supervisor:

Md. Hasanul Kabir, PhD

Associate Professor

Department of Computer Science and Engineering (CSE)

Islamic University of Technology (IUT)

A thesis submitted to the Department of Computer Science and Engineering (CSE) in partial fulfilment of the requirements for the degree of M.Sc. in CSE

Department of Computer Science and Engineering (CSE)

Islamic University of Technology (IUT)

A subsidiary organ of the Organization of Islamic Cooperation (OIC)

Gazipur-1704, Dhaka, Bangladesh

November 2015

Abstract

Facial expression is one of the most powerful masses of non-verbal communication through which we can easily enter the world of one's instant emotions or intuitions. As most of the time, it elicits naturally, so it brings out a lot of applications in the field of machine intelligence, behavioral science, clinical practices, biometric security, gaming, human computer interactions, psychological research, data-driven animation etc. Proper expression recognition can lead to an intelligent machine for taking commands more effectively, can show the insight psychological condition of a patient to a psychiatrist or researcher, can show the next suggested path for any computer game. But automatic facial expression recognition is a challenging task due to the different factors such as variations in illumination, pose, facial expression, alignment, different ages, occlusions etc. In this thesis paper, we propose a novel feature representation by a new feature descriptor, named Patterns of Oriented Motion Flow (POMF) from the optical flow information, to recognize the proper facial expression from a facial video. The POMF computes different directional motion information and encodes those directional flow information with enhanced local texture micro pattern. As it captures the spatial temporal changes of facial movements through optical flow and enables to observe both local and global structures, it shows its robustness for the facial expression. Finally, the POMF histogram is used to train the expression model by Hidden Markov Model (HMM). To train through the HMM, the objective sequences are produced by the generation of codebook using K-means clustering technique. The performance of the proposed method has been evaluated over the RGB camera based and Depth camera based video. We also compare the proposed method with the other promising appearance based methods. Experimental results demonstrate that the proposed POMF descriptor is more robust in extracting facial information and provides higher classification rate compared to other existing promising methods.

Acknowledgements

It is an auspicious moment for me to submit my Master's thesis work by which I am eventually going to end my Master's study. At the beginning, I want to express our heart-felt gratitude to Almighty Allah for his blessings to bestow upon me which made it possible to complete this thesis research successfully. Without the mercy of Allah, I wouldn't be where I am right now. All thanks and praises be to Allah.

Secondly, I would like to thank my thesis supervisor, Dr. Md. Hasanul Kabir, Associate Professor, CSE, IUT, for his support and guidance on this thesis. Dr. Kabir has been an instrumental to this work and my career. He taught me how to do research, think critically, be a graduate student, and teach effectively. His all-time guidance, encouragement and continuous observation made the whole matter as a successful one. Without his continuous support, this thesis would not see the path of proper itinerary of the research world.

It was my pleasure to get the cooperation and coordination from the Head of the Department, Professor Dr. M.A. Mottalib during various phases of the work. I am grateful to him for his constant and energetic guidance, constructive criticism and valuable advice. The faculty members of the CSE department of IUT helped make my working environment a pleasant one, by providing a helpful set of eyes and ears when problems arose.

I must thank to Dr. Zia Uddin, Assistant Professor, Department of Computer Education, College of Education Sungkyunkkwan University, South Korea, for his rigorous support by providing the depth database and giving valuable suggestions. I also wish to take an opportunity to articulate my sincerest gratitude and heartiest thanks to Dr. Md. Abu Raihan Mostofa Kamal, Associate Professor, CSE, IUT and Dr. Md. Kamrul Hasan, Associate Professor, CSE, IUT for their continuous support and encouragement to improve this thesis work.

I would like to thank the jury members of my thesis committee for the many interesting comments and criticism that helped improve this manuscript. Lastly, I am deeply grateful to my friends and family for their unconditional support. This work would have never been completed without the consistent support and encouragement from them throughout my Master's program.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we first present an overview of our thesis that includes the significance of the problem and the problem statement in detail. Besides, we also discuss about the different research challenges what we are going to face in the whole scenario. After that, we present our thesis objectives and contributions. The chapter ends with a short description of the organization of this thesis.

1.1 Overview

Facial expression provides non-verbal cues which are the representation of a person's emotions or intentions. We can easily capture anyone's behavior or reaction based on this natural indications. Facial expression recognition system attracts the researchers a lot in the last few decades because of its increasing demand in the field of automatic human computer interaction system. Basically, its natural identity makes it more applicable over the other biometrics.

An automatic facial expression recognition system refers to a computer system which tries to analyze and recognize the facial feature from the visual perspective. The fundamental components embedded in a facial expression recognition system are: image acquisition, pre-processing, feature extraction, classification, and post-processing, as shown in figure 1.1. Besides these steps, a key issue in successful facial expression analysis is to find an efficient feature extraction method, which will provide a robust facial feature representation for the classifier to train and test.

The facial expression recognition system can be applied over static image or video image. If we consider a static image, then the desired facial featured will be extracted only from that particular image. On the contrary, if we consider a video image then we will have to consider a sequence of the image frames and need to track our feature over all the

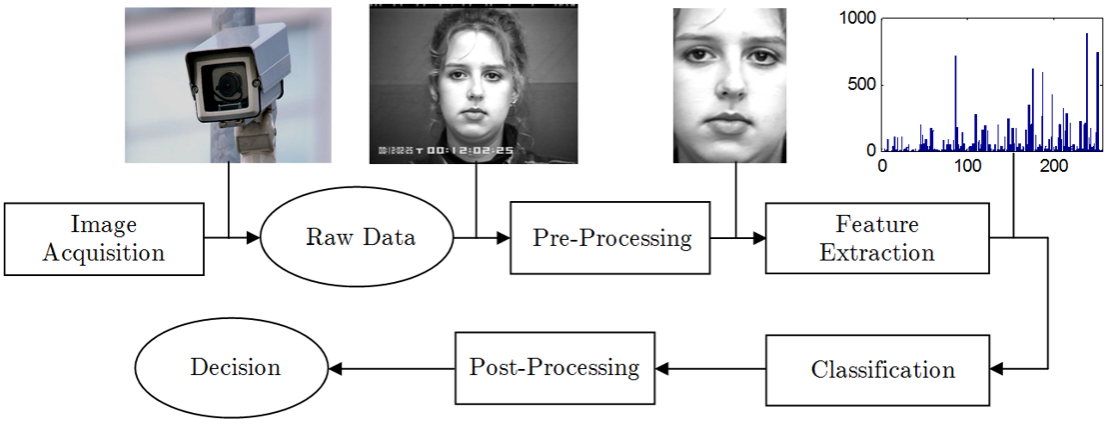

FIGURE 1.1: Components of a generic facial expression recognition system

FIGURE 1.2: Representation of different types of facial expression

frames for that particular video. So in order to get a viable facial expression recognition system we need to capture the valid facial parameters over the image or video. Figure 1.2 shows different types of facial expression.

1.2 Significance of the Problem

Facial expression recognition system addresses a lot of applications ranging from machine intelligence, biometric, human computer interaction, behavioral science, gaming,

interaction to psychological research, clinical practice, etc. Because of the availability of the both improved versions and inexpensive versions of camera, a widespread applications are appearing day by day in the different fields of computer interaction. Let's think about a home environment where if someone enters, based on his/her mode some home applicants will be activated. For instance, if he/she is in sad mood, cool music will be played on; if he/she is fear mode, more lights will be switched on etc. Let's say, we want to get the feedbacks of the clients in a photography exhibition. Whenever they are going through the photographs, we can analyze their facial expressions and get the glimpse of their feedbacks. Let's say, someone is trying to search something in the search engine. After examining his/her mode, the searched contents will be appeared. These are some recent trends of the promising applications of facial expression analysis.[Figure 1.3]

1.3 Research Challenges

Deriving an efficient, robust, and effective feature representation is a critical issue for any successful facial expression recognition system [1]. Although it receives considerable attention, the inherent variability of facial appearances makes recognition in unconstrained environment a difficult and challenging task. Human faces are non-rigid, dynamic objects with a large diversity in shape, color and texture, due to multiple factors such as head pose, lighting conditions (contrast, shadows), occlusions (glasses), gender, illness, aging, and other facial features (makeup, beard). Therefore, a key challenge is achieving optimal preprocessing, feature extraction, and classification, particularly under conditions of input data variability. For example, in the figure 1.4, from the first row, first, second and third images are the samples of happiness, but their appearances are totally different considering their gender, hair, facial view, skin color.

1.4 Thesis Objectives

Our research aimed at designing an effective appearance-based facial expression recognition system based on the optical flow information of the video sequences which will overcome different challenges of the facial expression recognition problems. The designed facial expression recognition system should satisfy the following criteria:

- Facial expression recognition should be done from the video sequences using image frames.
- It will address an efficient feature extraction method which will convey the image information from frame to frame.

FIGURE 1.3: Examples of some recent applications of facial expression

FIGURE 1.4: Different types of challenging situation of facial expression

- The proposed feature descriptor should be robust against different factors like variations in illumination, pose, alignment, occlusion, facial expression and aging etc.
- Proper training method should be used for the classification of the video sequences.
- The comparative performance should be evaluated against different existing promising methods using proper benchmark dataset.

1.5 Thesis Contributions

In this thesis work, we have proposed a Facial Expression Recognition system using the optical flow information from the video sequences, addressing a new feature descriptor. The main contributions of this thesis are summarized as follows:

- For increasing the machine intelligence, the image frames are extracted from depth video sequences instead of the conventional RGB video sequences.
- Optical flow is utilized to convey the facial expression information from frame to frame by introducing motion changes.
- We have proposed a novel facial descriptor named Patterns of Oriented Motion Flow (POMF).

- The robustness of the proposed feature descriptor is increased by incorporating both local and global information from the image frame.
- To train the feature, we need time sequential feature, which is produced by the codebook generation using K-means clustering technique.
- Proper training method for the classification of each expression is done using Hidden Markov Model (HMM).
- The recognition performance is evaluated with existing different promising methods with both RGB and Depth video.

1.6 Organization of the Thesis

The rest of the thesis will be organized as follows: in Chapter 2, we present the literature review of different types of approaches for facial expression recognition system and different machine learning techniques for modeling the FER system. In Chapter 3, we propose our proposed POMF descriptor and motivation behind the idea. There, we discuss about the overall idea of our proposed descriptor and step by step implementation process. In Chapter 4, experimental setup, experimental results and performance analysis of our proposed descriptor with various promising methods are discussed. Finally, in Chapter 5, we conclude our thesis contributions and shows the future scopes for further developing the proposed method.

Chapter 2

Literature Review

In this chapter, we first present a discussion on different geometric and appearance-based facial feature representation, which is followed by a review on different appearance-based methods. Finally, we end the literature review with the description of some pattern recognition methods used for the facial expression recognition system.

2.1 Facial Feature Representation

During the last two decades, many methods have been proposed for different face-related problems, where different facial feature extraction techniques have been introduced. Based on the types of features used, facial feature extraction approaches can be roughly divided into two different categories: geometric feature-based methods and appearance-based methods. Besides, facial expression recognition can be extracted from a static image or video image. So two types of method is proposed; one is individual frame based another is sequence based. Figure 2.1 shows the basic steps of a facial expression recognition system and their various types.

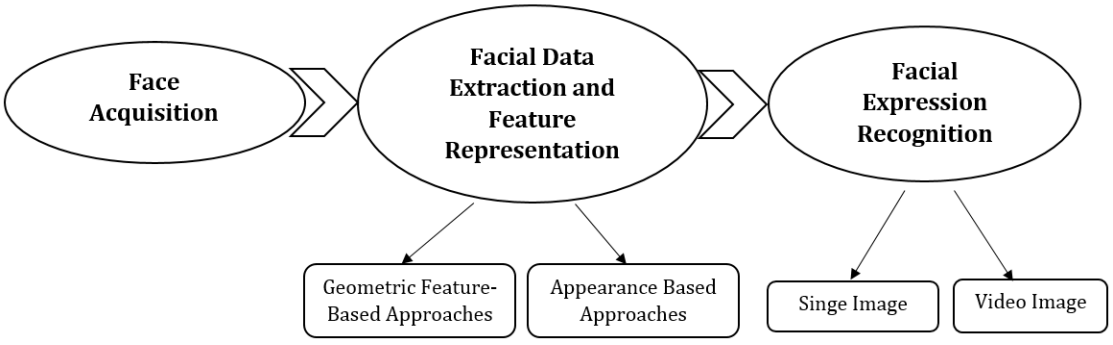

FIGURE 2.1: Basic steps of facial expression recognition system

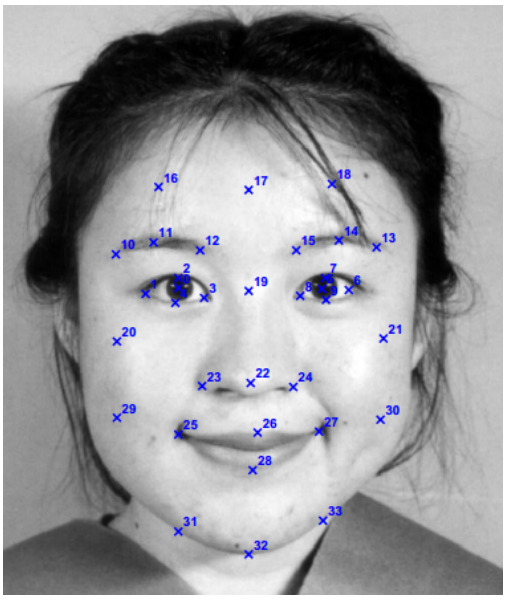

FIGURE 2.2: 34 fiducial points for face

2.1.1 Geometric Feature-Based Approaches

In geometric feature-based methods, the feature vector is formed based on the geometric relationships, such as positions, angles or distances between different facial components (eyes, ears, nose etc.). Earlier methods for facial recognition are mostly based on these geometric feature representations.

For facial expression recognition, facial action coding system (FACS) [2, 3] is a popular geometric feature-based method which represents facial expression with the help of a set of action units (AU). Each action unit represents the physical behavior of a specific facial muscle. Later, Zhang [4] proposed a feature extraction method based on the geometric positions of 34 manually selected fiducial points (Figure 2.2). A similar representation was adopted by Guo and Dyer [5], where they employed linear programming in order to perform simultaneous feature selection and classifier training. Recently, Valstar et al. [6, 7] have studied facial expression analysis based on tracked fiducial point data and reported that geometric features provide similar or better performance than appearance-based methods in action unit recognition (Figure 2.3).

Virtually, all of the existing vision systems for facial muscle action detection deal only with frontal-view face images and cannot handle temporal dynamics of facial actions. Maja Pantic et al. [8] present a system for automatic recognition of facial action units (AUs) and their temporal models from long, profile-view face image sequences (Figure 2.4). They exploit particle filtering to track 15 facial points in an input face-profile sequence, and introduce facial-action-dynamics recognition from continuous video input using temporal rules. The algorithm performs both automatic segmentation of an input

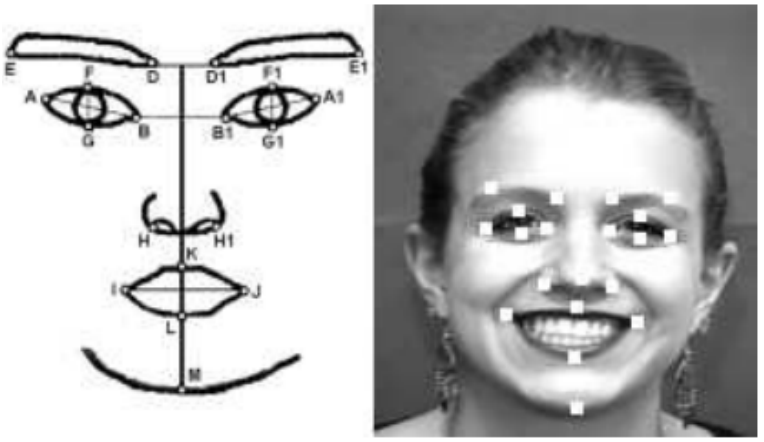

FIGURE 2.3: Fiducial facial points which will be tracked

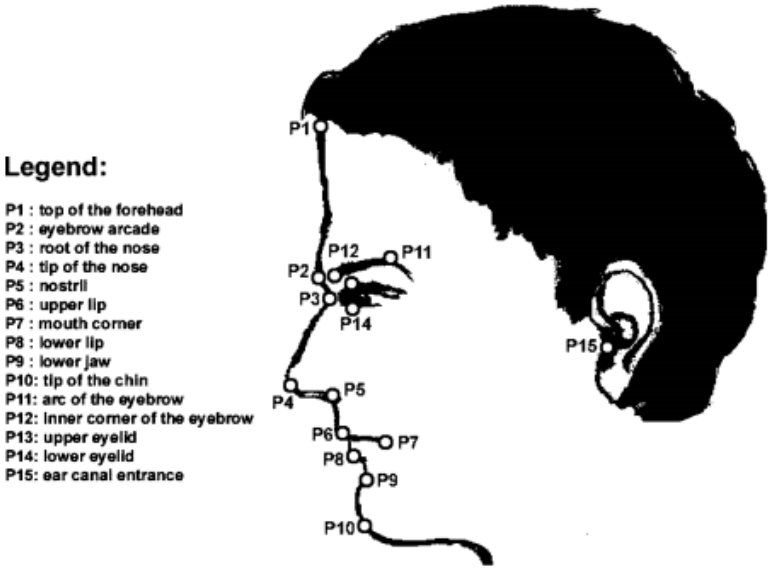

FIGURE 2.4: Facial points of the face components

video into facial expressions pictured and recognition of temporal segments (i.e., onset, apex, offset) of 27 AUs occurring alone or in a combination form in the input face-profile video.

On the other hand, some grid tracking methods were also proposed based on the geometric models. Irene Kotista et al. [9] proposed a grid-tracking and deformation system, based on deformable models, tracking the grid in consecutive video frames over time, as the facial expression evolves, until the frame that corresponds to the greatest facial expression intensity (Figure 2.5). User will select the grid point at first frame and then it will be tracked over the other frame. The geometrical displacement of certain selected Candide nodes, defined as the difference of the node coordinates between the first and the greatest facial expression intensity frame, is used as an input to a novel multi-class

FIGURE 2.5: Grid tracking

Support Vector Machine (SVM) system of classifiers that are used to recognize either the six basic facial expressions or a set of chosen Facial Action Units (FAUs).

However, the effectiveness of geometric methods is heavily dependent on the accurate detection of facial components, which is a difficult task in changing and unconstrained environment, thus making geometric methods difficult to accommodate in many scenarios.

2.1.2 Appearance-Based Approaches

Appearance-based methods extract the facial appearance by applying image filter or filter bank on the whole face image or some specific facial regions. Basically, we can observe two types of approaches in appearance-based methods. One type of approach tries to apply some feature reduction or class separation methods directly on the intensity values to minimize the feature size. Another type of approach uses any descriptor on the image intensity values and generate some key features from the image.

In case of feature minimization or class separation approaches Principal component analysis (PCA) [1, 10, 11], Linear Discriminant Analysis (LDA) [12], independent component analysis (ICA) [13, 14], Gabor wavelets [15] are the commonly-used appearance-based methods for facial expression recognition. Among these techniques, PCA is a global feature extraction method, where the whole facial image is taken into account during feature vector generation. This method provides an optimal linear transformation from the original image space to an orthogonal eigenspace with reduced dimensionality in the sense of least mean squared reconstruction error. In [16], the authors applied Fisher Linear Discriminant Analysis (FLDA) to classify further the principal component (PC) features of the facial expression images (Figure 2.6). Basically, FLDA is based on the class information that projects the data onto a subspace with the criterion that maximizes the between-class scatter and minimizes the within-class scatter of the projected data. On the other hand, ICA and Gabor wavelets extract the local features of an image, therefore called local feature descriptors. ICA is a better choice for FER than PCA. Basically, ICA is the generalization of PCA that seeks the independences of the

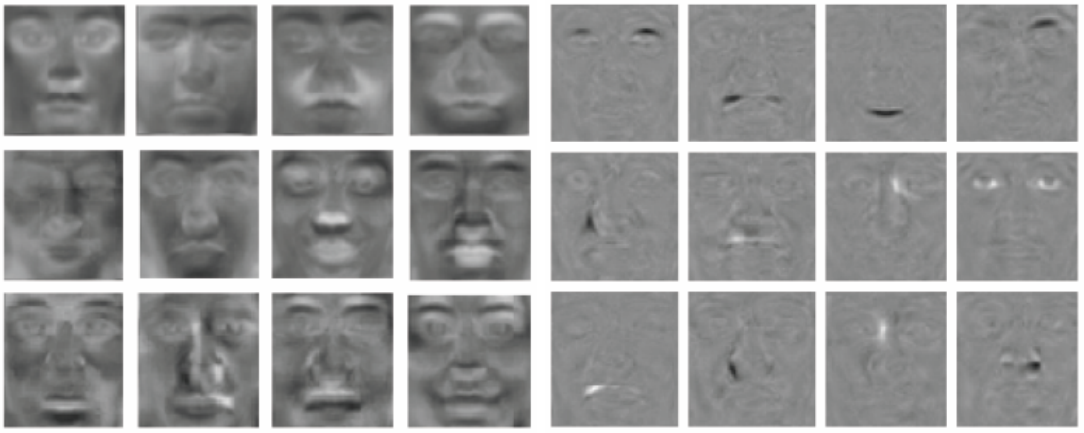

FIGURE 2.6: Facial feature representation based on PCA (left) and ICA (right) for expression recognition

image features. It performs blind source separation with the assumption that the input data is the linear mixture of the sources and then reduces the statistical dependencies of data to produce statistically independent bases and coefficients.

On the other hand, key feature generation type approaches apply any descriptor to the image intensity values. These type of approaches try to generate some fruitful information from the neighborhood region of an image and generate the key features. LBP [17], LDP [18], POEM [19] are some popular descriptors for the feature extraction in case of facial expression recognition system.

2.2 Feature Descriptors in Appearance-Based Approaches

In this section, we present a review on some facial feature representation techniques from some appearance-based approaches.

2.2.1 Enhanced ICA

Previously, we have already seen the widely use of PCA, LDA and ICA feature extraction techniques. In [16], Zia et al. present a new method to recognize several facial expressions from time sequential facial expression images. To produce robust facial expression features, Enhanced Independent Component Analysis (EICA) is utilized to extract locally independent component (IC) features which are further classified by Fisher Linear Discriminant Analysis (FLDA) (Figure 2.7). So their proposed EICA is consists of three fundamental stages: firstly, PCA is performed first for dimension reduction, then, ICA is applied on the reduced PCA subspace to find statistically independent basis images, and finally, FLDA is employed to compress the same classes as close as possible and to separate the different classes as far as possible.

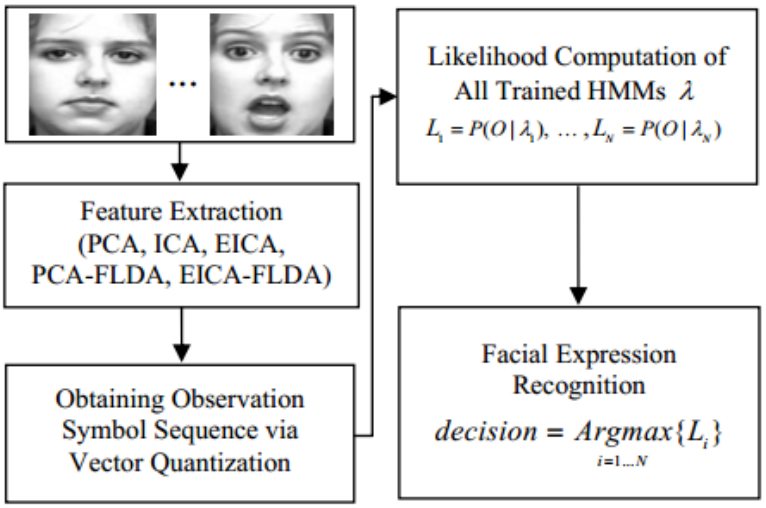

FIGURE 2.7: Overview of the EICA FER system

Their proposed FER system consists of preprocessing of sequential facial expression images in video, feature extraction via EICA-FLDA, codebook generation via vector quantization algorithm, and modeling and recognition via HMM. Figure 2.7 shows the overall architecture of our proposed FER system where λ denotes HMMs and L the likelihoods of HMMs.

2.2.2 Local Binary Pattern (LBP)

Local binary pattern (LBP) is a gray-scale and rotation invariant texture primitive that describes the spatial structure of the local texture of an image. Among the appearance-based feature extraction methods, Ojala et al. [17] the local binary pattern (LBP) method which was originally introduced for the purpose of texture analysis and its variants were used as a feature descriptor for facial expression representation [20], [21], [22], but it has also been robustly used on face recognition [23]. The LBP method is computationally efficient and robust to monotonic illumination changes. However, it is sensitive to non-monotonic illumination variation and also shows poor performance in the presence of random noise.

Local Binary Pattern (LBP), a gray-scale invariant texture pattern has gained much popularity among the researchers for encoding the spatial information of image texture. The basic LBP [17] was developed based on the presumption that image texture will be represented by two aspects, a pattern, and its strength. It encodes the gray-scale structure of an image using a binary code. It generates a label to each pixel of an image by thresholding its neighbor values with the center value. The resulting pattern represents a binary number which is converted to a decimal number before assigning to

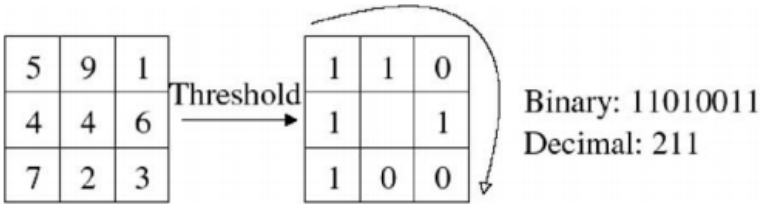

FIGURE 2.8: Illustration of LBP encoding

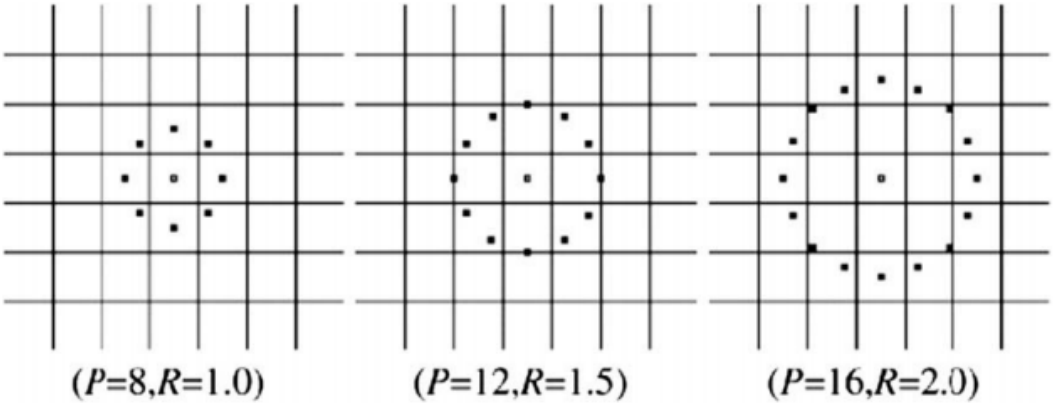

FIGURE 2.9: Three examples of the extended LBP; the circular (8,1) neighborhood, the circular(12,1.5) neighborhood, and the circular (16,2) neighborhood, respectively (from left to right)

each pixel.

$$LBP_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(g_p - g_c)2^p \tag{2.1}$$

$$s(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{2.2}$$

Here, g_c denotes the gray value of the center pixel (x_c, y_c) and g_p corresponds to the gray values of equally spaced pixels P on the circumference of a circle with radius R.

This encoded pattern ensures the rotation invariance of the gray-scale structure of the image. For further improvement of rotation invariance and finer quantization of the angular space, a variation of LBP was also proposed which is known as Uniform LBP. As in the significant image area, a certain local binary pattern appears frequently, they contain very few transitions from 0 to 1 and 1 to 0 in a circular bit sequence.

Ojala et al. [17] observed that LBP patterns with $U \leq 2$ are the fundamental properties of texture, which provide a vast majority of all the 8-bit binary patterns present in any texture image. Therefore, uniform patterns are able to describe significant local texture information, such as bright spot, flat area or dark spot, and edges of varying positive and

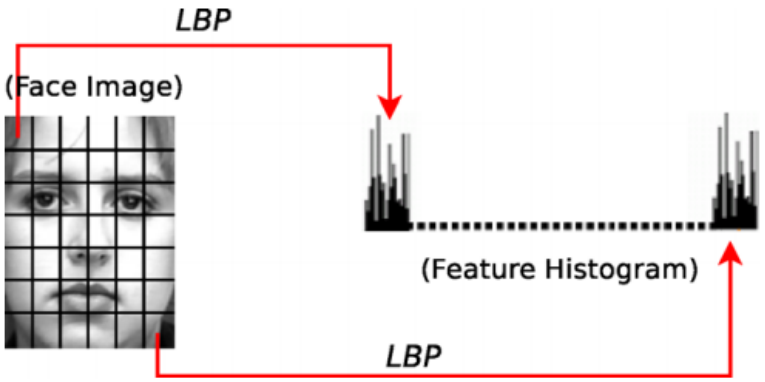

FIGURE 2.10: LBP encoding for facial expression recognition system

negative curvature. All the other patterns ($U > 2$) are grouped under a miscellaneous label.

In the facial expression recognition, at first, a face image is divided into some non-overlapping region from which LBP histograms are extracted and concatenated into a single, spatially enhanced feature histogram (Figure 2.10).

2.2.3 Local Directional Pattern (LDP)

As LBP is highly sensitive to noise and non-monotonic illumination changes, so a new local micro pattern is introduced. Jabid et al. [18] first proposed the local directional pattern (LDP) to represent local features. As with LBP, LDP features have a tolerance to illumination variation; however, they represent much more robust features than LBP as they consider gradient information. Thus, LDP can be considered a robust approach to face recognition system [18], facial expression recognition system [24], object recognition [25]. Recently, Zia [26] proposed LDP-PCA for the better recognition system in the facial expression recognition from video sequences.

LDP [18, 24, 25] is a gray-scale texture pattern which characterizes the spatial structure of a local image texture. An LDP operator computes the edge response values in all eight directions at each pixel position and generates a code from the relative strength magnitude. Since the edge responses are more illumination and noise insensitive than intensity values, the resultant LDP feature describes the local primitives including different types of curves, corners, and junctions, more stable and retains more information. Given a central pixel in the image, the eight directional edge response values $m_i, i = 0, 1, ..., 7$ are computed by Kirsch masks M_i in eight different orientations centered on its position, as shown in figure 2.11.

Presence of edge or corner will cause high edge-response values in their respective directions. Likewise, uniform or smooth regions will provide edge response values of same

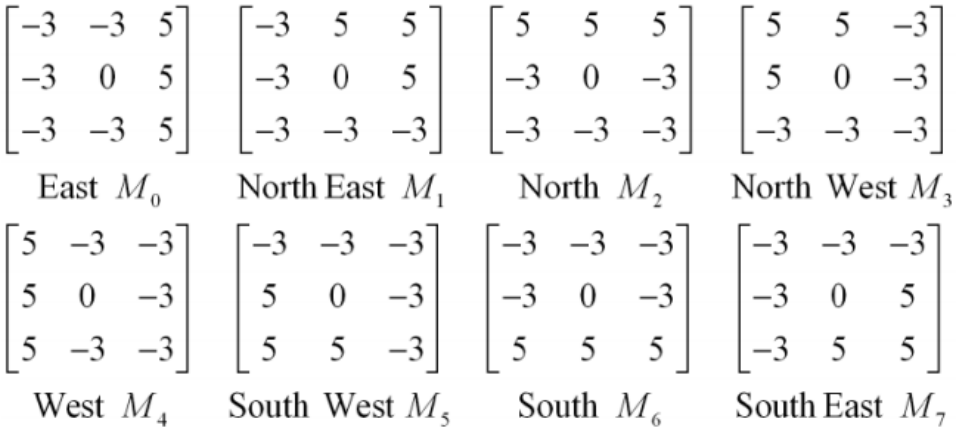

FIGURE 2.11: Kirsch edge response masks in all eight directions

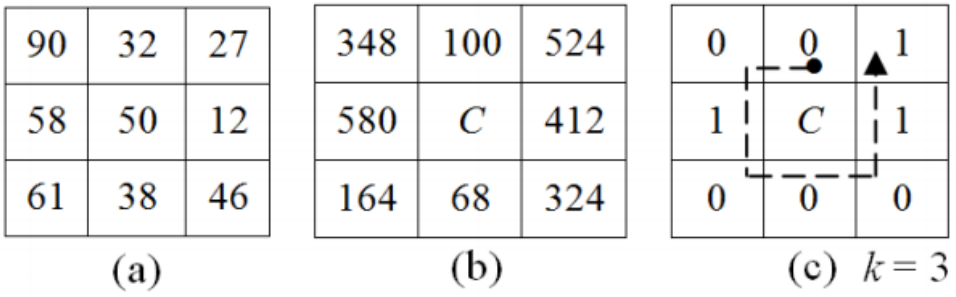

FIGURE 2.12: Illustration of the LDP encoding process; (a) original image, (b) magnitude of eight directional edge responses, (c) LDP binary code = 00100011 for center C

or similar magnitudes in different directions. Therefore, The LDP operator sets the most prominent k directions to 1 and others to 0 in order to obtain an 8-bit binary pattern based on the relative strength of the edge response values in different directions. Formally, the LDP code is derived by

$$LDP_k = \sum_{i=0}^{7} b_i(m_i - m_k)2^i \tag{2.3}$$

$$b_i(a) = \begin{cases} 1 & a \geq 0 \\ 0 & a < 0 \end{cases} \tag{2.4}$$

Here, m_k is the magnitude of the k^{th} most significant directional response. Since the edge responses are less sensitive to illumination and noise than intensity values, the resultant LDP feature retains more information and characterizes the texture primitives in a more robust manner. The LDP encoding process is illustrated in figure 2.12.

2.2.4 Patters of Oriented Edge Magnitude (POEM)

Although most of the feature selection was based on intensity value, but Vu et al. [19] first proposed a robust feature selection based on gradient value of the image. They proposed a descriptor named Patterns of Oriented Edge Magnitudes (POEM) which has desirable properties: POEM (1) is an oriented, spatial multi-resolution descriptor capturing rich information about the original image; (2) is a multi-scale self-similarity based structure that results in robustness to exterior variations; and (3) is of low complexity and is therefore practical for real-time applications. They first applied this robust descriptor on the face recognition system [19], [27], [28], [29], [30]. Later on, because of its robustness it has been used for the facial expression recognition system [31]. The POEM feature extraction consists of three steps:

1. **Gradient computation and orientation quantization:** first the gradient image is computed, then orientation of each pixel is discretized over $0 - \pi$(unsigned representation) or $0 - 2\pi$ (signed representation) (in the original work they used unsigned representation).
2. **Magnitude accumulation:** a local histogram of orientations over all pixels within a local image patch (cell) is calculated to incorporate information from neighboring pixels.
3. **Self-similarity calculation:** the accumulated magnitudes are encoded across different directions using the self-similarity LBP-based operator within a larger patch (block).

The final POEM descriptor at each pixel is the concatenation of all unidirectional POEMs at different orientations.
Firstly, at the pixel position p, a POEM feature is calculated for each discretized direction θ_i

$$POEM^{\theta_i}_{L,w,n}(p) = \sum_{j=1}^{n} f(S(I^{\theta_i}_p, I^{\theta_i}_{c_j}))2^j \tag{2.5}$$

$$f(x) = \begin{cases} 1 & x \geq t \\ 0 & x < t \end{cases} \tag{2.6}$$

Where I_p, I_{c_j} are the accumulated gradient magnitudes of central and surrounding pixels p, c_j respectively; $S(.,.)$ is the similarity function (e.g. the difference of two gradient magnitudes); L, w refer to the size of blocks and cells, respectively; n, set to 8 by default in this paper, is number of pixels surrounding the considered pixel p; and f is defined where the value t is slightly larger than zero to provide some stability in uniform regions. The final POEM feature set at each pixel is the concatenation of these unidirectional

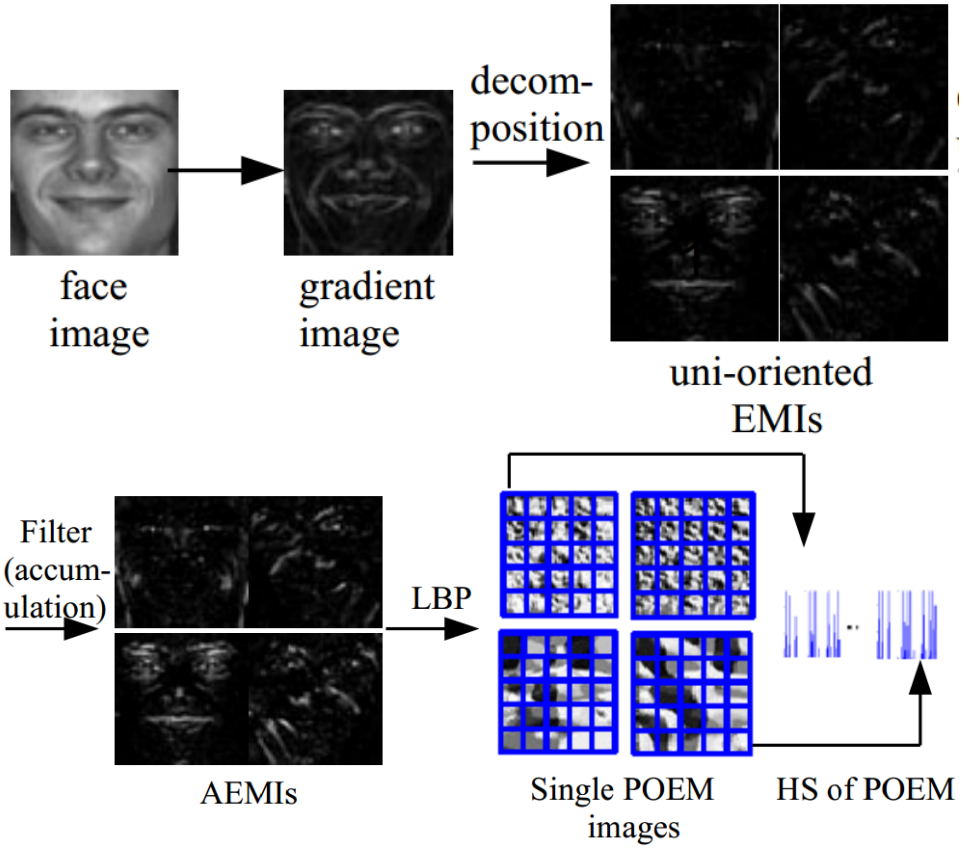

FIGURE 2.13: Fundamental steps of POEM facial feature extraction

POEMs at each of our m orientations:

$$POEM^{\theta_i}_{L,w,n}(p) = \{POEM^{\theta_1}, ..., POEM^{\theta_m}\} \tag{2.7}$$

In the figure 2.13 step by step POEM feature extraction procedure is shown. Finally, concatenated histogram of POEM will be the feature vector of the image.

2.2.5 Optical Flow Based Methods

Optical flow [32], [33], [34], [35] is the technique of independent motion estimation at each pixel of an image sequence. The aim of this optical flow is to calculate an approximation of image velocity or image changes from frame to frame which can be applied for a wide verity of tasks like image segmentation, image registration, object tracking etc.

Optical flow contains prominent information in case of facial expression. As facial expression starts from a neutral stage and goes through continuous changes and again ends with another neutral stage, so it contains the expression changing behaviors which can easily be captured by optical flow information.

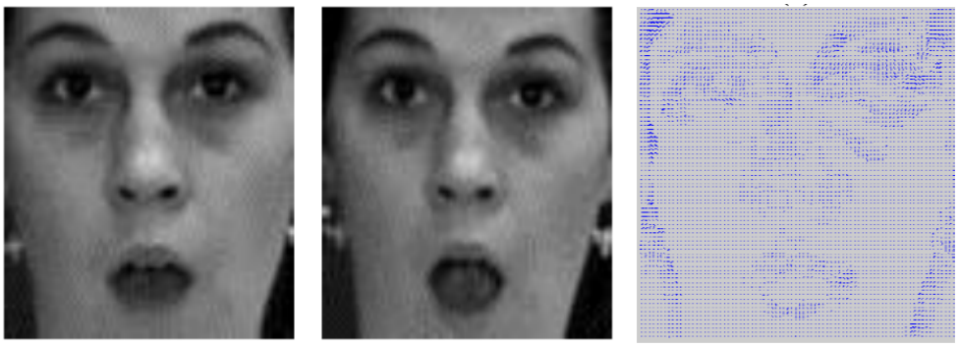

FIGURE 2.14: Two consecutive surprise images and its corresponding optical flows

It is not for the first time that optical flow is being used for facial expression recognition system. Mase [36] used optical flow (OF) to recognize facial expressions. He was one of the first to use image processing techniques to recognize facial expressions. Lanitis et al. [37] used a flexible shape and appearance model for image coding, person identification, pose recovery, gender recognition, and facial expression recognition. Black and Yacoob [38] used local parameterized models of image motion to recover non-rigid motion. Once recovered, these parameters were used as inputs to a rule-based classifier to recognize the six basic facial expressions. Yacoob and Davis [39] computed optical flow and used similar rules to classify the six facial expressions. Irfan et al. [40] described a computer vision system for observing facial motion by using an optimal estimation optical flow method coupled with geometric, physical and motion-based dynamic models describing the facial structure.

Recently, Zia et al. [41] used the optical flow information for facial expression recognition which is further enhanced by Principal Component Analysis (PCA) and Generalized Discriminant Analysis (GDA). Using these features, discrete Hidden Markov Models (HMMs) are applied to model the facial expression system. But to train by HMM, objective sequences are needed which was produced by the codebook generation using Linde, Buzo and Gray (LBG) algorithm [42]. Figure 2.14 shows the optical flows of two consecutive images of a surprise expression.

2.3 Machine Learning Techniques

After having the proper features from facial expression, we need a proper training method to model the FER system. There are a number of machine learning approaches such as Hidden Markov Model [26], Support Vector Machine (SVM) [24], Neural Network [43], K-Nearest Neighbor classifier [44] etc. But among these, when we use time sequential trainer like HMM, we need at first objective sequences which are produced from the feature vectors through codebook generation.

2.3.1 Codebook Generation

In pattern recognition applications, codebook generation is performed by the Vector Quantization technique. Vector quantization (VQ) is a classical quantization technique from signal processing that allows the modeling of probability density functions by the distribution of prototype vectors. It was originally used for data compression. It works by dividing a large set of points (vectors) into groups having approximately the same number of points closest to them. This vector quantization is also used as clustering methods. The main advantage of VQ in pattern recognition is its low computational burden when used with other techniques such Hidden Markov Model (HMM). K-means [45] and LBG [42] algorithms are two popular vector quantization techniques which are discussed below.

2.3.1.1 K-means Clustering

The K-means [45] algorithm takes the input parameter, k and partitions a set of n objects into k clusters so that the resulting intra-cluster similarity is high but the inter-cluster similarity is low. Cluster similarity is measured in regard to the mean value of the objects in a cluster, which can be viewed as the cluster's centroid or center of gravity. These centroids should be placed in a cunning way because of different location causes different results. So, the better choice is to place them as much as possible far away from each other.

The k-means algorithm proceeds as follows. First, it randomly selects k of the objects, each of which initially represents a cluster mean or center. For each of the remaining objects, an object is assigned to the cluster to which it is the most similar, based on the distance between the object and the cluster mean. It then computes the new mean for each cluster. This process iterates until the criterion function converges. Typically, the square-error criterion is used, defined as

$$E = \sum_{i=1}^{k} \sum_{p \exists c_i} |p - m_i|^2 \tag{2.8}$$

where E is the sum of the square error for all objects in the data set; p is the point in space representing a given object and m_i is the mean of cluster C_i (both p and m_i are multidimensional). In other words, for each object in each cluster, the distance from the object to its cluster center is squared, and the distances are summed. This criterion tries to make the resulting k clusters as compact and as separate as possible.

Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the

global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster centers. The k-means algorithm can be run multiple times to reduce this effect.

2.3.1.2 LBG Clustering

The Linde-Buzo-Gray algorithm (LBG) [42] is a vector quantization algorithm to derive a good codebook. The algorithm is like a K-means algorithm which takes a set of input vectors $S = \{x_i \epsilon R^d | i = 1, 2, ..., n\}$ as input and generates a representative subset of vectors $C = \{c_j \epsilon R^d | j = 1, 2, ..., K\}$ with a user specified $K << n$ as output according to the similarity measure. The algorithm is summarized below.

LBG Algorithm:

1. input training vectors $S = \{x_j \epsilon R^d | i = 1, 2, ..., n\}$.
2. initiate a codebook $C = \{c_j \epsilon R^d | j = 1, 2, ..., K\}$.
3. set $D_o = 0$ and let $k = 0$.
4. classify the n training vectors into K clusters according to $x_i \epsilon S_q$ if $||x_i - c_q||_p \leq ||xi - c_j||_p$ for $j \neq q$.
5. update cluster centers $c_j, j = 1, 2, ..., K$ by $c_j = \frac{1}{|S_j|} \sum_{x_i \epsilon S_j} x_i$.
6. set $k \leftarrow k + 1$ and compute the distortion $D_k = \sum_{j=1}^{K} \sum_{x_i \epsilon S_j} ||x_i - c_j||_p$.
7. $(D_{k-1} - D_k)/D_k > \epsilon$ (a small number), repeat steps $4 \sim 6$.
8. output the codebook $C = \{c_j \epsilon R^d | j = 1, 2, ..., K\}$,

The convergence of LBG algorithm depends on the initial codebook C, the distortion D_k, and the threshold ϵ. In implementation, we need to provide a maximum number of iterations to guarantee the convergence.

2.3.2 Classifiers

Classifiers are the mechine learning techniques which will finally generate our desired model of facial expression and recognize the new samples of expresion. Some of the most commonly used classifiers for facial expression recognition are SVM [46], Neural network [47], HMM [48, 49].

2.3.2.1 SVM

Support Vector Machine(SVM) [46], are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. Given a set of training examples, each marked for belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. It performs the classification by constructing a hyper plane in such a way that the separating margin between positive and negative examples is optimal. This separating hyper plane then works as the decision surface.

Given a set of labeled training samples $T = \{(x_i, l_i), i = 1, 2, ..., L\}$, where $x_i \epsilon R^P$ and $l_i \epsilon \{-1, 1\}$, a new test data x is classified by

$$f(x) = sign(\sum_{i=1}^{L} \alpha_i l_i K(x_i, x) + b) \tag{2.9}$$

Here, α_i are Lagrange multipliers of dual optimization problem, b is a threshold parameter, and K is a kernel function. The hyper plane maximizes the separating margin with respect to the training samples with $\alpha_i > 0$, which are called the support vectors.

SVM makes binary decisions. To achieve multi-class classification, the common approach is to adopt the one-against-rest or several two-class problems. Basically, an SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on.

2.3.2.2 Neural Network

In machine learning and cognitive science, artificial neural networks (ANNs) [47] are a family of models inspired by biological neural networks (the central nervous systems of animals, in particular, the brain). Artificial neural networks are generally presented as systems of interconnected "neurons" which exchange messages between each other. The connections have numeric weights that can be tuned based on experience, making neural nets adaptive to inputs and capable of learning.

Neural networks are consists of nodes or units which are connected by some links. For example, a link from unit j to unit i is represented by the activation a_j from j to i. Besides, each link also has a numeric weight $W_{j,i}$ which determines the strength and

sign of the connection. Each unit i first computes a weighted sum of its inputs:

$$in_i = \sum_{j=0}^{n} W_{j,i} a_j, \tag{2.10}$$

Then it applies an activation function g to this sum to derive the output:

$$a_i = g(in_i) = g(\sum_{j=0}^{n} W_{j,i} a_j). \tag{2.11}$$

The activation function g is designed to fulfill to goals. First, we want the unit to be "active" (near +1) when the "right" inputs are given, and "inactive" (near 0) when the "wrong" inputs are given. Second, the activation needs to be nonlinear, otherwise, the entire neural network collapses into a simple linear function.

In supervised learning, for a given set of example pairs $(x, y), x\epsilon X, y\epsilon Y$ and the aim is to find a function $f : X \rightarrow Y$ in the allowed class of functions that matches the examples. In other words, the desired goal is to infer the mapping implied by the data; the cost function is related to the mismatch between our mapping and the data and it implicitly contains prior knowledge about the problem domain. A commonly used cost is the mean-squared error, which tries to minimize the average squared error between the network's output, $f(x)$, and the target value y over all the example pairs. When one tries to minimize this cost using gradient descent for the class of neural networks called multilayer perceptrons, one obtains the common and well-known back-propagation algorithm for training neural networks.

2.3.2.3 Hidden Markov Model (HMM)

Hidden Markov Model (HMM) [48, 49] is a statistical Markov model where the system is developed based on Markov process with some unobserved (hidden) states. It is known for its wide application in temporal pattern recognition especially in the field of machine learning and data mining. The basic theory of HMM was developed by Baum et al. [48] and it has been applied extensively to solve a large number of problems. Due to its successful usage in pattern classification to decode the time-sequential events, HMM has been adopted as a model and recognizer for expression recognition. Besides, different enhance versions [50] of HMM have been used widely in the field of speech recognition, gesture recognition, human activity recognition, facial expression recognition, language modeling, motion analysis and tracking etc.

A basic HMM is represented by a set of parameters $\lambda = \{\pi, A, B\}$ where, π = prior probabilities of the states, A = transition probabilities of one state to another, B = observation symbol probability matrix. The main purpose of HMM is to find out the

model type (λ_k) with the highest probability of the likelihood $P(O|\lambda_k)$ for the observation sequence O. If we denote the states in the model by $S = \{s_1, s_2, ..., s_N\}$ and each state at given time t by $Q = \{q_1, q_2, ..., q_t\}$, then the HMM parameters can be presented as follows.

$$A = \{a_{ij}\}, a_{ij} = P(q_{t+1} = S_j | q_t = S_i), where\, 1 \leq i, j \leq N \tag{2.12}$$

$$B = \{b_j(O_t)\}, b_j = P(O_t | q_t = S_j), where\, 1 \leq j \leq N \tag{2.13}$$

$$\pi = \{\pi_j)\}, \pi_j = P(q_1 = S_j) \tag{2.14}$$

To train each HMM, first vector quantization is performed on the training features from the facial expression image sequences to obtain discrete symbols. Those obtained sequential symbols are then trained with HMMs to learn the proper model for each expression. On the other hand, to test a sample video of facial expressions, at first proper feature vectors are extracted by POMF feature descriptor using the same procedure. Then each of the trained models (λ_k) is used to generate the likelihood response for the particular sample observation sequence (O). Finally, to determine the test observation sequence, highest likelihood response from all N-trained expression HMMs evokes the corresponding desired class of the facial expression as follows:

$$\text{Detected Expression} = \arg \max_{k=1}^{N} (P(O|\lambda_k)) \tag{2.15}$$

Chapter 3

Proposed Method

In this chapter, we present the overall framework of our proposed facial expression recognition system and explain the proposed Patterns of Oriented Motion Flow (POMF) descriptor introducing the flow energy estimation process and micro pattern coding. After that, we present how we can incorporate the POMF descriptor to generate the model of facial expression system.

3.1 Framework of the Proposed FER

Our proposed method starts with the optical flow information from image frames. For facial video, depth camera-based video is preferred. Then from the optical flow information, the proposed POMF feature will be generated and those will be trained by HMM. Finally, from the maximum likelihood response of HMM, the desired expression will be recognized. Figure 3.1 shows the general steps of the proposed Facial Expression Recognition (FER) system.

3.2 Optical Flow Estimation

Optical flow features have been used increasingly over the past decade in the field of any motion detection or object tracking. As it defines the image changes from frame to frame nowadays it is being used for facial expression recognition from video [36], [38], [39], [40] and already it has exposed its robustness. From the video image of expression, our first step is to calculate the motion change from frame to frame. And it is done by the estimation of optical flow.

The optical flow methods try to calculate the motion between two image frames which are taken at times t and $t + \Delta t$ at every voxel position. For a $2D + t$ dimensional case

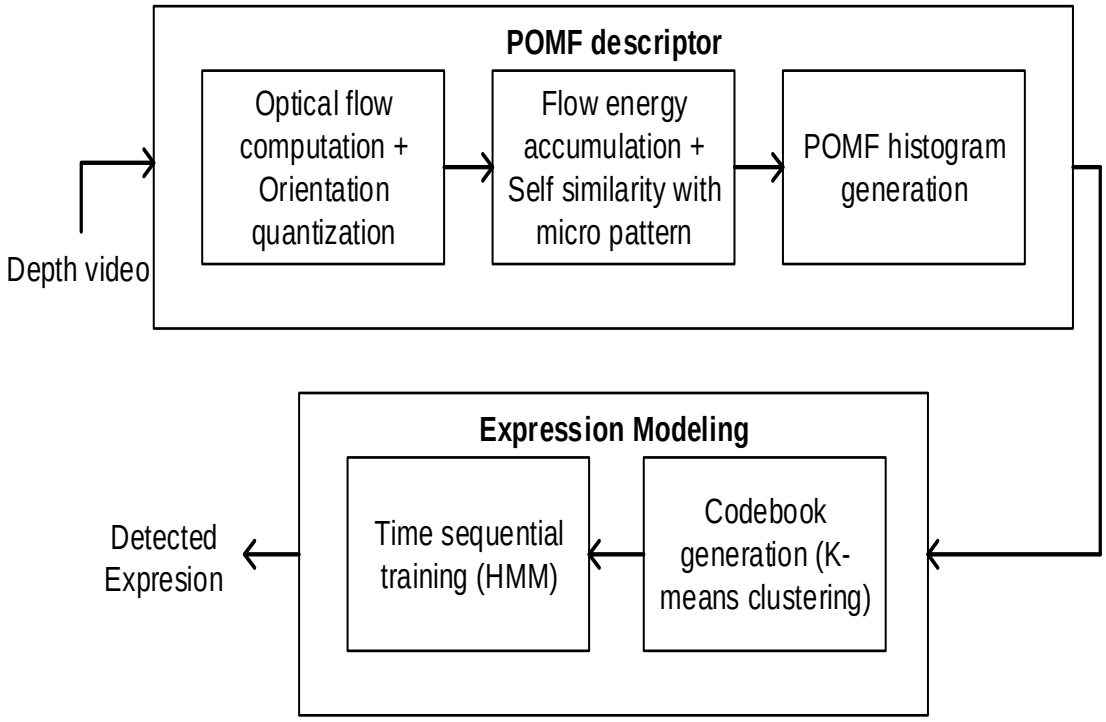

FIGURE 3.1: Steps of the proposed FER system

($3D$ or $n-D$ cases are similar) a voxel at location (x, y, t) with intensity $I(x, y, t)$ will have moved by Δx, Δy and Δt between the two image frames, and the following image constraint equation can be given:

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \tag{3.1}$$

If we consider that the movement is very small, the image constraint at I(x,y,t) with Taylor series can be developed to get:

$$I(x + \Delta x, y + \Delta y, t + \Delta t) = I(x, y, t) + \frac{\delta I}{\delta x}\Delta x + \frac{\delta I}{\delta y}\Delta y + \frac{\delta I}{\delta t}\Delta t + H.O.T \tag{3.2}$$

From this equation it follows that:

$$\frac{\delta I}{\delta x}\Delta x + \frac{\delta I}{\delta y}\Delta y + \frac{\delta I}{\delta t}\Delta t = 0, \tag{3.3}$$

which results

$$I_x u + I_y v + I_t = 0 \tag{3.4}$$

where $I_x u$ and $I_y v$ represent the derivatives in the corresponding u, v and time dimensions. Here, the main challenge of optical flow estimation is which property to track and how to track it. More precisely, it needs to track a property which includes motion information more robustly. Several image properties have been used for this purpose throughout different optical flow estimation methods [32, 33]. Considering the velocity,

filed associated with image changes [51], it can be estimated based on the assumptions of brightness constancy, gradient constancy and flow smoothness.

Another important issue is that in the previous equation we have two unknowns but one equation only. This is known as the aperture problem of the optical flow algorithms. There are two kinds of classic approaches to solve this problem. One is local approach and another one is global approach.

The local approaches consider that the displacement of the image contents between two nearby instants (frames) is small and approximately constant within a neighborhood of a particular point under consideration. Thus the optical flow equation can be assumed to hold for all pixels within a window centered at a specific point. So, the local image flow (velocity) vector (V_x, V_y) which is the vector representation of (u, v) must satisfy the following equations.

$$\begin{array}{rcl} I_x(q_1)V_x + I_y(q_1)V_y + I_t(q_1) &=& 0 \\ I_x(q_2)V_x + I_y(q_2)V_y + I_t(q_2) &=& 0 \\ & \vdots & \\ I_x(q_n)V_x + I_y(q_n)V_y + I_t(q_n) &=& 0 \end{array} \tag{3.5}$$

Where, $q_1, q_2, ..., q_n$ are the pixels inside the window and $I_x(q_i), I_y(q_i), I_t(q_i)$ are the partial derivatives of the image I with respect to x, y and time t, evaluated at the point q_i. Lucas et al. [32] used this kind of solution. Later on, Anandan et al. [33] showed how a series of local discrete search steps can be interleaved with Lucas-Kanade [32] incremental refinement steps in a coarse-to-fine pyramid scheme, which allows the estimation of large motions.

On the other hand, the global approaches consider smoothness in the flow over the whole image. The main goal is to minimize the distortions in the flow and prefers solutions which represent more smoothness. The flow is formulated as a global energy function E which is then attempted to be minimized. This function is given for two-dimensional image streams like the following equations.

$$E = \int\int [(I_x u + I_y v + I_t) + \alpha^2(\|\nabla u\|^2 + \|\nabla v\|^2)]dxdy \tag{3.6}$$

Where, I_x, I_y and I_t are the derivatives of the image intensity values respectively to the x, y and time dimensions; ∇u and ∇v are the optical flows and α is a regularization constant. This function can be minimized by solving the associated multi-dimensional Euler-Lagrange equations.

$$\begin{array}{rcl} \frac{\partial L}{\partial u} - \frac{\partial}{\partial x}\frac{\partial L}{\partial u_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial u_y} &=& 0 \\ \frac{\partial L}{\partial v} - \frac{\partial}{\partial x}\frac{\partial L}{\partial v_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial v_y} &=& 0 \end{array} \tag{3.7}$$

Where L is the integrand of the energy expression, giving

$$\begin{aligned} I_x(I_x u + I_y v + I_t) - \alpha^2 \Delta u &= 0 \\ I_y(I_x u + I_y v + I_t) - \alpha^2 \Delta v &= 0 \end{aligned} \tag{3.8}$$

where, $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ denotes the Laplace operator. For example, Horn and Schunck [33] used this type of solution.

From any optical flow estimation, two kinds of flow information are found which are known as horizontal flow (u) and vertical flow (v). Each of the u and v reveals two directional flow information from an image. The positive u and the negative u represent the flow information from left to right and right to left respectively. On the other hand, the positive v and the negative v represent the flow information from top to bottom and bottom to top respectively. In our method, we have used Lucas-Kanade [32] methods to estimate the optical flow information.

3.3 Patterns of Oriented Motion Flow (POMF) Descriptor

In this thesis, we propose a directional optical flow based descriptor which is called Patterns of Oriented Motion Flow (POMF). The rudimentary idea of the POMF is to discretized the motion change information and captures the encoded micro pattern from those motion changes. At first, discretized motion change will be enhanced by the local motion changes and then further incorporated by the self-similarity measurements of LBP micro pattern. Taking the directional changes of image information and encoding those directional image velocity by LBP in POMF descriptor, a robust pattern will be generated.

3.3.1 POMF Code

Optical flow computation and orientation quantization: The first step in extracting the POMF feature, is the computation of optical flow between consecutive two image frames from the video. The optical flow of the image produces the two flows information: horizontal (u) and vertical (v) for each pixel. From these motion flow information, four directional flows information are produced discretely over the u and v. Positive u_p represents the horizontal flow from left to right whereas the negative u_n represents the reverse direction. Similarly, positive v_p represents the vertical flow from top to bottom and negative v_n represents the reverse direction. As a facial expression starts from a neutral state, and then expose the expression, and again end with a neutral expression, so here, certainly a prominent motion change is elicited through the different facial part like mouth, eye, eyebrow, nose, chin, forehead etc. Figure 3.2 shows prominent motion

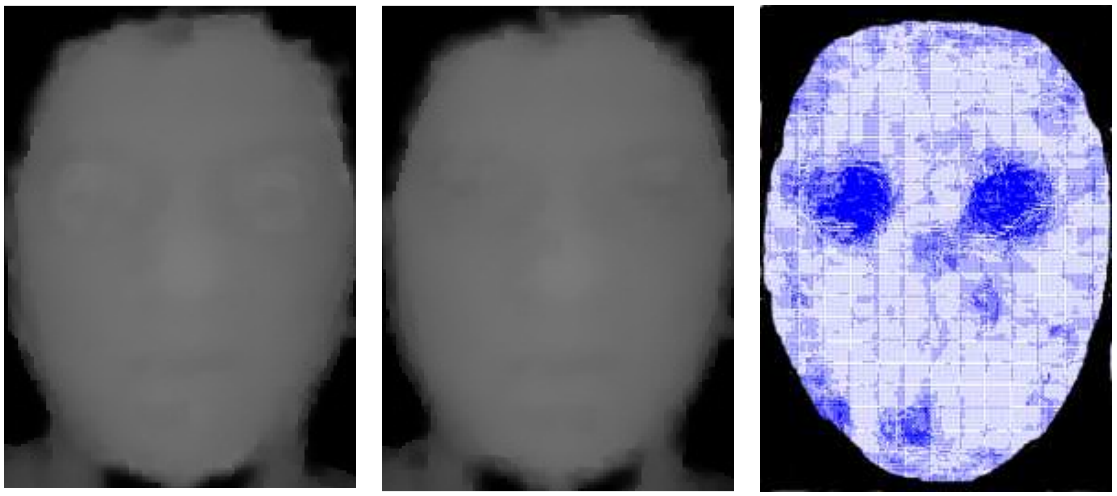

FIGURE 3.2: Consecutive two frames of an anger expressed depth video sample and corresponding optical flow response respectively (from left to right).

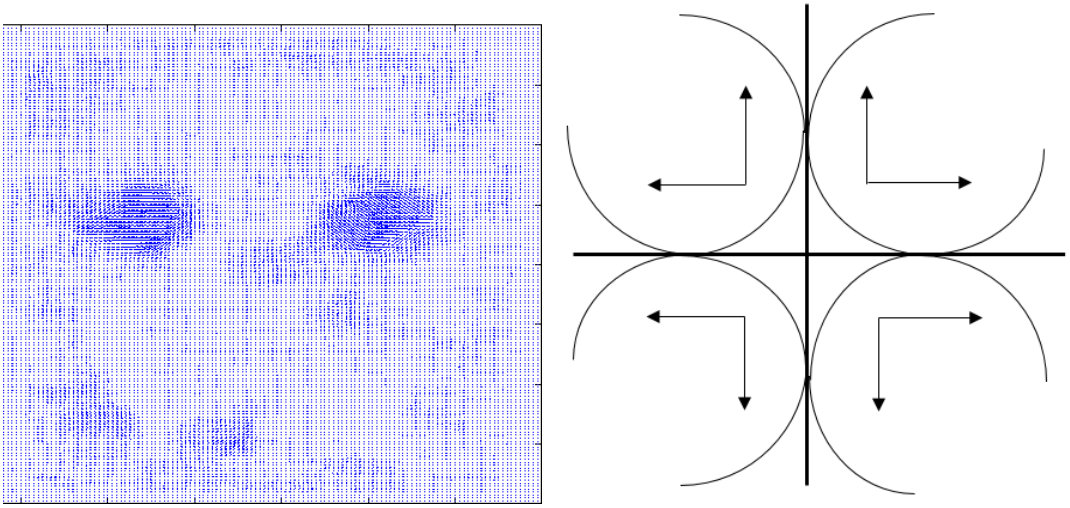

FIGURE 3.3: Optical flow response of t^{th} and $(t+1)^{th}$ frames of an anger expressed depth video sample (left), four divisions of discrete directional motion flow (right)

changes of two consecutive depth image frames. Our intention is to make some directional motion vector from these horizontal and vertical motion information. A total number of four directional patterns are created which represent flow energy information in four discrete directions U_pV_p, U_pV_n, U_nV_n, U_nV_p (figure: 3.3). As a result, we can get a robust pattern from the response of the directional approach (Figure: 3.4).

Local flow energy accumulation: The second step is to introducing the motion flow information from the neighboring region. A local histogram of the motion orientation changes over all cell pixels is estimated. Here, flow energy $(u^2 + v^2)$ can be used as a vote weight or some function of the flow energy. In our original POMF descriptor, we have used flow energy. As a result, each significant motion pixel is identified as any of the four directional motion change contributing pixel where it contains the flow energy.

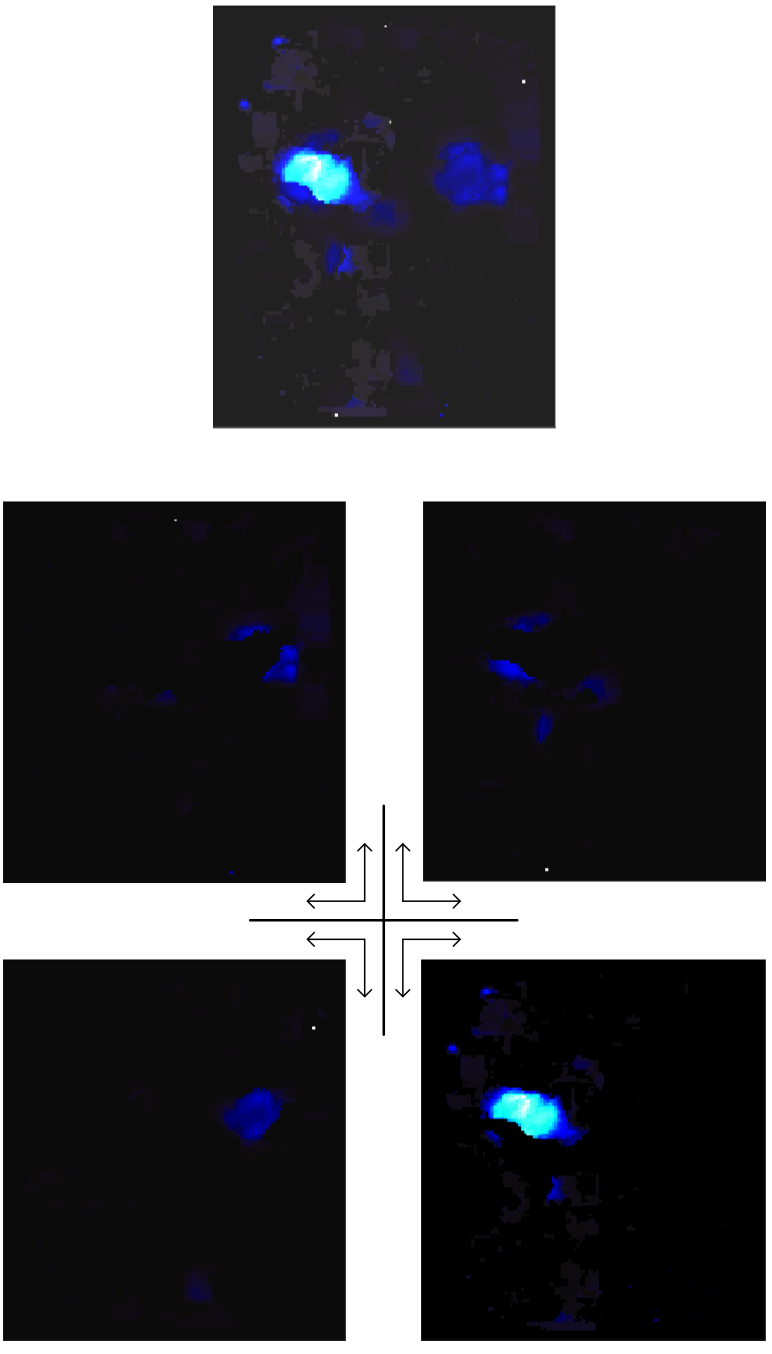

FIGURE 3.4: Responses of the two consecutive frames after discrete flow orientations. Flow energy response of two consecutive anger expressed depth video sample (upper row), directional flow U_nV_p (middle row, left), directional flow U_pV_p (middle row, right), directional flow U_nV_n (lower row, left), and directional flow U_pV_n (lower row, right).

Global self-similarity estimation: The last step of the POMF descriptor is to encode the accumulated directional flow information by LBP descriptor. Here our intention is to find out the self-similarity in a global region. LBP is a robust texture descriptor which is robust for encoding any pattern information providing the rotation invariance of the structure. In the original LBP descriptor [17], a uniform LBP was suggested which more robust with finer angular space. In our experiment, the uniform LBP is used as the significant facial expression pattern appear frequently and it performs better than the LBP.

We continue LBP encoding process on the accumulated flow energy across four different flow direction to build the final POMF descriptor. A POMF feature is calculated at every pixel position p over each discretized flow direction (figure: 3.5).

$$POMF^{\theta_i}_{B,C,N}(p) = \sum_{j=1}^{N} f(S(E^{\theta_i}_{p}, E^{\theta_i}_{j}))2^j \tag{3.9}$$

Here, θ_i represents the directional flow at $i(i = U_pV_p, U_pV_n, U_nV_p, U_nV_n)$ direction; $E^{\theta_i}_{p}, E^{\theta_i}_{j}$ are the directional optical flow value of central pixel p, and surrounding pixels respectively; S is the similarity function; B, C refer to the size of blocks and cells respectively; N is the number of pixels surrounding the considered central pixel p; and f is defined as:

$$f(x) = \begin{cases} 1 & x \geq \alpha \\ 0 & x < \alpha \end{cases} \tag{3.10}$$

Here, the value α is slightly larger than zero which ensures some stability in uniform regions. Finally, from two consecutive image frames, the descriptor will be the concatenation of these unidirectional POMF of four directional flow:

$$POMF_{B,C,N}(p) = \{POMF^{U_pV_p}, POMF^{U_pV_n}, POMF^{U_nV_p}, POMF^{U_nV_n}\} \tag{3.11}$$

3.3.2 POMF Histogram

For facial expression recognition from a video sequence, at first consecutive image frames are extracted from the video where an expression will be represented by some sequential static images. Here, image frame should be extracted in such a way so that a significant motion change is introduced. As optical flow information will be used for POMF, so the presence of significant motion change will provide better results. Then for every two consecutive image frames, an optical flow information is estimated. In our experiment, we used Lucas-Kanade [32] method for estimating the optical flow information. From the optical flow information, two kinds of motion flow information: horizontal (u) and

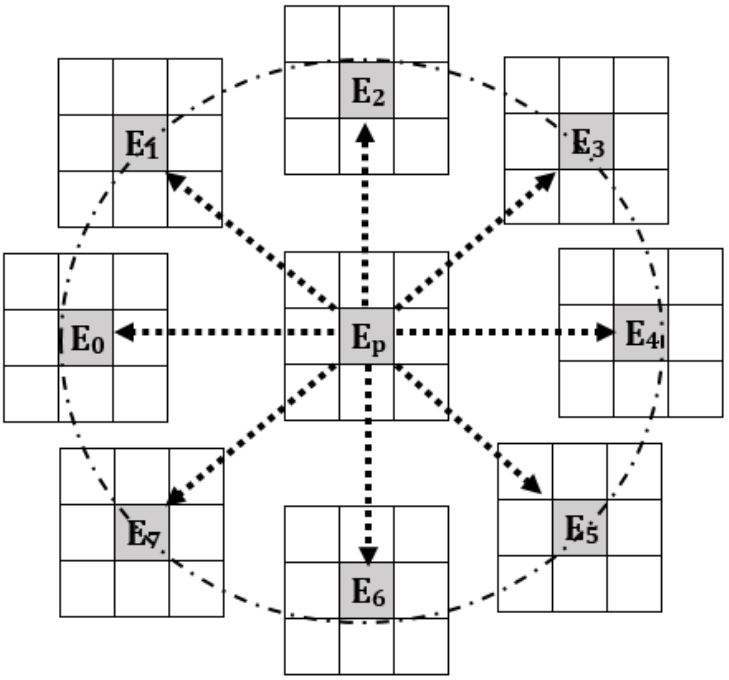

FIGURE 3.5: Estimation of self-similarity over region

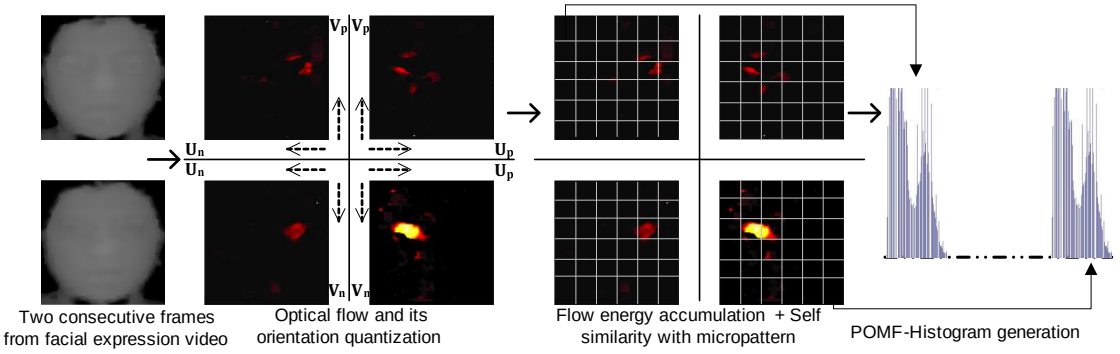

FIGURE 3.6: Steps of POMF feature extraction from optical flow

vertical (v) are generated. On this u and v, the proposed POMF code is applied and produced the four directional encoded motion flow information. Now from each consecutive two image frames, a POMF feature vector is generated. Basically, from the optical flow information of each frame, four new directional frames will be appeared and each of them is divided into multiple non-overlapping regions and a histogram is extracted from each region which will create the POMF-HS. So if the video is divided into n number of image frames, then $(n-1)$ number of POMF Histogram (POMF-HS) will be generated. All the accumulated feature vectors are used as the feature representation of that particular facial expression from the video. Figure 3.6 shows the steps of POMF feature extraction from two consecutive image frames.

3.3.3 Robustness of POMF descriptor

POMF descriptor contains not only the local significant motion changes from two consecutive frames but also represents the directional motion information in neighboring regions. It determines the following properties:

- POMF is a directional approach. So it clarifies its robustness against any directional motion changes with different levels of accuracy.
- POMF descriptor reveals multi-resolution feature because of different scales of cells and blocks. At a time it contains both local and global information.
- Introducing flow energy information at each pixel, it represents horizontal and vertical motion effect at the same time which make it more robust for identifying proper motion changes.
- As we consider the motion flow information from the image frames, so we can easily overcome the facial problems like age, beard, pose, gender etc.

Therefore, the POMF descriptor contains richer information from the facial expression video. It considers the relationship of a frame to frame by the directional optical flow. As a result, the expression changes from neutral status to final status is robustly represented in the POMF. Moreover, the rest of the part except the expression is almost same throughout all of the frames of any particular video. As a result, only the expression exposed information will be captured which makes POMF robust against the variations like illumination, background pattern, beard, facial hair, gender, pose etc.

3.4 Facial Expression Modeling and Recognition

While proper robust feature extraction is done, then the next stage of the facial expression recognition is modeling or training the sample sequences. In order to do that, we applied HMM modeling and we test the samples using each HMM trained model for recognition. Highest likelihood containing expression is ultimate recognized facial expression.

3.4.1 Modeling the Expression

After the POMF feature extraction, discrete HMMs are applied for expression training. But HMM takes the observation sequences value. So for doing it, we have applied K-means [45] clustering technique to generate our desired observation sequences. K-means

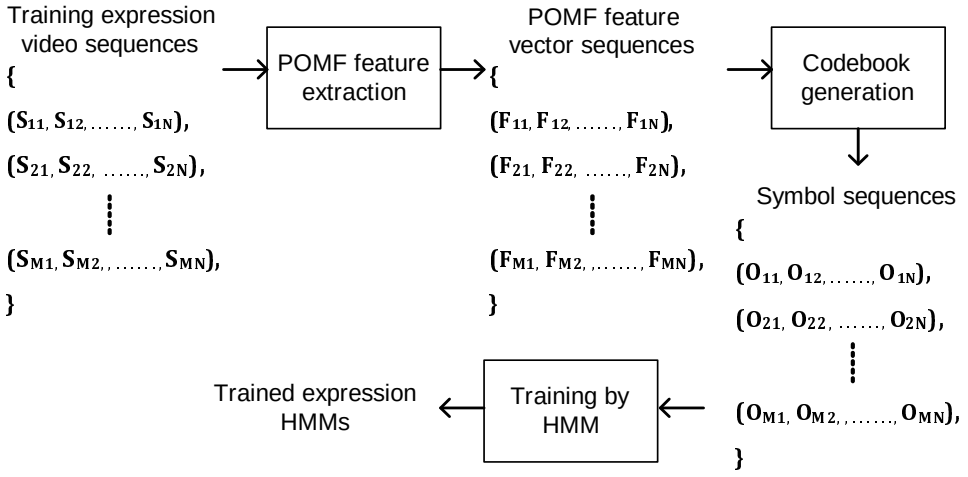

FIGURE 3.7: Training phase of the FER system

clustering technique takes all the samples and clusters the samples based on their features. Here, the optimum cluster size is needed to declare. For all type of expressions, all sequences are clustered by the codebook and observation symbol sequences are produced which will be trained by HMMs.

In the figure 3.7, how the training phase is done explained step by step. First of all, we have different image sequences S_M which are the representations of sample videos. Each of the samples S_M is divided into several sequences of frames S_N. From these, each of S_Ns, feature vector F_Ns are extracted using the POMF histogram feature. Then each of the feature vector F_Ns is needed to convert an observation number and that is done by the K-means clustering technique. While K-means clustering is performed, each of the feature vector F_Ns will be represented by a particular symbol. All the feature vector F_Ns from a sample video will create symbol sequences of O_Ns. Then these symbolized sequences O_Ms will be used to train the models of expressions by HMM.

3.4.2 Recognizing the Expression

Now after the model creation, if we want to test a sample video of facial expression, then like the same procedure, at first proper feature vectors should be extracted by POMF feature descriptor. Each of the trained models is used to generate the likelihood response for the particular sample observation sequence. Finally, to determine the test observation sequence, highest likelihood response of all T-trained expression HMMs evokes the corresponding desired class of the facial expression.

$$Detected\ Expression = arg \max_{k=1}^{k=T}(P(O|H_T)) \tag{3.12}$$

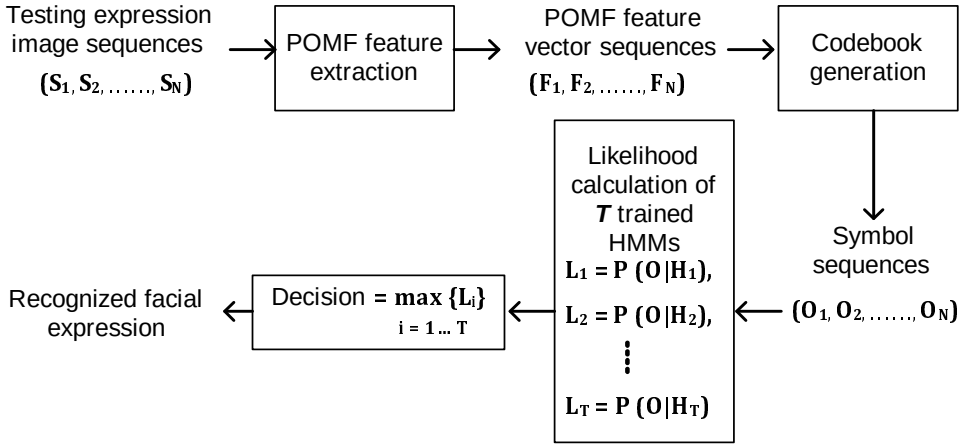

FIGURE 3.8: Testing phase of the FER system

In the figure 3.8, how to test a video sequence is shown step by step. Like the training phase, at first from the test sample video, S_Ns image frames are extracted. Then from each of the frames S_N, POMF histogram feature vectors are extracted which will generate the feature vector F_N. To create the symbolic sequence O_N K-means clustering technique is applied. Now, this symbolic sequence is tested over the trained HMM expression models. Finally, among all of the expression models, highest likelihood captured model will be considered as the recognized model of the facial expression.

Chapter 4

Experimental Analysis

In this chapter, we evaluate the performance of the proposed POMF descriptor for facial expression recognition system. We will present our proposed method's performance drawing the comparison with some other prominent methods which will be applied in both RGB and Depth database. Besides, in the latter part, we will also show the performance of our proposed method in case of changing other parameters and the computation time.

4.1 Data Set and Experimental Setup

In our FER experiments, two types of facial expression database are used which are CK database [52] and Depth database [26]. As we are dealing with video, so for both case video samples of facial expression are used.

The CK database [52] consists of 100 university students who at the time of their inclusion were between 18 to 30 years old; 65% were female, 15% were African-American, and 3% were Asian or Latino. Subjects were instructed to perform a series of facial expression displays starting from neutral or nearly neutral to one of six target prototypic emotions. Image sequences from neutral to target display were digitized into 640×480 or 640×690 pixel arrays of gray scale frames. In our setup, we selected 60 image sequences, each of which was labeled as one of the six basic emotions: surprise, anger, fear, disgust, happiness and sadness. Figure 4.1 and 4.2 show some of the samples of the CK database [52]. On the other hand, the Depth database [26] contains both RGB and Depth image sequences containing the six basic expressions: surprise, anger, fear, disgust, happiness and sadness. Human is able to percept RGB images but we are dealing with Machine. Machine's perception is different than Human. So we can also provide some more information rich image to the Machine. Hence, the concept comes about depth image. In the depth image, high pixel value represents a near distance and low pixel value represents

FIGURE 4.1: Sample facial expression images from the CK database

FIGURE 4.2: Examples of different expressions of the CK database. Anger, Disgust, Fear, Happiness, Sadness, Surprise expression respectively (from left to right)

a far distance. Depth information greatly contributes to the facial expression. Figure 4.3 shows examples of different facial expression of the Depth database[26] and Figure 4.4 shows the contributions of depth information in a facial image. Besides, In depth database [26], the expression video clips were of different length, and each video began and ended with a neutral expression. A total of 120 sequences, each of which was one of the six types of expressions, are used.

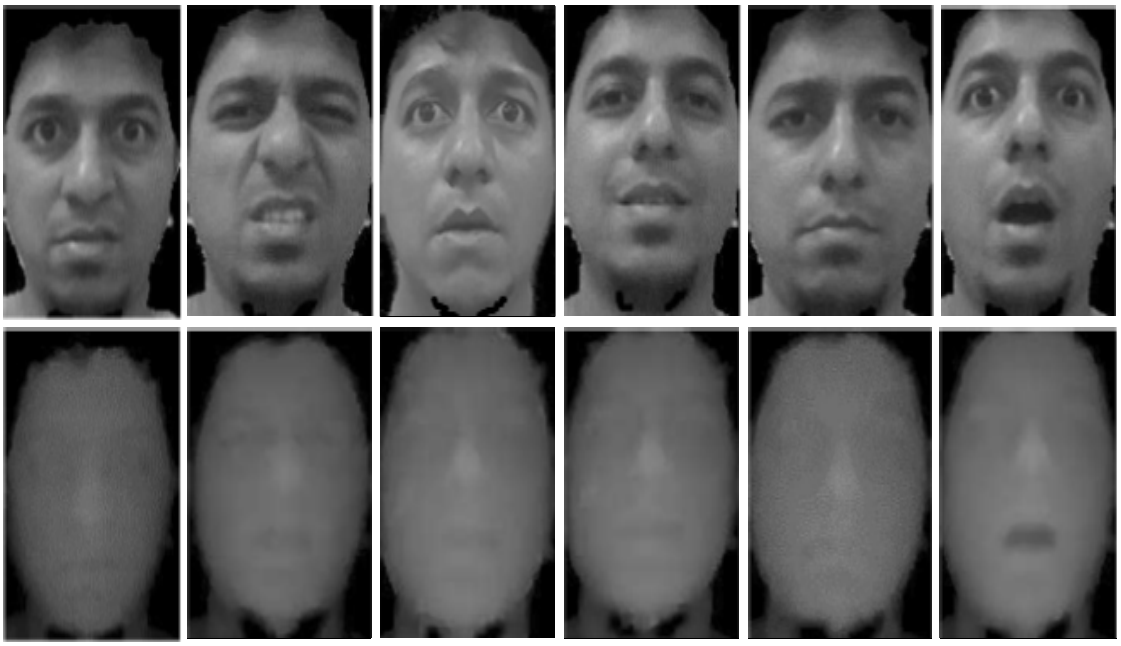

FIGURE 4.3: Examples of different expressions of the Depth database. Anger, Disgust, Fear, Happiness, Sadness, Surprise expression respectively (from left to right); normal gray faces (upper row) and the corresponding depth faces (lower row)

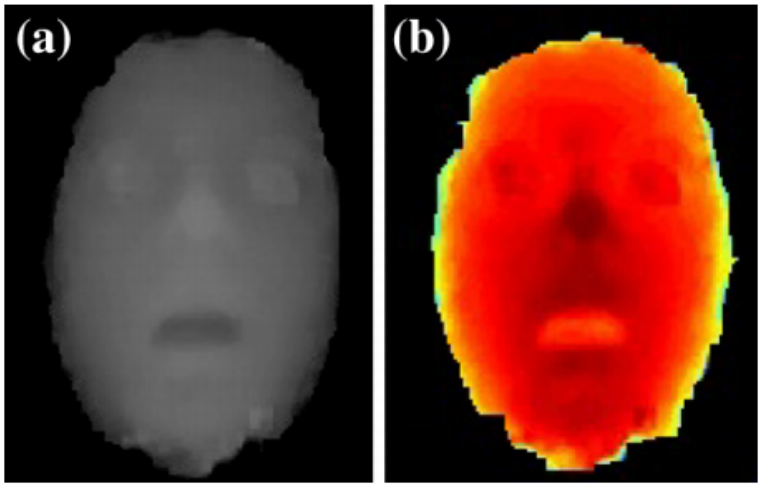

FIGURE 4.4: Depth image (a) and corresponding pseudo-color-distribution image (b) of a surprise expression

TABLE 4.1: Confusion matrix using CK database with Optical flow-PCA

Expression	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Anger	70	20	0	0	10	0
Disgust	0	80	20	0	0	0
Fear	0	0	70	20	10	0
Happiness	0	0	10	90	0	0
Sadness	0	0	20	0	80	0
Surprise	0	10	10	0	0	80

TABLE 4.2: Confusion matrix using CK database with POMF

Expression	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Anger	80	10	0	0	10	0
Disgust	10	80	10	0	0	0
Fear	0	0	80	20	0	0
Happiness	0	0	10	90	0	0
Sadness	10	0	0	0	90	0
Surprise	0	0	0	0	0	100

TABLE 4.3: Average expression recognition rates on CK database

Face Representation	**Recognition Rate (%)**
Optical flow-PCA	78.83
POMF	86.67

4.2 Performance Analysis

4.2.1 Performance on CK Database

CK database [52] is the collection of RGB image sequences only. There is no depth information in the samples. Most of the optical flow based methods tried to track the geometric shape through optical flow and, therefore, most of them are geometric-based methods. But we are working with optical flow in the appearance-based method. Recently, Zia [41] proposed a method based on optical flow information and PCA methods. So at first, we try to compare the performance with Optical flow-PCA [41]. When we test the samples, some are identified as misclassified and results in other classes which are represented by the confusion matrix. Tables 4.1, 4.2 show the confusion matrixes of both Optical flow-PCA [41] and our proposed POMF with the same experimental setup. If we look at the experimental results on Table 4.3, it looks very clear how we get the upgrade in the accuracy. For the same experimental setup, our proposed POMF gains 86.67%, whereas the Optical flow-PCA [41] was 78.83%.

TABLE 4.4: Confusion matrix using RGB faces with OF-PCA.

Expression	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Anger	72.50	0	10	0	17.50	0
Disgust	10	75	0	0	15	0
Fear	0	0	75	5	20	0
Happiness	10	0	0	80	0	10
Sadness	0	0	15	0	77.50	7.50
Surprise	10	10	0	0	0	80

TABLE 4.5: Confusion matrix using RGB faces with POEM.

Expression	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Anger	90	0	0	0	10	0
Disgust	10	85	0	0	5	0
Fear	0	0	85	0	15	0
Happiness	0	0	0	85	0	15
Sadness	10	0	0	0	90	0
Surprise	12.50	2.50	0	0	0	85

4.2.2 Performance on Depth Database

For expression recognition we have also compared the performance of the proposed POMF with some promising existing methods, namely Optical flow-PCA [41], POEM [31], LDP-PCA [26]. All these approaches were performed based on the same experimental setup. Although all of the methods are not optical flow based but we try to draw a comparative performance with the other promising appearance based methods.

4.2.2.1 Performance Evaluation on RGB Video Sequences

Since depth database [26], contains both RGB and Depth video sequences, so applied the methods for the both cases. In case of RGB video images, confusion matrixes of the methods are shown in Tables 4.4, 4.5, 4.6, 4.7. If we carefully look at the Table 4.8 it will be more clear, actually how different methods are acting. POEM and LDP-PCA show almost similar performance. POEM achieved 86.67% while LDP-PCA achieved 87.08%. Both of the approaches use a feature representation instead of intensity values. LDP uses the neighborhood edge responses values whereas, POEM considers the gradient value. On the other hand, proposed POMF outperforms others achieving the highest accuracy 87.91% while the direct Optical flow-PCA approach was only 76.67%.

4.2.2.2 Performance Evaluation on Depth Video Sequence

So far we have seen the performance with the RGB images. But like we said previously, depth information contains great contributions in case of facial images. Tables 4.9,

TABLE 4.6: Confusion matrix using RGB faces with LDP-PCA.

Expression	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Anger	85	10	0	0	5	0
Disgust	0	87.50	0	2.50	10.50	0
Fear	0	0	85	5	10	0
Happiness	2.50	0	5	87.50	0	5
Sadness	0	0	0	2.50	87.50	10.5
Surprise	0	7.50	2.50	0	0	90

TABLE 4.7: Confusion matrix using RGB faces with POMF.

Expression	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Anger	90	0	0	0	10	0
Disgust	5	85	0	0	10	0
Fear	0	0	85	0	15	0
Happiness	0	0	0	87.50	0	12.50
Sadness	10	0	0	0	90	0
Surprise	0	0	0	10	0	90

TABLE 4.8: Average expression recognition rates for different approaches on RGB sequences

Face Representation	**Recognition Rate (%) on RGB**
Optical Flow-PCA	76.67
POEM	86.67
LDP-PCA	87.08
POMF	87.91

TABLE 4.9: Confusion matrix using Depth faces with OF-PCA.

Expression	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Anger	77.50	0	5	0	17.50	0
Disgust	5	85	0	0	10	0
Fear	0	0	77.50	5	17.50	0
Happiness	10	0	0	85	0	5
Sadness	0	0	10	0	80	10
Surprise	10	5	0	0	0	85

4.10, 4.11, 4.12 show the performance of the existing methods on the Depth video sequences providing the confusion matrixes. It can be easily observed that how all the method's performance is improved in the case of Depth images. In the Table 4.13, all the approaches manifest a better recognition rate than the RGB video images. POEM and LDP-PCA both approaches reached a better recognition rate to 92.50% and 92.92% respectively. An improved recognition rate of 81.67% is also found for Optical flow-PCA approach than its prior RGB video. But POMF gave the best recognition rate of 94.17%. However, PCA-based method takes a higher computation time for large feature vector as it needs to calculate the covariance matrix for the computation of eigen vector and eigen value.

TABLE 4.10: Confusion matrix using Depth faces with POEM.

Expression	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Anger	92.50	0	0	0	7.50	0
Disgust	7.50	92.5	0	0	0	0
Fear	0	0	90	0	10	0
Happiness	0	0	0	95	0	5
Sadness	5	0	0	0	95	0
Surprise	5	5	0	0	0	90

TABLE 4.11: Confusion matrix using Depth faces with LDP-PCA.

Expression	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Anger	92.50	0	0	0	7.50	0
Disgust	0	92.50	0	0	7.50	0
Fear	0	0	92.50	0	7.50	0
Happiness	7.50	0	0	92.50	0	0
Sadness	0	0	0	0	92.50	7.50
Surprise	0	5	0	0	0	95

TABLE 4.12: Confusion matrix using Depth faces with POMF.

Expression	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Anger	95	0	0	0	5	0
Disgust	7.50	92.50	0	0	0	0
Fear	0	0	92.50	0	7.50	0
Happiness	0	0	0	95	0	5
Sadness	5	0	0	0	95	0
Surprise	0	0	0	5	0	95

TABLE 4.13: Average expression recognition rates for different approaches on Depth sequences

Face Representation	**Recognition Rate (%) on Depth**
Optical Flow-PCA	81.67
POEM	92.50
LDP-PCA	92.92
POMF	94.17

4.2.3 Performance Evaluation with Different Parameters

In all of our experiments, we have used the same experimental setup to evaluate the performance. To train by HMM, the features were symbolized by the K-means clustering [45] technique using a cluster size of 40 and there were 5 intermediate hidden states throughout all the experiments which are selected empirically. For the Depth database [26], to train and test each facial expression model, 20 and 40 image sequences were applied respectively.

HMM is popular to decode time-sequential events and considered to be better than others [26, 53, 54]. Our depth image-based experiments are further enhanced in this

TABLE 4.14: Expression recognition performance (%) using different classifiers on Depth Video.

Activity	SVM	KNN	Naïve-Bayes	HMM
Anger	85	80	82.50	95
Disgust	77.50	82.50	87.50	92.50
Fear	77.50	87.50	92.50	92.50
Happiness	82.50	82.50	92.50	95
Sadness	77.50	87.50	82.50	95
Surprise	85	82.50	82.50	95
Average	80.83	83.75	86.67	94.17

TABLE 4.15: Accuracy with different size of codebook using POMF on Depth Database

Codebook Size	**Recognition Rate (%) on Depth**
20	91.67
30	94.13
40	94.17
50	91.67

work to show HMM's superiority over other traditional classifiers such as multiclass Support Vector Machine (SVM) [46] using polynomial kernel, K-Nearest Neighbours [55] and Nave-Bayes [56] classifier. Table 4.14 shows the comparative performance of different classifiers using POMF on Depth database [26].

Table 4.15 shows the average accuracy rate with different codebook size. For the codebook size 20, the accuracy rate was 91.67% which was further increased to 94.13% by the increase of the codebook size 30. And it remained almost same by the codebook size 40. But after 40 when we took the size to 50, its accuracy again was fallen down.

Basically, it happens because of the inter-classes and intra-classes scattering. When the codebook size is too small, some overlaps encounter among different classes. As the proper number of clusters is not available, so the different feature vectors are taken to the same class. On the contrary, when the codebook size is too large, then some new classes are again clustered into new classes. As a result, same type of feature vectors who are supposed be in the same cluster, they are divided into some new clusters. So the solution is an optimum number of cluster size. And for our experiments, we can easily see that the optimum number of cluster size lied between 30-40.

4.2.4 Computation Time Analysis

On the other hand, nowadays because of the availability of high computer processing power the computation time is negligible. But we have analyzed to identify proposed system performance with the other methods. Table 4.16 shows the performance considering the computation time. Here, one particular depth video sequence was used to

TABLE 4.16: Average computation time for different methods on a particular video sample

Methods on Depth	**Computation Time(s)**
LBP	3.51
LDP	7.70
LDP-PCA	7.70
POEM	10.49
Optical Flow-PCA	28.20
POMF	34.19

measure the performance. There were total 10 image frames on that video sample. LBP method takes 3.51s whereas LDP takes 7.70s, which is almost double because in LDP, at first, we need to calculate the directional edge responses. On the other hand, POEM is taking 10.49s, which is almost 3 times higher time than LBP. Because at first, it finds the gradient image, then in two layers in encodes the results. Now all of the methods are based local appurtenance based. In the case of optical flow based method at first we need to calculate the optical flow from the image frames and it takes 27.89s and after that, if we want to apply optical flow-PCA method then it takes a lot of time depending on the sample size. Basically, when we will train the samples, a lot of time will be taken by PCA. But if we only consider the testing phase, then only projection time is the need which is considerably less and was around 28.20s. On the contrary, in the proposed POMF for the same case, it needs only 34.19s, in which the actual step needs only 6.3s which is close to POEM and considerable.

Chapter 5

Conclusion

5.1 Summary of the Contributions

In this thesis, an optical flow based facial expression recognition system is proposed where the directional pattern encoded information is used from the optical flow of consecutive depth images. We proposed a novel and robust facial descriptor called Patterns of Oriented Motion Flow (POMF). Using this descriptor POMF histogram is generated from the sample frames to produce the expression feature vector. Finally, the objective sequences of the feature vectors are trained by the Hidden Markov Model (HMM) to produce the expression model.

As we work with the optical flow information and it represents only the changing information from a video, so we can easily capture the significant changes which occur because of the expression. Besides, we can easily overcome different challenges of expression like age, gender, beard, glasses etc. Moreover, the directional optical flow information ensures more robust feature description by generating an oriented pattern.

An experimental analysis on both RGB and Depth based video images is performed including some salient approaches to evaluate the strength of our proposed method. From the empirical results, it is obvious that our proposed POMF descriptor represents better recognition rate for depth based facial expression recognition system. Besides, it is also turn out that Depth image shows superior performance over RGB image.

5.2 Future Works

Although our proposed method showing better results but it is possible to enhance its performance by introducing the solution for nonlinearity. As human face images with large pose variation demonstrate significant nonlinearity, so we are planning to

incorporate the solution for nonlinearity to the descriptor. On the other hand, because of the robustness of the POMF descriptor, it should yield a better result in some other dynamic applications like human activity recognition or gait detection etc. So in the future, we would like to extend its usability over the other potential application fields.

Bibliography

[1] Andrew J Calder, A Mike Burton, Paul Miller, Andrew W Young, and Shigeru Akamatsu. A principal component analysis of facial expressions. *Vision research*, 41(9):1179–1208, 2001. [Cited on pages 3 and 10]

[2] Paul Ekman and Wallace V Friesen. Facial action coding system. 1977. [Cited on page 8]

[3] Joseph C Hager, Paul Ekman, and Wallace V Friesen. Facial action coding system. *Salt Lake City, UT: A Human Face*, 2002. [Cited on page 8]

[4] Zhengyou Zhang. Feature-based facial expression recognition: Sensitivity analysis and experiments with a multilayer perceptron. *International journal of pattern recognition and Artificial Intelligence*, 13(06):893–911, 1999. [Cited on page 8]

[5] Guodong Guo and Charles R Dyer. Simultaneous feature selection and classifier training via linear programming: A case study for face expression recognition. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 1, pages I–346. IEEE, 2003. [Cited on page 8]

[6] Michel F Valstar, I Patras, and Maja Pantic. Facial action unit detection using probabilistic actively learned support vector machines on tracked facial point data. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)-Workshops*, pages 76–76. IEEE, 2005. [Cited on page 8]

[7] Michel Valstar and Maja Pantic. Fully automatic facial action unit detection and temporal analysis. In *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, pages 149–149. IEEE, 2006. [Cited on page 8]

[8] Maja Pantic and Leon J. M. Rothkrantz. Automatic analysis of facial expressions: The state of the art. *IEEE Transactions on pattern analysis and machine intelligence*, 22(12):1424–1445, 2000. [Cited on page 8]

[9] Irene Kotsia and Ioannis Pitas. Facial expression recognition in image sequences using geometric deformation features and support vector machines. *IEEE transactions on image processing*, 16(1):172–187, 2007. [Cited on page 9]

[10] Curtis Padgett and Garrison W Cottrell. Representing face images for emotion classification. *Advances in neural information processing systems*, pages 894–900, 1997. [Cited on page 10]

[11] Gianluca Donato, Marian Stewart Bartlett, Joseph C. Hager, Paul Ekman, and Terrence J. Sejnowski. Classifying facial actions. *IEEE transactions on pattern analysis and machine intelligence*, 21(10):974–989, 1999. [Cited on page 10]

[12] Séverine Dubuisson, Franck Davoine, and Mylène Masson. A solution for facial expression representation and recognition. *Signal Processing: Image Communication*, 17(9):657–673, 2002. [Cited on page 10]

[13] Ioan Buciu, I Pitas, et al. Ica and gabor representation for facial expression recognition. In *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, volume 2, pages II–855. IEEE, 2003. [Cited on page 10]

[14] CHEN Fan and Kazunori Kotani. Facial expression recognition by supervised independent component analysis using map estimation. *IEICE TRANSACTIONS on Information and Systems*, 91(2):341–350, 2008. [Cited on page 10]

[15] Michael J Lyons, Julien Budynek, and Shigeru Akamatsu. Automatic classification of single facial images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(12):1357–1362, 1999. [Cited on page 10]

[16] Md Zia Uddin, JJ Lee, and T-S Kim. An enhanced independent component-based human facial expression recognition from video. *IEEE Transactions on Consumer Electronics*, 55(4), 2009. [Cited on pages 10 and 11]

[17] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987, 2002. [Cited on pages 11, 12, 13, and 30]

[18] Taskeed Jabid, Md Hasanul Kabir, and Oksam Chae. Local directional pattern (ldp) for face recognition. In *Consumer Electronics (ICCE), 2010 Digest of Technical Papers International Conference on*, pages 329–330. IEEE, 2010. [Cited on pages 11 and 14]

[19] Ngoc-Son Vu and Alice Caplier. Face recognition with patterns of oriented edge magnitudes. *Computer Vision–ECCV 2010*, pages 313–326, 2010. [Cited on pages 11 and 16]

[20] Caifeng Shan, Shaogang Gong, and Peter W McOwan. Robust facial expression recognition using local binary patterns. In *Image Processing, 2005. ICIP 2005. IEEE International Conference on*, volume 2, pages II–370. IEEE, 2005. [Cited on page 12]

[21] Guoying Zhao and Matti Pietikainen. Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE transactions on pattern analysis and machine intelligence*, 29(6), 2007. [Cited on page 12]

[22] Caifeng Shan, Shaogang Gong, and Peter W McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing*, 27(6):803–816, 2009. [Cited on page 12]

[23] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 28(12):2037–2041, 2006. [Cited on page 12]

[24] Taskeed Jabid, Md Hasanul Kabir, and Oksam Chae. Robust facial expression recognition based on local directional pattern. *ETRI journal*, 32(5):784–794, 2010. [Cited on pages 14 and 18]

[25] Taskeed Jabid, Md Hasanul Kabir, and Oksam Chae. Local directional pattern (ldp)–a robust image descriptor for object recognition. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, pages 482–487. IEEE, 2010. [Cited on page 14]

[26] Md Zia Uddin. An efficient local feature-based facial expression recognition system. *Arabian Journal for Science and Engineering*, 39(11):7885–7893, 2014. [Cited on pages 14, 18, 35, 36, 39, 41, and 42]

[27] Ngoc-Son Vu, Hannah M Dee, and Alice Caplier. Face recognition using the poem descriptor. *Pattern Recognition*, 45(7):2478–2488, 2012. [Cited on page 16]

[28] Ngoc-Son Vu, Huu-Tuan Nguyen, and Alice Caplier. Multiple patterns of gradient magnitudes for face recognition. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 589–592. IEEE, 2012. [Cited on page 16]

[29] Ngoc-Son Vu and Alice Caplier. Enhanced patterns of oriented edge magnitudes for face recognition and image matching. *IEEE Transactions on Image Processing*, 21(3):1352–1365, 2012. [Cited on page 16]

[30] Ngoc-Son Vu. Exploring patterns of gradient orientations and magnitudes for face recognition. *IEEE Transactions on Information Forensics and Security*, 8(2):295–304, 2013. [Cited on page 16]

[31] E. Silva, C. Esparza, and Y. Meja. Poem-based facial expression recognition, a new approach. In *2012 XVII Symposium of Image, Signal Processing, and Artificial Vision (STSIVA)*, pages 162–167, Sept 2012. [Cited on pages 16 and 39]

[32] Bruce D Lucas, Takeo Kanade, et al. An iterative image registration technique with an application to stereo vision. 1981. [Cited on pages 17, 25, 26, 27, and 30]

[33] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981. [Cited on pages 17, 25, 26, and 27]

[34] Padmanabhan Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2(3): 283–310, 1989. [Cited on page 17]

[35] James R. Bergen, P. Anandan, Th J. Hanna, and Rajesh Hingorani. Hierarchical model-based motion estimation. pages 237–252. Springer-Verlag, 1992. [Cited on page 17]

[36] MASE Kenji. Recognition of facial expression from optical flow. *IEICE TRANSACTIONS on Information and Systems*, 74(10):3474–3483, 1991. [Cited on pages 18 and 24]

[37] Andreas Lanitis, Christopher J Taylor, and Timothy F Cootes. A unified approach to coding and interpreting face images. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 368–373. IEEE, 1995. [Cited on page 18]

[38] Michael J Black and Yaser Yacoob. Tracking and recognizing rigid and non-rigid facial motions using local parametric models of image motion. In *Computer Vision, 1995. Proceedings., Fifth International Conference on*, pages 374–381. IEEE, 1995. [Cited on pages 18 and 24]

[39] Yaser Yacoob and Larry S Davis. Recognizing human facial expressions from long image sequences using optical flow. *IEEE Transactions on pattern analysis and machine intelligence*, 18(6):636–642, 1996. [Cited on pages 18 and 24]

[40] Irfan A. Essa and Alex Paul Pentland. Coding, analysis, interpretation, and recognition of facial expressions. *IEEE transactions on pattern analysis and machine intelligence*, 19(7):757–763, 1997. [Cited on pages 18 and 24]

[41] Md Zia Uddin, Tae-Seong Kim, and Byung Cheol Song. An optical flow featurebased robust facial expression recognition with hmm from video. *International Journal of Innovative Computing, Information and Control*, 9(4):1409–1421, 2013. [Cited on pages 18, 38, and 39]

[42] Yoseph Linde, Andres Buzo, and Robert Gray. An algorithm for vector quantizer design. *IEEE Transactions on communications*, 28(1):84–95, 1980. [Cited on pages 18, 19, and 20]

[43] Y-I Tian, Takeo Kanade, and Jeffrey F Cohn. Recognizing action units for facial expression analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 23(2):97–115, 2001. [Cited on page 18]

[44] Fabrice Bourel, Claude C Chibelushi, and Adrian A Low. Robust facial expression recognition using a state-based model of spatially-localised facial dynamics. In *Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on*, pages 113–118. IEEE, 2002. [Cited on page 18]

[45] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA., 1967. [Cited on pages 19, 32, and 41]

[46] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995. [Cited on pages 20, 21, and 42]

[47] Paul J Werbos. Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE*, 78(10):1550–1560, 1990. [Cited on pages 20 and 21]

[48] Leonard E Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The annals of mathematical statistics*, 37(6):1554–1563, 1966. [Cited on pages 20 and 22]

[49] Leonard E Baum, Ted Petrie, George Soules, and Norman Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The annals of mathematical statistics*, 41(1):164–171, 1970. [Cited on pages 20 and 22]

[50] Maodi Hu, Yunhong Wang, Zhaoxiang Zhang, De Zhang, and James J Little. Incremental learning for video-based gait recognition with lbp flow. *IEEE transactions on cybernetics*, 43(1):77–89, 2013. [Cited on page 22]

[51] Kelson RT Aires, Andre M Santana, and Adelardo AD Medeiros. Optical flow using color information: preliminary results. In *Proceedings of the 2008 ACM symposium on Applied computing*, pages 1607–1611. ACM, 2008. [Cited on page 26]

[52] Takeo Kanade, Jeffrey F Cohn, and Yingli Tian. Comprehensive database for facial expression analysis. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 46–53. IEEE, 2000. [Cited on pages 35 and 38]

[53] Ira Cohen, Nicu Sebe, Ashutosh Garg, Lawrence S Chen, and Thomas S Huang. Facial expression recognition from video sequences: temporal and static modeling. *Computer Vision and image understanding*, 91(1):160–187, 2003. [Cited on page 41]

[54] Y Zhu, Liyanage C De Silva, and Chi Chung Ko. Using moment invariants and hmm in facial expression recognition. *Pattern Recognition Letters*, 23(1):83–91, 2002. [Cited on page 41]

[55] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967. [Cited on page 42]

[56] Andrew McCallum, Kamal Nigam, et al. A comparison of event models for naive bayes text classification. In *AAAI-98 workshop on learning for text categorization*, volume 752, pages 41–48. Citeseer, 1998. [Cited on page 42]