

University of Memphis

University of Memphis Digital Commons

Electronic Theses and Dissertations

7-25-2011

On The Evaluation of Model Based Approaches for Applications in Affective Computing

Md Iftekhar Tanveer

Follow this and additional works at: <https://digitalcommons.memphis.edu/etd>

Recommended Citation

Tanveer, Md Iftekhar, "On The Evaluation of Model Based Approaches for Applications in Affective Computing" (2011). *Electronic Theses and Dissertations*. 281.

<https://digitalcommons.memphis.edu/etd/281>

This Thesis is brought to you for free and open access by University of Memphis Digital Commons. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of University of Memphis Digital Commons. For more information, please contact khhgerty@memphis.edu.

To the University Council:

The Thesis Committee for Md. Iftekhar Tanveer certifies that this is the final approved version of the following electronic thesis: "On the Evaluation of Model Based Approaches for Applications in Affective Computing"

Mohammed Yeasin, Ph.D.
Major Professor

We have read this thesis and recommend
its acceptance:

Eddie Jacobs, D.Sc.

Xiangen Hu, Ph.D.

Accepted for the Graduate Council:

Karen D. Weddle-West, Ph.D.
Vice Provost for Graduate Programs

ON THE EVALUATION OF MODEL BASED APPROACHES FOR
APPLICATIONS IN AFFECTIVE COMPUTING

by

Md Iftekhar Tanveer

A Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

Major: Electrical and Computer Engineering

The University of Memphis

August 2011

Copyright © 2011 Md. Iftexhar Tanveer

All rights reserved

ACKNOWLEDGMENTS

I would like to thank Dr. Mohammed Yeasin for his constant support and guidance throughout the period of my graduate studies in United States. It would have been impossible for me to accomplish this work without his constructive suggestions, constant inspiration and kind co-operation throughout the progress of this work. I would also like to thank Dr. Eddie Jacobs and Dr. Xiangen Hu for being in my thesis committee.

This research was partially supported by grant NSF-IIS-0746790. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and not necessarily reflect the views of the funding institution. Portions of the research in this paper uses the MMI-Facial Expression Database collected by Valstar and Pantic.

I would like to thank my parents and my wife for always inspiring and supporting me. Without their support I would not be able to come along this far in my life. This thesis is dedicated to them.

ABSTRACT

Tanveer, Md. Iftekhhar. M.S. Electrical and Computer Engineering. The University of Memphis. August 2011. On The Evaluation of Model Based Approaches for Applications in Affective Computing. Major Professor: Mohammed Yeasin, Ph.D.

Automatic recognition of emotion has a huge potential in several applications. In order to address such potential, researchers from diverse fields are collaborating together to build systems capable of recognizing human emotion. As a preliminary step towards such systems, many works are being done to automatically detect facial expressions. A technique generally termed as “Model Based Technique” has gained significant attention among the researchers for its utility in detecting facial expressions.

However, methods currently used for evaluation of the performance of such systems have several flaws and inefficiencies. Due to these inefficient evaluation methods, it becomes difficult to compare among the systems from their literary descriptions. In this thesis, origins of such flaws are analyzed and efforts have been made to derive some solutions. As a part of this endeavor, a Three Level Evaluation (TLE) model has been proposed. In addition, some new and efficient assessment metrics have been suggested that can make faithful comparison of the systems.

TABLE OF CONTENTS

Contents	Pages
List of Figures	vii
1 Introduction	1
1.1 Background	1
1.2 Current Problems	2
1.3 Research Objectives	4
1.4 Challenges	4
1.5 Overview of the Following Chapters	5
2 Related Literature	6
3 Theoretical Background	9
3.1 Three Level Evaluation Model	9
3.2 Normalized Root Mean Squared Point Error (NRMS-PE)	10
3.3 Parameter of Discrimination	15
3.4 Normalizing the Differences among the Ground Truth Schemes	19
4 Methodologies	21
4.1 Databases	21
4.1.1 Extended Cohn Kanade Database	21
4.1.2 MMI Database	21
4.2 Experimental Setup	22
4.2.1 Standardized Points Features (SPTS)	25
4.2.2 Split Triangle Canonical Appearance Features (CAPPX)	27
5 Experiments and Results	32
5.1 Point Level Evaluation	32
5.1.1 Efficacy of NRMS-PE	32
5.1.2 Cumulative Error Distribution (CED) Chart	32
5.2 Feature Level Evaluation	35
5.2.1 Efficacy of Parameter of Discrimination	35
5.2.2 Relation between Number of Features and Parameter of Discrimination	35
5.2.3 Morphological and Appearance Properties of AUs	38
5.2.4 Effect of Hybridization on Parameter of Discrimination	40
5.3 Prediction Level Evaluation	43
5.3.1 Receiver Operating Characteristics (ROC)	43
5.3.2 AUC vs. AU: Effect of Features	43
5.3.3 AUC vs. AU: Effect of Classifiers	46
5.3.4 AUC vs. Number of Features	46
5.3.5 AUC vs. Parameter of Discrimination	51
6 Conclusion	52

Appendices	58
A Names and Examples of Action Units	58
B Model Based Landmark Detection Techniques	62
B.1 Active Appearance Model	62
B.2 Constrained Local Model (CLM)	64

LIST OF FIGURES

Figures	Pages
1.1 Example of a face annotated using a model based landmark tracker.	3
2.1 Chronological flow diagram of some important works related to model based analysis of face.	7
3.1 A typical block diagram of a facial expression detection system. (a) shows the current evaluation approach which is dependent on the last block only. (b) illustrates the proposed approach which evaluates the outcomes of all the blocks.	10
3.2 Effect of proportionate scaling.	11
3.3 Demonstration of the basic idea of Linear Discriminant Analysis (LDA). The stars and the crosses are representatives of high (m) dimensional data. LDA algorithm returns a set of orthogonal eigenvectors. If the stars and crosses are projected on the first eigenvector, it will maximally separate the projections. The second eigenvector will provide a separation lesser than the first one and so on. Dimensions of the eigenvectors are equal to the number of variables used to constitute a data point.	17
3.4 Probability Density Function of Normal Distribution with Standard Deviations [1].	18
3.5 Distances between two normal distributions.	18
3.6 Process of up-sampling the number of landmarks in order to normalize two non-compatible annotation schemes.	20
4.1 Number of Images from Extended Cohn Kanade database for different Action Units used in this work.	21
4.2 Number of Images from MMI database for different Action Units used in this work.	22
4.3 Overall block diagram of the prototype system implemented in order to evaluate a simple action unit recognition system.	23
4.4 Output given by FaceTracker.	24
4.5 Three Level Evaluation Module.	24
4.6 Translating to origin [Note: the lines are drawn for illustration purpose].	26

4.7	Process of removing rotation transformation. The dark shape denotes a reference shape.	27
4.8	Example of an image for which the process of PAW will be demonstrated.	28
4.9	The effect of Piecewise Affine Warping (PAW). The face region shown in Fig. 4.8 is warped into the reference base triangulation. The base triangulation is shown in (a). The resulting image is shown in (b).	29
4.10	Demonstration of splitting the triangles for calculating CAPP features. (a) CAPP91 (b) CAPP140.	30
5.1	(a) The shape formed by the landmarks of original image and corresponding RMS errors for different parts of face (b) RMS errors when the x axis is scaled 5 times (c)NRMS-PE is same for both the original and scaled versions.	33
5.2	Cumulative Error Distribution chart for the tracker employed in this work (FaceTracker [2]). The calculations were done in CK+ database.	34
5.3	Examples of images where the FaceTracker fails.	36
5.4	Projection of the positive and negative features for AU10 on the 1 st LDA eigenvector. \mathcal{D} represents the Parameter of Discrimination for a particular Number of Features (X).	37
5.5	Plot of Parameter of Discrimination (\mathcal{D}) vs. Number of Features for (a) MMI Database and (b) Extended Cohn Kanade (CK+) Database. Three most discriminative and three least discriminative AUs are shown as well as the mean of all the AUs used in this work.	39
5.6	Scatter plot of different AUs positioned based on the values of \mathcal{D} s obtained from SPTS features and CAPP570 features. (a) shows the plot for CK+ database and (b) shows the same for MMI database.	41
5.7	Parameter of Discrimination for different Features and Action Units. (a) shows the plot for CK+ (b) shows for MMI.	42
5.8	An example of ROC Curve.	44
5.9	AUC for different Features and Action Units. (a) shows the plot for CK+ (b) shows for MMI.	45
5.10	AUC for different classifier. (a) shows the plot for CK+ (b) shows for MMI.	47

- 5.11 Plots and 2nd order polynomial trend-lines of (a) AUC vs Number of Features for a few action units and (b) Mean AUC of all the action units vs Number of features. All the calculations are made using MMI database. 48
- 5.12 Plots and 2nd order polynomial trend-lines of (a) AUC vs Number of Features for a few action units and (b) Mean AUC of all the action units vs Number of features. All the calculations are made using CK+ database. 49
- 5.13 Plots and 2nd order polynomial trend-lines of (a) AUC vs Parameter of Discrimination for a few action units and (b) Mean AUC of all the action units vs Parameter of Discrimination. All the calculations are made using MMI database. 50

Chapter 1

Introduction

1.1 Background

“Affective computing is a field of computing that relates to, arises from, or deliberately influences human emotions” [3]. A major area of affective computing is the automated understanding of human emotion. While the human beings have innate abilities to predict emotions and affective states; such capabilities are very difficult for machines to emulate. However, machine recognition of emotion has huge potential in different fields. For example, it can be used to build an intelligent tutoring agent such as Auto-Tutor [4, 5] which is able to recognize whether a student is bored, confused or frustrated etc. It can also be used in medical or psychological treatments to detect pain [6] or depression [7] of the patients. Affective computing works to bridge the tremendous gap between the limited capability of machines and the large potential of emotion recognition applications.

A number of psychological studies [8, 9] have demonstrated that facial expression is strongly associated with human emotion. In order to represent and measure facial expressions, behavioral scientists use a widely known method named as “*Facial Action Coding System*” or *FACS* [10]. In 1976, Paul Ekman and Wallace Friesen developed this coding system by analyzing the anatomy of facial muscles and their effects on appearance of the face. According to this code, every possible change in face can be represented by 32 different Action Units (AU)¹. Although no emotional significance is carried through the AUs, they specify actions of certain muscle groups that produce a particular change in the face. Trained human coders observe these changes and express any facial expression in terms of AUs. However, scoring of AUs by human annotators is time consuming and requires much effort. Moreover, this is

¹ A list of Action Units used in this work is given in Appendix A

not suitable for automatic systems. Therefore, it is useful to build a system which is able to automatically detect AUs corresponding to a particular facial expression.

A plethora of reported literature has used many techniques to detect expressions from visual data. State-of-the-art techniques [11, 6, 7] have shown the potential in robust recognition of emotion from facial expression. Among these techniques, “Model Based” [12, 13, 14, 15, 16, 17] representations of the face have gained considerable attention. These methods utilize a mathematical model of probable deformations of the face in order to track some predefined landmark locations. They exhibit good tracking performance with a dense mesh of about 60-70 landmark points as shown in Fig. 1.1. Examples of model based techniques are Active Shape Model (ASM) [12], Active Appearance Model (AAM) [15], Constrained Local Model (CLM) [16, 17] etc. Several expression recognition approaches use these techniques to detect landmark points and to extract morphological and appearance features of the face. Then various machine learning techniques are used to predict facial expressions based on these extracted features.

1.2 Current Problems

Due to the success of model based techniques, a number of variants are reported and still being proposed by accounting for their limitations [12, 15, 18, 17, 16, 19, 20, 21, 14]. The evaluation of the variants of model based approaches became increasingly difficult due to their rapid growth. In order to effectively compare it is necessary to have a set of “*concise*” and “*invariant*” evaluation metric. By the term “*invariant*” it is indicated that the metrics should be consistent in their interpretations. An invariant evaluation metric makes two systems comparable with each other regardless their differences in implementation. On the other hand the term “*concise*” refers to a small set of parameters that is rendered to be more expressive in terms of the evaluation of systems. Without a set of concise and invariant metric it is difficult to compare competing approaches and to deduce right conclusions.

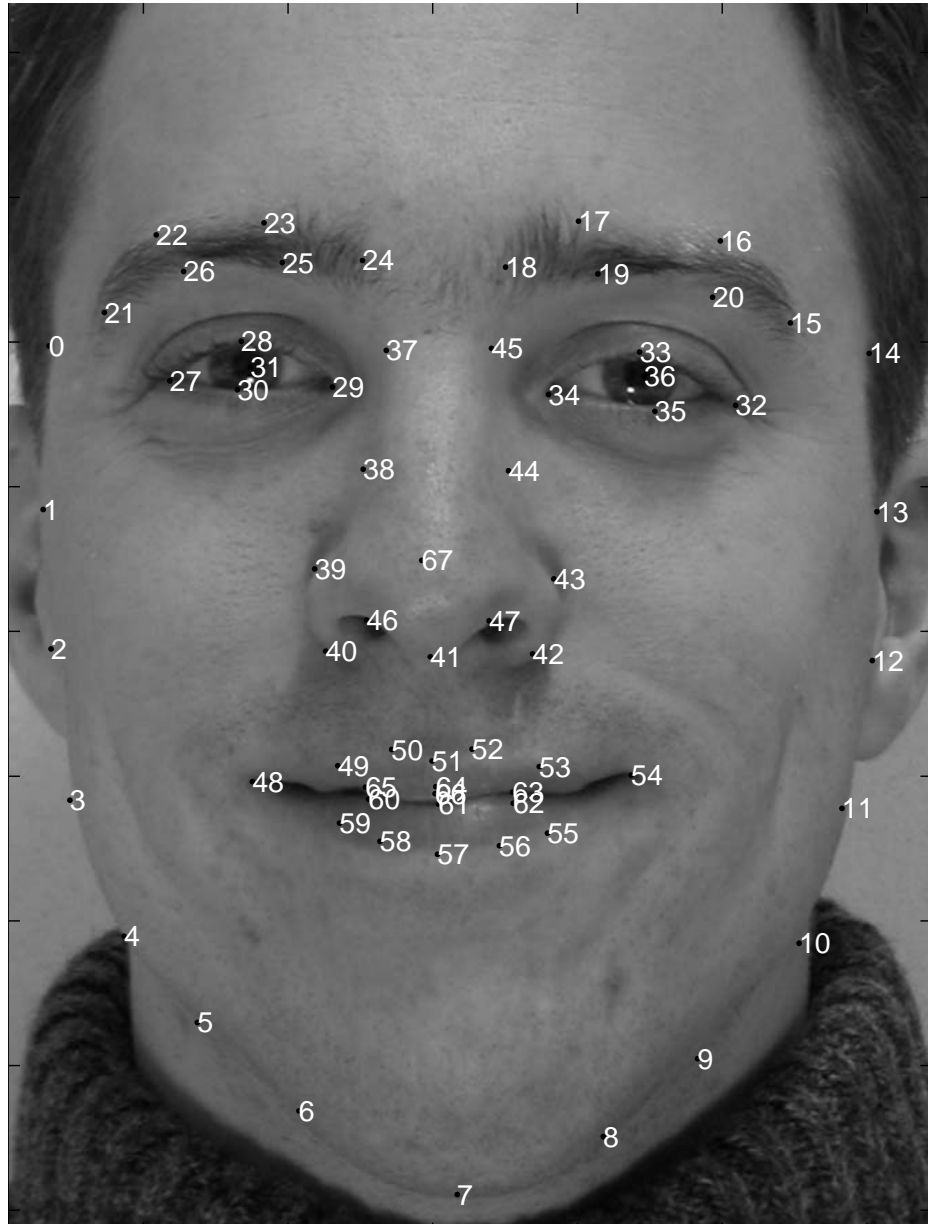


Fig. 1.1: Example of a face annotated using a model based landmark tracker.

The existing metrics for the evaluation of model based approaches are often found to be non-invariant. For example, sometimes accuracy in detection of landmarks is expressed solely by the use of pictures showing the fitting performances [22]. Since it is possible to find a few good results in a poorly performing system or vice versa, this kind of evaluation metric is not informative enough. Moreover, in order to measure accuracy in landmark prediction, Root Mean Squared (RMS) error is commonly used [23, 20]. Note that, it is possible to generate a low RMS error with low resolution images compared to a high resolution image while having the same content. Without explicitly mentioning resolutions of all the images such measures fail to maintain consistency in interpretation.

Moreover, evaluation in current approaches is performed only at the last level based on the final predictions made by the system. Although such strategies are simple and indicative of overall performance, but it fails to localize the sources of inefficiencies and hence deemed not to be concise enough.

1.3 Research Objectives

The objective of this research is to study the current model based approaches for facial expression recognition and to propose a method for analyzing their strengths and weaknesses. For effective comparison among these systems it is necessary to use an evaluation model which is invariant to the context of experiments. In other words, a general conceptual evaluation approach has to be decided which is applicable to any of the model based facial expression recognition techniques. The metrics used in such approaches have to be invariant and concise. This research is intended to analyze and propose solutions for all the problems encountered in effective evaluation among several candidate approaches.

1.4 Challenges

Evaluation and comparison of different approaches in a dynamically evolving field is a difficult task. The first challenge that comes into picture is the

non-comparable principles of working. It often happens that evaluation parameters are designed on certain assumptions which are not valid in some other approaches. This prohibits a standard one size fits all kinds of evaluation parameters.

Sometimes differences in implementation prohibit proper comparison of systems. Since many model based systems are not freely available and there are not enough baseline codes to compare with, it is very difficult for performance evaluation of such systems. Moreover, the databases used for evaluating model based facial expression recognition systems lack any standard scheme for annotation of ground truth data. As a result, it becomes hard to compare a system with more than one database.

1.5 Overview of the Following Chapters

The next chapter reviews works related to this thesis. Evaluation of the model based research works are discussed in the chapter. Going through these works will help better understand this thesis. Moreover, this chapter also discusses the works where some evaluation has been done in a wrong procedure. Chapter 3 will discuss about the theoretical knowledge necessary to understand this work. Otherwise explicitly mentioned, only the fundamental contributions of the author are discussed in this chapter. Other theories are either discussed very briefly or referred to their original works. Chapter 4 discusses the experimental setup. The first section in this chapter discusses the datasets used. The next section discusses the process of annotating the images for ground truth. Sec. 4.2 discusses the setup implemented in order to perform different experiments. It also discusses the methods of extracting different features. In Chapter 5, various experiments and their outcomes are described. Finally, chapter 6 concludes the thesis by discussing the contributions of this work.

Chapter 2

Related Literature

Model based approaches are gaining momentum since the seminal work of Lanitis et al, [23] on the recognition of facial expressions through ASM [12]. Major contributions in model based representation of face are shown in Fig. 2.1. ASM can detect shapes of deformable objects and this capability was rendered useful in facial expression research. Later on, several techniques were proposed similar to ASM. Active Appearance Model (AAM) [15] could account for not only the shape of deformable objects but also its texture. AAM algorithm was improved for faster convergence by Matthews et al. [20]. They used an elegant image registration technique known as Inverse Compositional Image Alignment [24, 25]. As a result it was possible to be used in real time videos. ICIA based AAM is used in several applications including detection of pain [6, 26, 27], depression [7] etc.

Although it was possible to use in determining pain and depression, a more general use of AAM was prohibited due to its dependency on a specific subject [28]. In the mean time some more model based approaches [29, 30, 17, 31] were proposed which claimed improvements like robustness in illuminations, faster and less complex registration algorithms etc. In 2008, Wang et al. converted the model based registration algorithm as a convex optimization algorithm [32]. This brought a huge increase in the performance of model based techniques. Later, in 2009 and 2010, through a series of their works, Saragih et al. proposed a technique known as Constrained Local Model (CLM) [33, 19, 34, 16] which is actually a generalized version encompassing all the model based techniques till 2010. CLM was found to be person independent [35] which solved the problem associated with AAM.

The methods used to evaluate the performance of these works were not always thoughtfully chosen to be invariant and concise. For example some works used RMS error without any scale normalization [23, 20]. Although more recent works

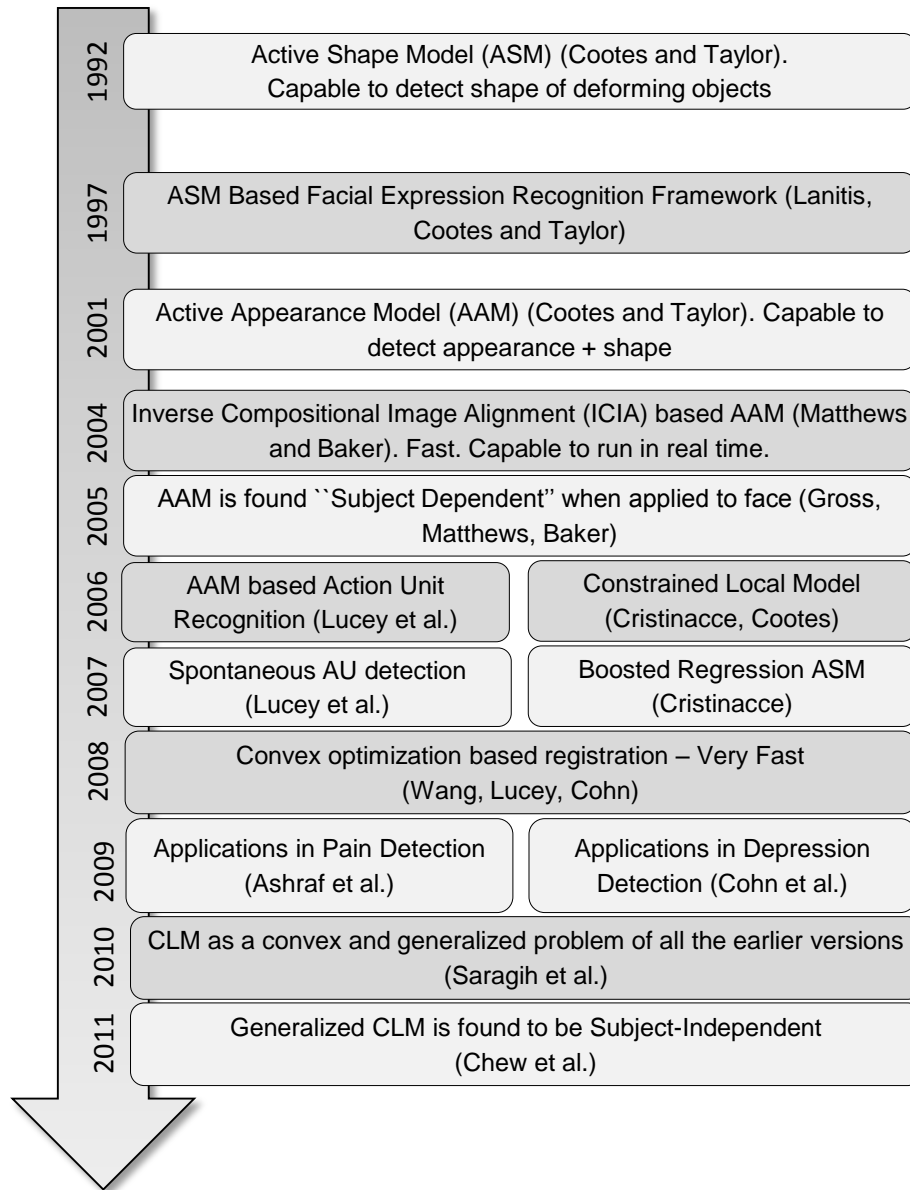


Fig. 2.1: Chronological flow diagram of some important works related to model based analysis of face.

have adopted a scale normalized error metric [17, 36], it is inconvenient because of the normalization factor (such as the distance between two pupils) is difficult to determine when parts of face is self-occluded.

Moreover, the facial expression and emotion detection techniques are often evaluated only based on the prediction performance of the machine learning techniques involved [37, 38, 7, 39, 6, 26, 27, 35]. Even works that were intended to define baseline performance of the systems [40] were also used only classification performance as evaluation metric. A better approach is described in this thesis which is termed as Three Level Evaluation Model. Moreover, some invariant and concise metrics are proposed for better comparison of the systems.

Chapter 3

Theoretical Background

3.1 Three Level Evaluation Model

A typical model based Action Unit (AU) recognition system [37, 38, 26] has the following conceptual blocks – landmark tracker, feature extractor and a classifier. A landmark tracker is a system that detects important points of a deformable objects (such as the face) based on a mathematical model of the object. These tracked landmarks are used to extract some appearance and morphological (shape) features. Details of extracting shape and appearance features are described in [37, 38, 26]. In the last level, a classifier is used to provide appropriate class labels to the images based on extracted features. Classically the performances of all the stages were measured only by evaluating the predictions given by the last stage as shown in Fig. 3.1(a).

As indicated in the introduction, this kind of evaluation model provides an indication only about the overall performance of the system. It cannot provide sufficient information to identify which block is performing poorly only by observing the classifier performance. Moreover, the overall performance can be affected by numerous factors coming from different blocks. As a result it becomes difficult to conclude well on the strengths and weaknesses of a system. In short, it lacks enough granularities in the evaluation process.

In order to better evaluate a system, a Three Level Evaluation (TLE) model is proposed so that it becomes possible to evaluate the performances of all the blocks rather than only the classification block. Such method provides much granularity and makes it possible to detect the sources of inefficiencies. For example, the landmarks tracked using a landmark tracker can be evaluated using ground truth locations provided with image databases. Landmark error measuring parameters are already available in current literature. The following section discusses some of these error

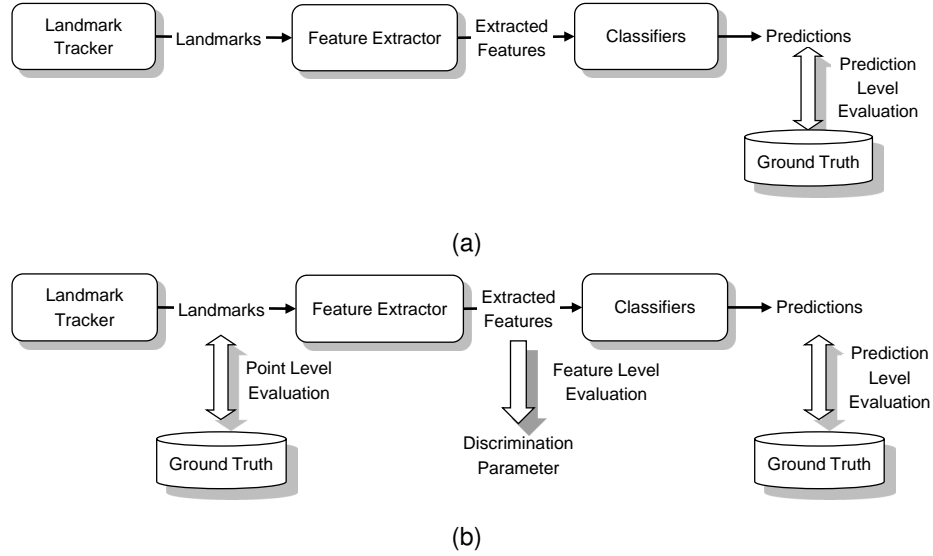


Fig. 3.1: A typical block diagram of a facial expression detection system. (a) shows the current evaluation approach which is dependent on the last block only. (b) illustrates the proposed approach which evaluates the outcomes of all the blocks.

metrics and problems associated with them. Also, a new metric will be defined to evaluate the ability of the features to discriminate among different classes. Lastly, the classical performance measuring techniques are used for evaluating the classification predictions.

3.2 Normalized Root Mean Squared Point Error (NRMS-PE)

In many of the recent works on model based representations of deformable objects (like ASM, AAM and CLM) RMS distance between ground truth and detected points is used as a performance evaluation parameter of landmark detectors [20, 23]. However, it is possible to show that this parameter is not invariant on image resolution. Due to such non-invariant properties, when this parameter is used to represent detection performance on two sets of images with identical content but different resolution, the values of the parameter vary significantly for these two datasets. This makes the parameter inconsistent and prohibits comparison of landmark detection systems without explicitly mentioning the image resolutions.

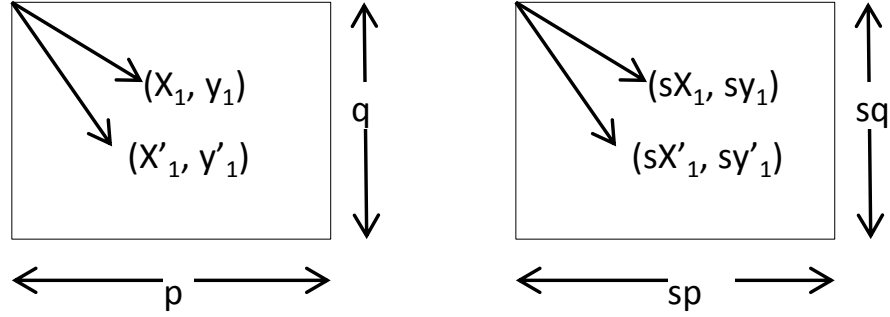


Fig. 3.2: Effect of proportionate scaling.

In order to show the dependency of the parameter on image resolution, let us consider a case where the coordinates of some landmark points are predicted to be $(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)$ where the true coordinates are $(x'_1, y'_1), (x'_2, y'_2) \dots (x'_n, y'_n)$. Also, the image width and height is assumed to be p and q pixels respectively. Root Mean Square distance between the predicted points and the ground truth is defined by (3.1).

$$d_{RMS} = \sqrt{\frac{\sum_{i=1}^n \{(x_i - x'_i)^2 + (y_i - y'_i)^2\}}{n}} \quad (3.1)$$

Now let us assume that, in another image the same content has been represented using s times the original image resolution as shown in Fig. 3.2. Therefore, the landmark and the predicted points will also be scaled s times their values in the earlier image. RMS distance for the new image will be as shown in (3.3). From this equation it is clear that RMS distance increases linearly with scale. It will produce different values for images with different resolution even if the image content is similar.

$$d_{RMS,Scaled} \tag{3.2}$$

$$= \sqrt{\frac{\sum_{i=1}^n \{(sx_i - sx'_i)^2 + (sy_i - sy'_i)^2\}}{n}}$$

$$= \sqrt{\frac{s^2 \sum_{i=1}^n \{(x_i - x'_i)^2 + (y_i - y'_i)^2\}}{n}}$$

$$= s \sqrt{\frac{\sum_{i=1}^n \{(x_i - x'_i)^2 + (y_i - y'_i)^2\}}{n}}$$

$$= sd_{RMS} \tag{3.3}$$

Some works in literature have used a parameter which represents the RMS distance as a fraction of some physiological measurements like width of face, inter-ocular length (i.e. the length between the pupil of two eyes) etc. [17, 36]. In these parameters, the effect of proportionate change in resolution gets canceled as shown in (3.4).

Where (x_r, y_r) and (x_l, y_l) represents the coordinates of pupils. It is clear from the equation that this parameter is not dependent on scale variation. However, it has two significant drawbacks. Firstly, the normalization does not take place if the horizontal and vertical axes are scaled differently. Secondly, the face width, inter-ocular distance (IOD) etc. may vary person to person and may be impossible to calculate when parts of the face are occluded due to head rotation.

A better approach might be normalizing the horizontal and vertical coordinates of the landmark points separately and expressing the errors as a fraction of RMS distance of landmarks from their centroid. Normalizing the coordinates separately takes care of the scaling problem that solves the first drawback mentioned in the previous paragraph. On the other hand, expressing errors with respect to the RMS

distance from centroid (RMSD) is more convenient than inter-ocular distance because some landmarks will be in frame even when an eye is occluded.

$$\begin{aligned}
& d_{RMS,Scaled,IOD} \\
&= \frac{1}{\sqrt{(sx_r - sx_l)^2 + (sy_r - sy_l)^2}} \sqrt{\frac{\sum_{i=1}^n \{(sx_i - sx'_i)^2 + (sy_i - sy'_i)^2\}}{n}} \\
&= \frac{s}{s\sqrt{(x_r - x_l)^2 + (y_r - y_l)^2}} \sqrt{\frac{\sum_{i=1}^n \{(x_i - x'_i)^2 + (y_i - y'_i)^2\}}{n}} \\
&= \frac{1}{\sqrt{(x_r - x_l)^2 + (y_r - y_l)^2}} \sqrt{\frac{\sum_{i=1}^n \{(x_i - x'_i)^2 + (y_i - y'_i)^2\}}{n}} \tag{3.4}
\end{aligned}$$

In such case both the x and y components of the coordinates should be scaled in such a way that the RMS distance from their respective means be equal to $\frac{1}{\sqrt{2}}$.

$$\begin{aligned}
\frac{1}{s'_x} \sqrt{\frac{\sum_{i=1}^n (x'_i - \mu'_x)^2}{n}} &= \frac{1}{\sqrt{2}} \\
\Rightarrow s'_x &= \sqrt{\frac{2 \sum_{i=1}^n (x'_i - \mu'_x)^2}{n}} \tag{3.5}
\end{aligned}$$

Therefore, the scaling factors for x and y components will be

$s'_x = \sqrt{\frac{2 \sum_{i=1}^n (x'_i - \mu'_x)^2}{n}}$ and $s'_y = \sqrt{\frac{2 \sum_{i=1}^n (y'_i - \mu'_y)^2}{n}}$ respectively. Where μ'_x and μ'_y are the mean of x and y coordinates of the ground truth points. Now let us normalize the ground truths by shifting the centroid to origin and scaling the axes so that the RMSD of normalized ground truth points be equal to one. In other words, x and y can be normalized using (3.6) and (3.7).

$$x'_{i,norm} \leftarrow \frac{1}{S'_x} (x'_i - \mu'_x) \quad (3.6)$$

$$y'_{i,norm} \leftarrow \frac{1}{S'_y} (y'_i - \mu'_y) \quad (3.7)$$

Algorithm 1: Algorithm to Calculate Normalized Root Mean Squared Point Error

Input: Predicted Points: $(x_1, y_1), (x_2, y_2) \dots (x_n, y_n)$ and True Points:
 $(x'_1, y'_1), (x'_2, y'_2) \dots (x'_n, y'_n)$

Output: Normalized Root Mean Squared Point Error

```

1 begin
2    $\mu'_x \leftarrow \frac{1}{n} \sum_{i=1}^n x'_i$ 
3    $\mu'_y \leftarrow \frac{1}{n} \sum_{i=1}^n y'_i$ 
4    $s'_x \leftarrow \sqrt{\frac{2 \sum_{i=1}^n (x'_i - \mu'_x)^2}{n}}$ 
5    $s'_y \leftarrow \sqrt{\frac{2 \sum_{i=1}^n (y'_i - \mu'_y)^2}{n}}$ 
6   for  $i \in \{1, 2, 3, \dots, n\}$  do
7      $x'_{i,norm} \leftarrow \frac{1}{s'_x} (x'_i - \mu'_x)$ 
8      $y'_{i,norm} \leftarrow \frac{1}{s'_y} (y'_i - \mu'_y)$ 
9      $x_{i,norm} \leftarrow \frac{1}{s'_x} (x_i - \mu'_x)$ 
10     $y_{i,norm} \leftarrow \frac{1}{s'_y} (y_i - \mu'_y)$ 
11     $d_{NRMS-PE} = \sqrt{\frac{\sum_{i=1}^n \{(x_{i,norm} - x'_{i,norm})^2 + (y_{i,norm} - y'_{i,norm})^2\}}{n}}$ 
12 end

```

Algorithm 1 describes the process of normalizing and calculating the error metric. The x and y coordinates of the centroid of ground truth landmark points has been calculated in lines 2 and 3 respectively. When these values are subtracted from each landmark points, the whole shape is translated to make the centroid to be located in origin. In lines 4 and 5, the appropriate scaling factor is calculated which is used to normalize the ground truth points so that the RMS distance of all the translated and

scaled (i.e. normalized) ground truth points from their origin becomes one. This can be proved using (3.8). One point is to be noted about the algorithm is it translates and scales the predicted points using the same parameters as it does with the ground truth points. This ensures no error is introduced in the translation and scaling process.

Therefore, the Normalized Root Mean Squared Point Error (NRMS-PE) $d_{NRMS-PE}$ is actually the normalized detection error expressed in terms of the RMS distance of ground truth points from origin. It is more convenient to use than the IOD based metric because it is possible to calculate this metric even if some parts of the face are occluded.

$$\begin{aligned}
& RMSD'_{Norm} \\
&= \sqrt{\frac{\sum_{i=1}^n \left\{ \left(\frac{x'_i - \mu'_x}{s'_x} \right)^2 + \left(\frac{y'_i - \mu'_y}{s'_y} \right)^2 \right\}}{n}} \\
&= \sqrt{\frac{\frac{1}{s'^2_x} \sum_{i=1}^n \{ (x'_i - \mu'_x)^2 \}}{n} + \frac{\frac{1}{s'^2_y} \sum_{i=1}^n \{ (y'_i - \mu'_y)^2 \}}{n}}{n}} \\
&= \sqrt{\frac{s'^2_x}{2s'^2_x} + \frac{s'^2_y}{2s'^2_y}} \\
&= \sqrt{\frac{1}{2} + \frac{1}{2}} \\
&= 1
\end{aligned} \tag{3.8}$$

3.3 Parameter of Discrimination

For feature level evaluation it is necessary to decide the discrimination power of features. By the term “discrimination power” it is implied that how good a set of features in discriminating between two discrete classes of data. In this work, a metric has been proposed to determine such capabilities of features termed as the “*Parameter of Discrimination*”. In order to do this, Deng Cai’s implementation [41] of

Linear Discriminant Analysis (LDA)[42] has been used. Using LDA it is possible to get a set of eigenvectors. The eigenvectors are oriented in such a way that if the data is projected onto these eigenvectors, the projections from two different classes will be maximally separated. This concept is demonstrated using Fig. 3.3.

Now let us assume that \mathbf{F}_1 and \mathbf{F}_2 ¹ are two sets of high dimensional features. Also let us denote the projections of \mathbf{F}_1 and \mathbf{F}_2 on the maximally separating eigenvector by the operator $\wp(\cdot)$. That is, if \mathbf{E}_1 be the maximally separating eigenvector then,

$$\wp(\mathbf{F}_i) = \frac{\mathbf{E}_1^T \mathbf{F}_i}{\|\mathbf{E}_1\|}; \quad \forall i = \{1, 2\} \quad (3.9)$$

It is possible to propose a parameter of discrimination. We know that for a normal distribution, only 0.1% of the data lies beyond the 3rd standard deviation from mean in each side as shown in Fig. 3.4.

Therefore, if we want two Gaussian distributions to be separated in such a way that they have less than 0.1% overlap, then the distance of their mean should be equal to or greater than three times the summation of their standard deviations as shown in Fig. 3.5. Therefore a parameter of discrimination \mathcal{D} can be define as shown in (3.10) so that the amount of overlap becomes less than 0.1% which is very small.

$$\begin{aligned} & |\mu(\wp(\mathbf{F}_1)) - \mu(\wp(\mathbf{F}_2))| && \geq 3(\sigma(\wp(\mathbf{F}_1)) + \sigma(\wp(\mathbf{F}_2))) \\ \Rightarrow & \frac{|\mu(\wp(\mathbf{F}_1)) - \mu(\wp(\mathbf{F}_2))|}{3(\sigma(\wp(\mathbf{F}_1)) + \sigma(\wp(\mathbf{F}_2)))} && \geq 1 \\ \Rightarrow & \mathcal{D} && \geq 1 \end{aligned} \quad (3.10)$$

¹Bold and capital letters denote a matrix. Bold and small letter denotes a vector and non-bold small letters denote a scalar

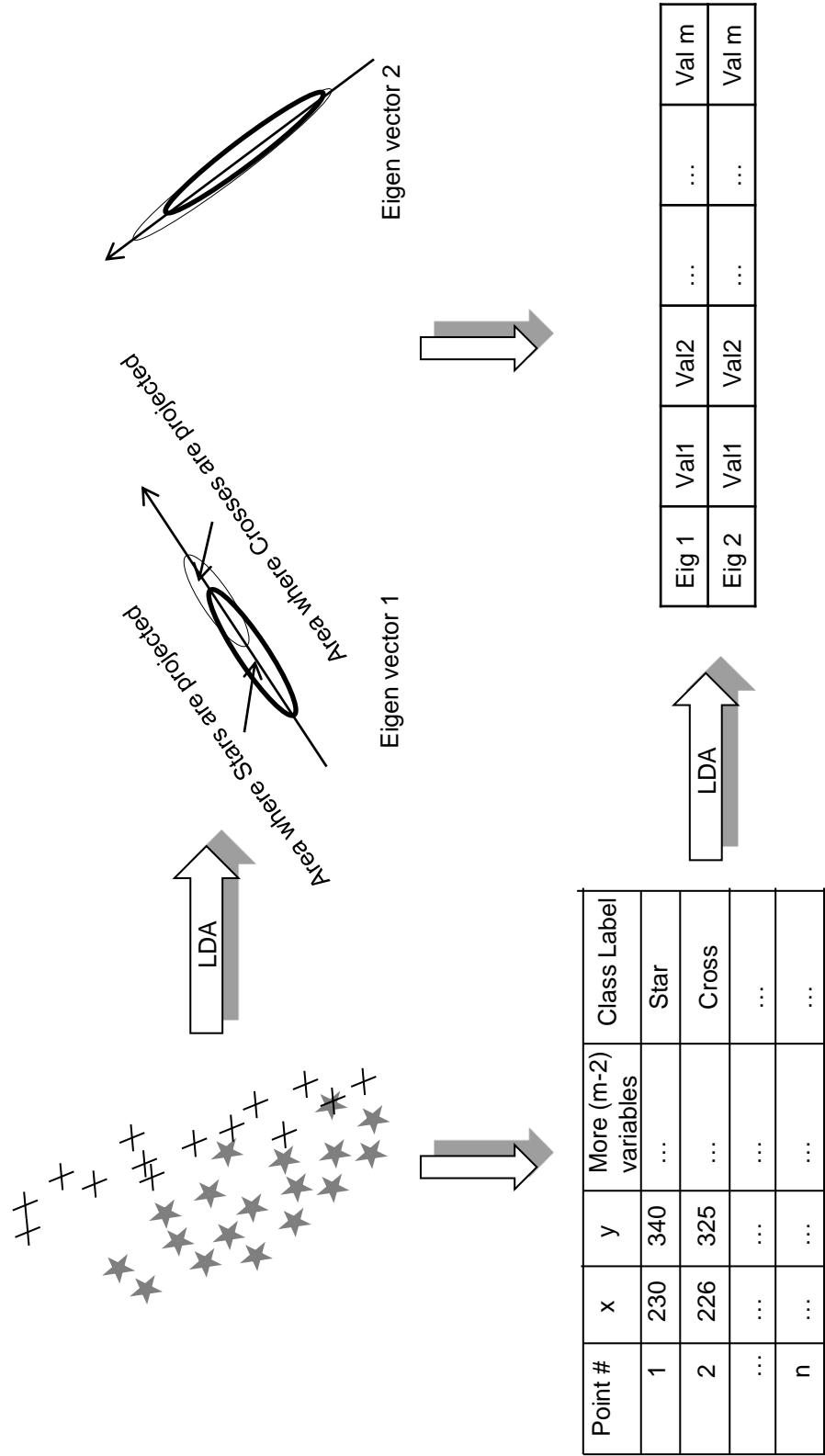


Fig. 3.3: Demonstration of the basic idea of Linear Discriminant Analysis (LDA). The stars and the crosses are representatives of high (m) dimensional data. LDA algorithm returns a set of orthogonal eigenvectors. If the stars and crosses are projected on the first eigenvector, it will maximally separate the projections. The second eigenvector will provide a separation lesser than the first one and so on. Dimensions of the eigenvectors are equal to the number of variables used to constitute a data point.

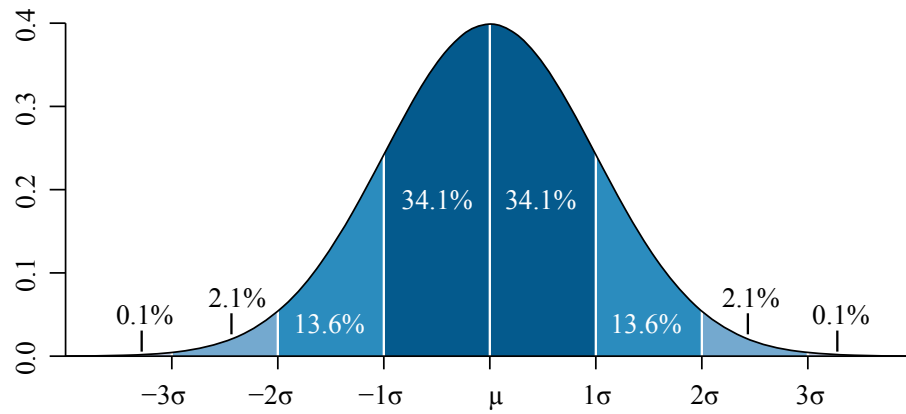


Fig. 3.4: Probability Density Function of Normal Distribution with Standard Deviations [1].

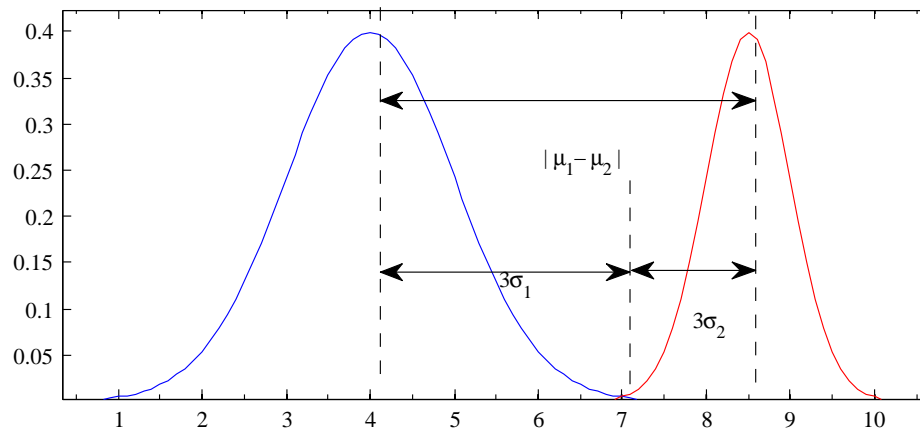


Fig. 3.5: Distances between two normal distributions.

Here, the operators $\mu()$ and $\sigma()$ represents mean and standard deviation respectively. Therefore, the parameter of discrimination between two classes of features is defined as below.

$$\mathcal{D} = \frac{|\mu(\varphi(\mathbf{F}_1)) - \mu(\varphi(\mathbf{F}_2))|}{3(\sigma(\varphi(\mathbf{F}_1)) + \sigma(\varphi(\mathbf{F}_2)))} \quad (3.11)$$

The parameter of discrimination, \mathcal{D} will be greater than or equal to one when \mathbf{F}_1 and \mathbf{F}_2 is perfectly Gaussian and has less than 0.1% overlap among their projections on maximally separating line. However, it can also be applied to distributions that are not Gaussian. In such cases, its value of unity might refer to some other amount of overlap than the one described here.

3.4 Normalizing the Differences among the Ground Truth Schemes

While comparing various implementations of landmark detectors with each other using some standard dataset, it is a common phenomenon that the number and positions of the chosen landmarks do not match over different datasets. This happens because the ground truth annotated datasets were developed in a scattered way to serve a particular interest and later on released to public for further use. Since no standard annotation scheme is in place, the datasets became incompatible to each other. For example, the tracker used in this work [2] is trained on Multi-PIE database [43]. Therefore it detects the landmark points according to the annotation scheme of Multi-PIE where 66 landmark points are detected.

On the other hand, the landmarks in Extended Cohn Kanade Database (CK+) are annotated using a different scheme which uses 68 landmark points. The lower part of the upper lip and the upper part of the lower lip is represented by two different numbers of landmark points in these two databases. As a result of this incompatibility, it becomes difficult to match the landmarks tracked by the tracker with the ground truth landmarks provided in CK+ dataset. This problem can be very difficult to solve in

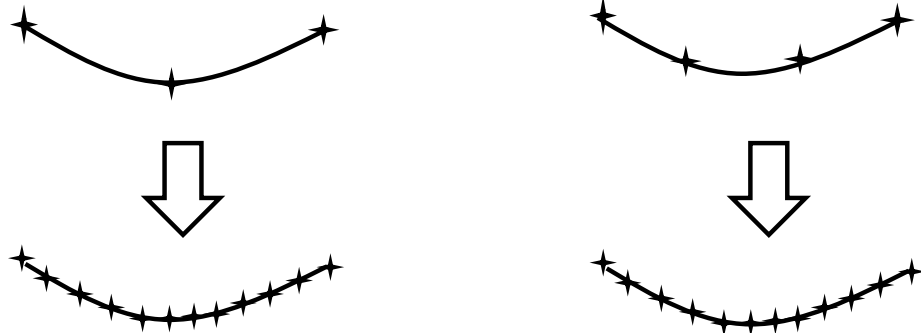


Fig. 3.6: Process of up-sampling the number of landmarks in order to normalize two non-compatible annotation schemes.

severe cases. However, in this case, it has been solved by re-sampling both annotation schemes to a common higher value. For example, let us consider the Fig. 3.6 where a path is represented using 3 points and 4 points respectively. In order to compare among those two sets of points it is needed to assume that only the beginning and the end points of the path corresponds to one another. Then the rest of the points in the path are interpolated in order to increase the number of samples to the least common multiple of the earlier number of samples (i.e. $LCM(3,4) = 12$). These up-sampled points are compared with each other. However, this method will not work if the initial points do not match.

Chapter 4

Methodologies

4.1 Databases

4.1.1 Extended Cohn Kanade Database

The Extended Cohn Kanade (CK+) database [40] is an updated version of Cohn Kanade Database [44]. It consists of 593 sequences from 210 adults posing different emotions. The images consist of a significant amount of diversity in ethnic groups and gender. In each sequence, an emotion is posed from onset to apex. The name of the action units occurred in the apex frame and the emotion expressed in each sequence are provided as ground truth. Also, 68 predefined landmarks locations in each picture of face are given. A total of 30 action units (AU) are annotated in the CK+ database. However, some of these are occurred in a very small number of images. In this work, only 21 AUs are chosen considering a minimum of twenty positive sample images per AU.

4.1.2 MMI Database

The MMI database for facial expression [45] is a web based [46] database for analyzing facial expressions. It is a continuously growing database with periodic accumulation of new video data and meta information. The database is divided into several parts among which the first part consists of 1767 clips from 20 participants showing fully synchronized frontal and profile display of several action units (AUs) and

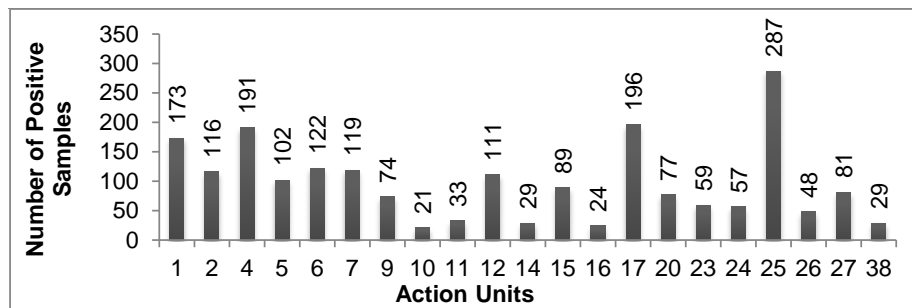


Fig. 4.1: Number of Images from Extended Cohn Kanade database for different Action Units used in this work.

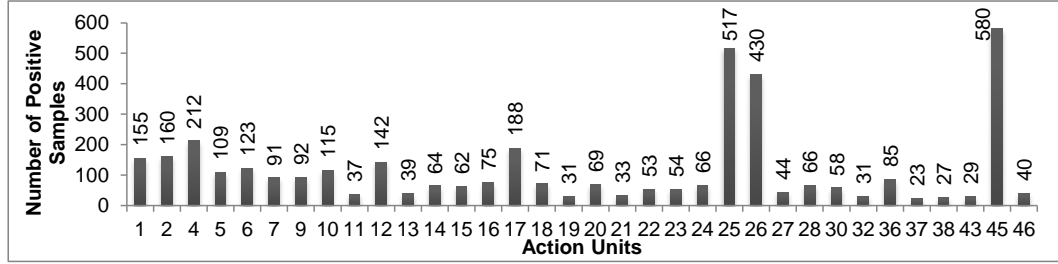


Fig. 4.2: Number of Images from MMI database for different Action Units used in this work.

action descriptors (ADs). It also contained several affective states along with the action units. In part two, there are 238 clips from 28 subjects showing six basic emotions. Part three includes high quality still images rather than video. It is comprised of 484 images of 5 subjects where all the AUs are displayed. Part four and five include videos that consist of spontaneous disgust, happiness and surprise emotion. There are different kinds of ground truth information associated with MMI database among which action unit annotations are also given. In this work, a total of 1374 action unit coded frontal images from the first three parts of the MMI database have been used. For video sequences, only the middle frame is considered. The images which contain synchronized profile picture with the frontal one, only the frontal picture is considered. The amount of positive samples for each AUs used in this work from MMI database is shown in Fig. 4.2

4.2 Experimental Setup

In order to discuss three level evaluation model, a simple action unit detection system is considered in this paper. Block diagram of the system is shown in Fig. 4.3. This system is intended to detect existence of several action units from pictures of face. Two image databases are used to train and test the system. These are Extended Cohn-Kanade (CK+) and MMI database. Descriptions of these databases are given in the previous section. The action unit detection system employs a facial landmark detector named FaceTracker [2]. FaceTracker is available online for download and to

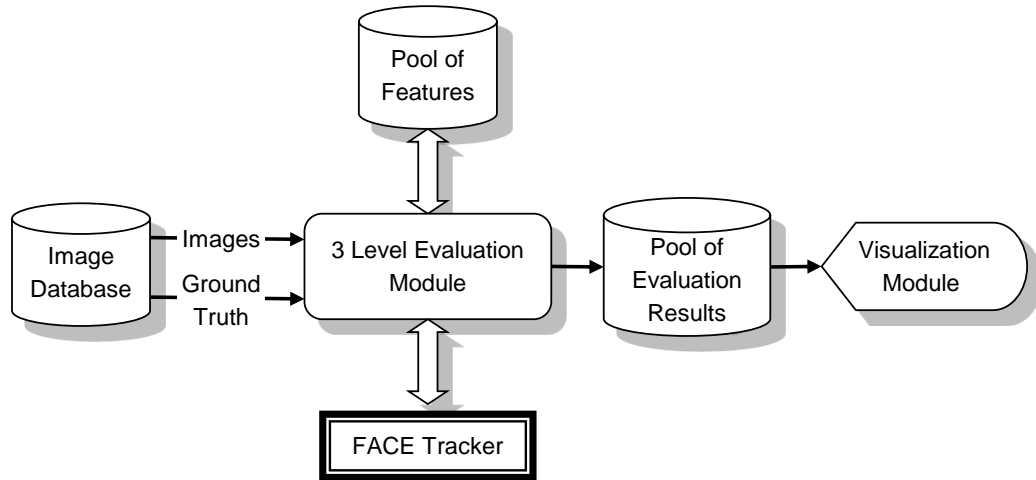


Fig. 4.3: Overall block diagram of the prototype system implemented in order to evaluate a simple action unit recognition system.

use in research purposes [34]. It is a Constrained Local Model (CLM) [19] based landmark detector which is trained on Multi-PIE database [43] and can detect 66 points on face. An example of the landmarks tracked by FaceTracker along with the defined triangulation is given in Fig. 4.4.

The system extracts normalized point features (SPTS) and canonical appearance features (CAPP) as described in the following section. These features are built based on the predictions of landmarks provided by the face tracker. A hybrid of the point and appearance features is also calculated. Hybridization is done just by concatenating the SPTS and CAPP features together. A dimensionality reduced version of the hybrid features is also evaluated for its performance in this system. Dimensionality is reduced by the use of Linear Discriminant Analysis (LDA)[42] as implemented by Deng Cai [41]. All these features and the associated class labels are stored in the pool of features as Comma Separated Value (CSV) files. A set of binary classifiers are used to detect the action units. Each classifier detects only one action unit by providing a positive or negative output. This system is evaluated in a three level evaluation module which is described using Fig. 4.5.



Fig. 4.4: Output given by FaceTracker.

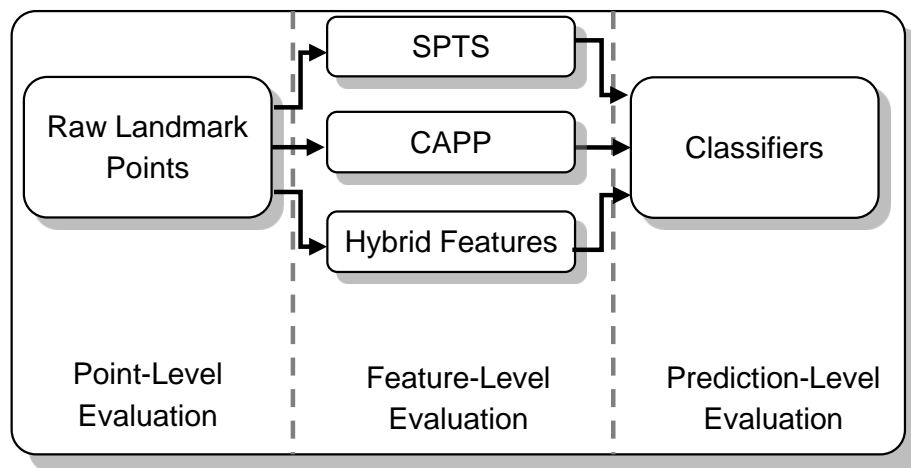


Fig. 4.5: Three Level Evaluation Module.

The point level evaluation is done on the raw points detected by the face tracker. A scale normalized evaluation parameter named “NRMS-PE” is used for this purpose. This parameter is invariant to different horizontal and vertical scaling and also convenient to use when parts of face is self-occluded due to head rotation. Details of this parameter are discussed in Chapter 3. For evaluating the goodness of extracted features, another parameter is constructed. As discussed in Chapter 3, this parameter is based on the ability of the features to discriminate among two discrete classes. It is used to evaluate the system in feature level.

Lastly, in the prediction level, the classifier predictions are evaluated for its performance. Several classifier performance measuring parameters exist in current literature. For example, in this work, the area under ROC (Receiver Operating Characteristics) is used as an evaluation parameter for classifier prediction.

4.2.1 Standardized Points Features (SPTS)

Landmark points of the face actually form a shape that is deformable and changes its pattern with different facial identity and expressions. A face-tracker detects the x and y coordinates of these landmark points in picture of face assuming the top left corner as origin and each pixel as unit distance. Suppose $\mathbf{X} = (x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ be the shape formed by the landmark points detected by the face-tracker. Since the size and location of face in the picture can be varied, therefore, for a good comparison of landmarks over several images it is necessary to “standardize” all the shape points [37]. In the process of standardization, the global similarity transformations (i.e. translation, scale and rotation) has to be removed.

In order to remove translation, the shape is centered onto the origin. This is done by subtracting the centroid of the landmarks from the coordinate location of each point. This is shown in Fig. 4.6. The centroid is obtained by averaging all the points in shape as shown below in 4.1.

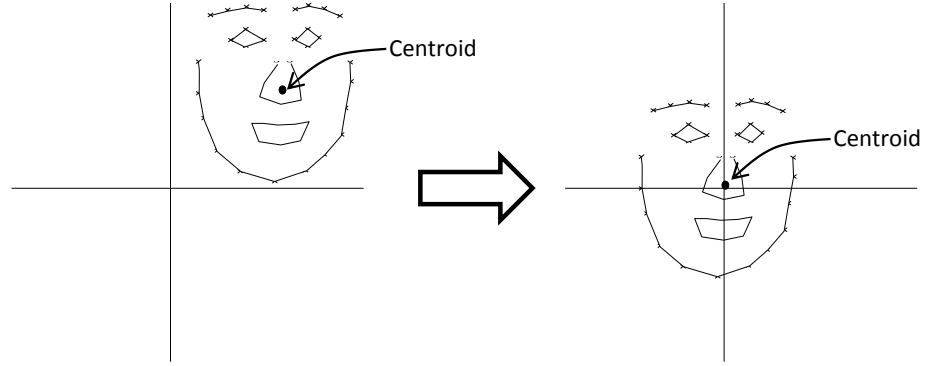


Fig. 4.6: Translating to origin [Note: the lines are drawn for illustration purpose].

$$\mathbf{X}_{COG} = \frac{1}{N} (\mathbf{X}_1 + \mathbf{X}_2 + \dots + \mathbf{X}_N) \quad (4.1)$$

For scale normalization the shape is rescaled so that the norm of shape becomes one. Norm of shape is defined as the Root Mean Squared (RMS) distance of each point from the centroid of the shape. Mathematically the RMS distance is defined by (4.2)

$$S = \sqrt{\frac{\sum_{i=1}^N (\mathbf{X}_i - \mathbf{X}_{COG})^2}{N}} \quad (4.2)$$

and it represents the scale of the shape. For normalizing the scale, each point is to be divided by the RMS distance of the shape as shown in (4.3)

$$\mathbf{X}_{norm} = \frac{1}{S} \{\mathbf{X}_i\}; \forall i = 1 \dots N \quad (4.3)$$

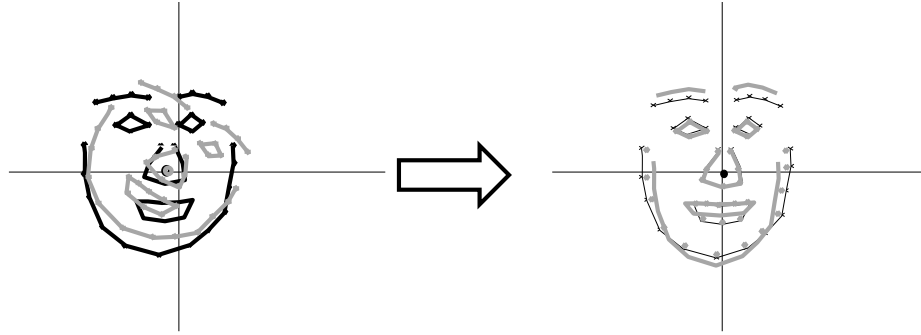


Fig. 4.7: Process of removing rotation transformation. The dark shape denotes a reference shape.

For removing the global rotation transformation, it is necessary to have a reference shape.

In this work, the mean of all the shapes is considered as reference. Rotation transformation is removed by aligning each shape with the reference. It is done in a process described in [47]. The aligning process does not superimpose the shapes completely because of the uniqueness of facial morphology and deformation due to facial expression. However, it results in similar orientation of the shapes. For more details of constructing SPTS please refer to [37] and [48].

4.2.2 Split Triangle Canonical Appearance Features (CAPPX)

Canonical Appearance (CAPP) features contain information about the texture or appearance of face. For better comparison of appearance from different face morphology and expressions, it is necessary to decouple the appearance from shape. This is done by warping the image in a canonical base form. A piecewise affine transformation with the help of a predetermined triangulation (as shown in Fig. 4.9(a)) is used to accomplish this work.

A triangulation is a collection of triangles formed by joining a number of points. The triangles are taken in such a way so that they do not overlap onto one another. For example, a Delaunay [49] triangulation may be used which is constructed by a

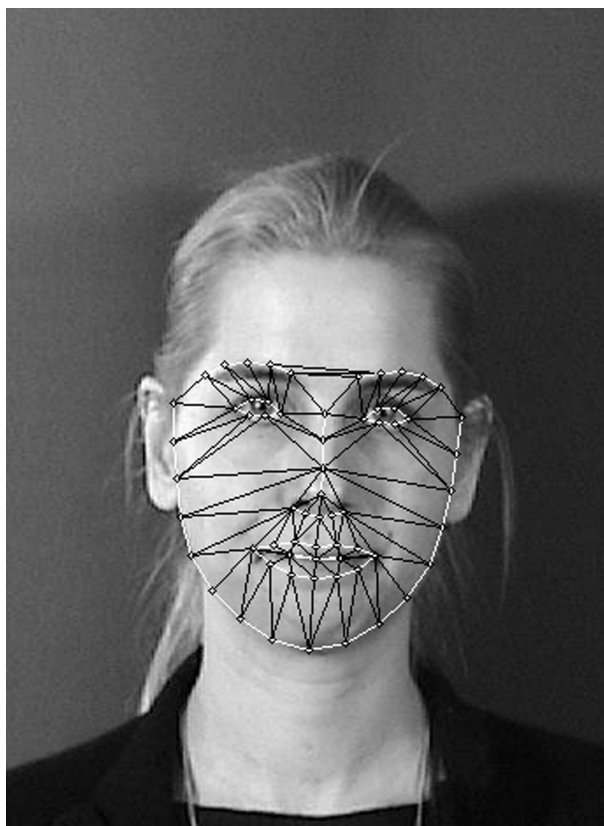


Fig. 4.8: Example of an image for which the process of PAW will be demonstrated.

constraint that no point lies inside the circum-circle of any triangle. Once a suitable triangulation for the reference shape is decided, it is applied on the points tracked on the face image. Then a Piecewise Affine Transformation (PAW) algorithm is applied to transform the content of each triangle in the image triangulation to the corresponding triangle in the base triangulation. The effect of PAW is shown in Fig. 4.8 and Fig. 4.9. Fig. 4.9(b) shows the image of face after PAW is applied to transform the image region from Fig. 4.8 to Fig. 4.9(b). Full detail of PAW is given in [48].

Once the appearance is warped into canonical base form, changes due to facial morphologies of different people are normalized. For minimizing the effect of illumination, histogram equalization [50, 51] algorithm is applied. Although this cannot normalize the local illumination variations due to the structure of face and position of

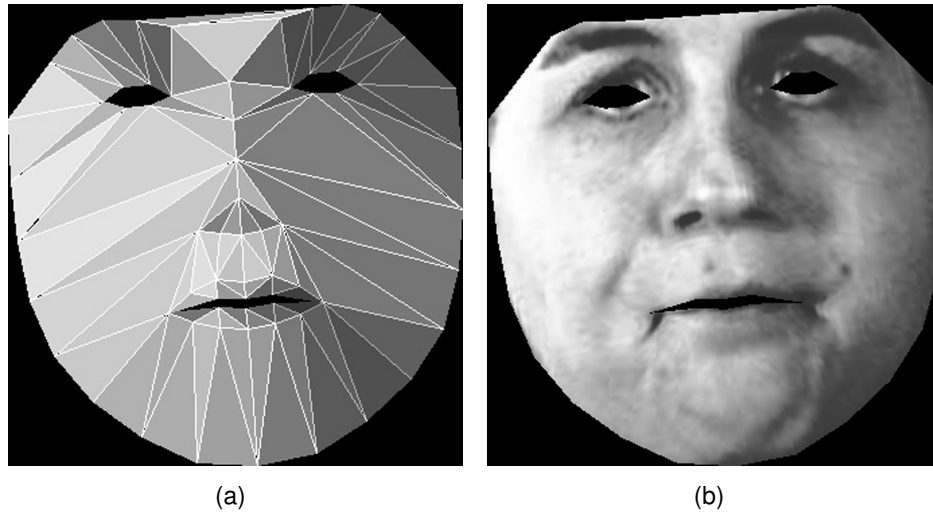


Fig. 4.9: The effect of Piecewise Affine Warping (PAW). The face region shown in Fig. 4.8 is warped into the reference base triangulation. The base triangulation is shown in (a). The resulting image is shown in (b).

light, it can effectively normalize the effect of global variation of brightness from picture to picture. Moreover, the image is converted to grayscale since the information corresponding to facial expression is not carried by skin color. Now, it is possible to use the pixel values as appearance feature which is used in [37]. However, this approach incorporates a large number of features. Using very high number of features is not desirable because that might introduce the phenomenon known as curse of dimensionality [52]. On the other hand, using too small of a number of features might miss significant information to discriminate between two discrete classes.

Therefore, in order to achieve optimal results, a method is needed through which it would be possible to control the total number of features. In this work this is done by splitting the larger triangles into smaller ones and considering the average value of pixels inside each triangle as a feature. The triangles are split iteratively. In each iteration, the triangle with largest area is split into three smaller triangles by considering the centroid as a new vertex. Once a specified number of splitting is done, the pixels inside a triangle are averaged to calculate a single valued feature for that

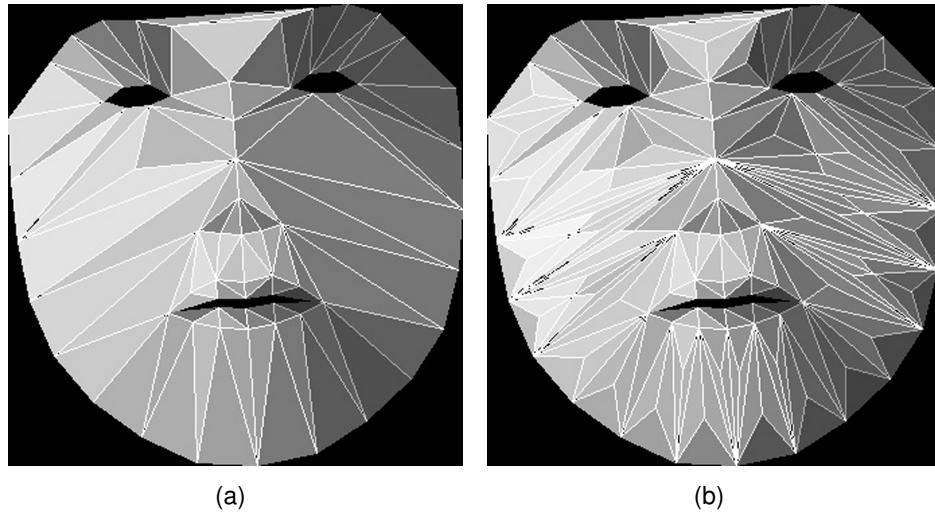


Fig. 4.10: Demonstration of splitting the triangles for calculating CAPP features. (a) CAPP91 (b) CAPP140.

particular triangle. In this process the number of triangles can be arbitrarily increased and in the asymptotic case the features become individual pixel values. Therefore, the traditional CAPP feature is a special instance of the appearance feature described here. In this thesis, appearance feature calculated using X number of triangles are called CAPPX (For example, CAPP90,CAPP150 etc.). Fig. 4.9(a) shows a CAPP90 feature because the base triangulation consists of 90 triangles. Each triangle in the figure are filled with a grayscale value corresponding to the feature value associated with that triangle. Fig. 4.10 shows examples of a CAPP91 and CAPP140 features. It is to be noted that the largest triangle is split first.

Once it is possible to control the total number of features, it becomes necessary to determine the amount of the features that will be optimum. The amount should be large enough to contain sufficient information to discriminate among different classes and at the same time, small enough to avoid curse of dimensionality. In this work, this is determined through the use of discrimination parameter as defined in Sec. 3.3. Since the appearance features are basically the averages of pixel intensities, according to central limit theorem they can be assumed to be normally

distributed. Although the parameter is defined assuming that the features are normally distributed, here it is used regardless of the normality assumption. In the next chapter values of discrimination factors for ideal case is determined through a series of experiments.

Chapter 5

Experiments and Results

5.1 Point Level Evaluation

5.1.1 Efficacy of NRMS-PE

An experiment was designed in order to evaluate the efficacy of the Normalized Root Mean Square Point Error (NRMS-PE). As stated earlier, it is a resolution invariant point tracking error measurement parameter. To check its resolution invariance property, landmark points were tracked in two different versions of a picture from the CK+ database. One version of the picture was represented using 640 x 490 pixels which is the original dimension of pictures from Extended Cohn Kanade (CK+) database. Another version was made by up-sampling the horizontal axis to 5 times its original size. Now the Root Mean Squared Point Error (RMS-PE) and the Normalized Root Mean Squared Point Error (NRMS-PE) was calculated using the tracked points and the ground truth points. For the up-sampled image ground truth was calculated by multiplying the horizontal coordinates by 5.

The RMS-PE and the NRMS-PE were measured for the several areas of face. The results are shown in Fig. 5.1. According to the figure, RMS-PE varies widely with image resolution while NRMS-PE remains same.

5.1.2 Cumulative Error Distribution (CED) Chart

Since NRMS-PE is scale invariant, it can be used for an invariant and concise representation of the fitting performance of the tracker. To do this Cumulative-Error-Distribution (CED) chart is a popular and useful method. The CED chart is computed for different areas of face in CK+ database as shown in Fig. 5.2. It is actually a plot of percentage of database versus Normalized Root Mean Squared Point Error (NRMS-PE). It essentially represents what fraction of the whole set of images in the database lie within a certain amount of error threshold. In this chart a significantly deteriorated performance for tracking of the lower lip is evident. This may

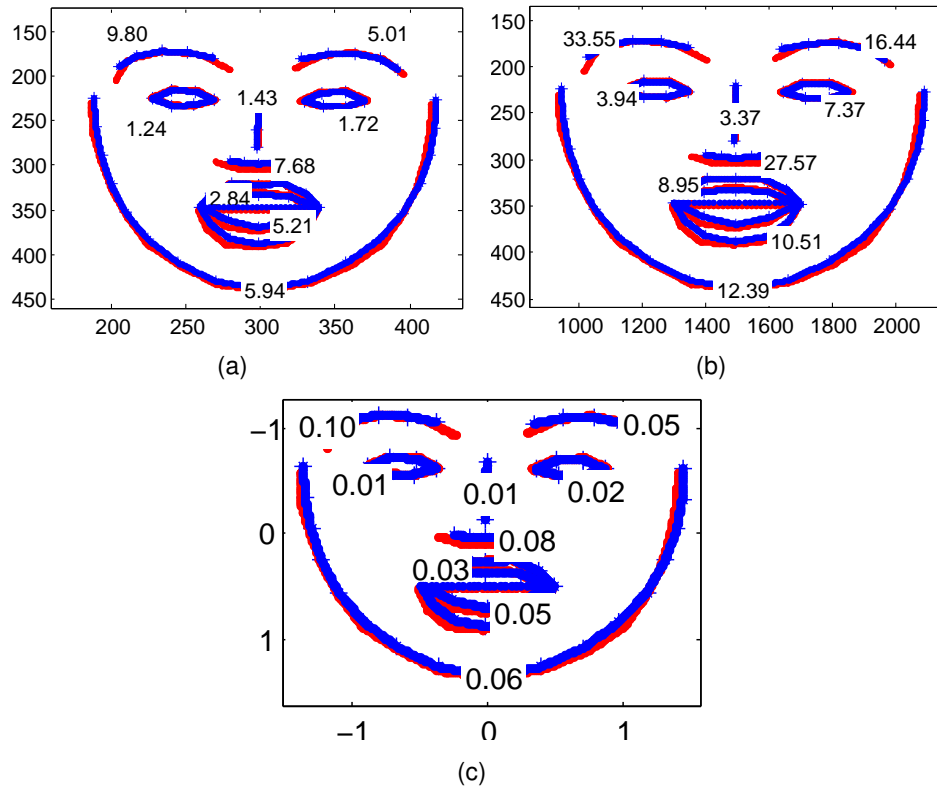


Fig. 5.1: (a) The shape formed by the landmarks of original image and corresponding RMS errors for different parts of face (b) RMS errors when the x axis is scaled 5 times (c) NRMS-PE is same for both the original and scaled versions.

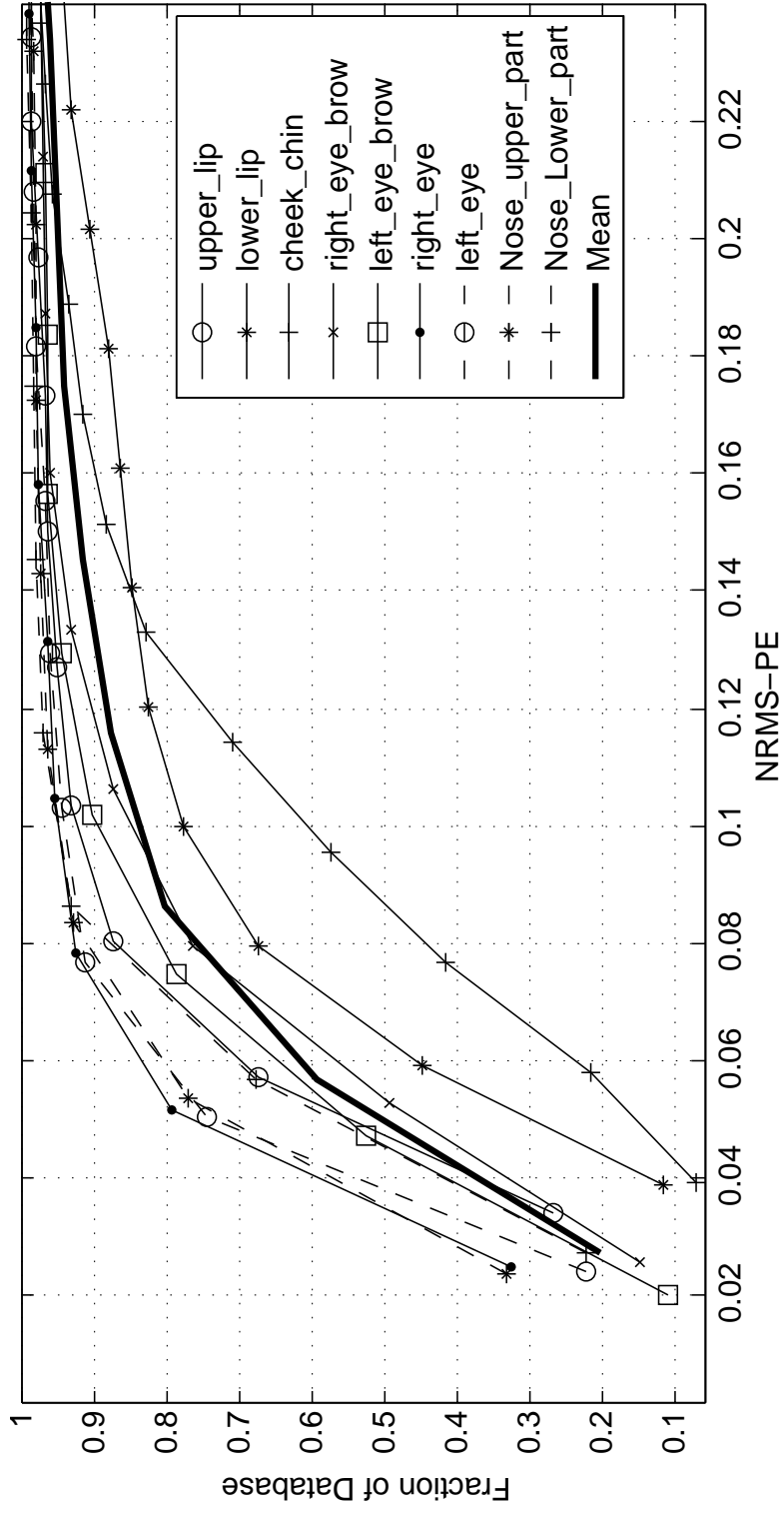


Fig. 5.2: Cumulative Error Distribution chart for the tracker employed in this work (FaceTracker [2]). The calculations were done in CK+ database.

be due to the abrupt changes in the appearance of mouth region when it is opened widely. Since the tracker is based on local models with limited capacities, they fail to account for when there is significant changes in appearance. Moreover, in some pictures of CK+ database the timestamps fell on to the face image which prohibited local detectors to detect the landmarks accurately. Some of these fitting inaccuracies are shown in Fig. 5.3.

5.2 Feature Level Evaluation

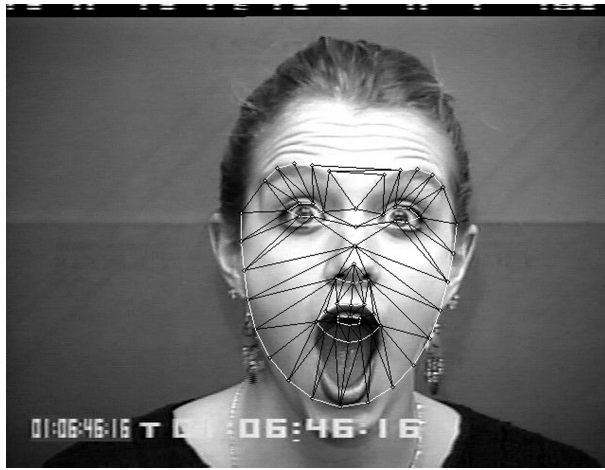
5.2.1 Efficacy of Parameter of Discrimination

MMI database was used in this experiment. 1374 videos and images were selected where different action units were acted out. In every video, the action units were expressed and then gradually diminished. From each of these videos only the middle frame was extracted. All the frames extracted were frontal faced. For every frame, the CAPPX features were extracted using the method discussed in Sec. 4.2.2. The set of features corresponding to a frame where a particular action unit is present was taken as a positive example of that action unit. Now, for each action unit, parameter of discrimination among the positive features and negative features was calculated. As it is discussed earlier Linear Discriminant Analysis (LDA) is used in the process of calculating parameter of discrimination (\mathcal{D}).

Fig. 5.4 shows the projection of the data on maximally separating eigenvector as well as the value of \mathcal{D} for action unit 10 (AU10). Along vertical axis some arbitrary zitter is incorporated for greater visibility of the data points. It can be noticed from the figures that the parameter of discrimination can successfully represent the amount of overlap on the line of maximum discrimination.

5.2.2 Relation between Number of Features and Parameter of Discrimination

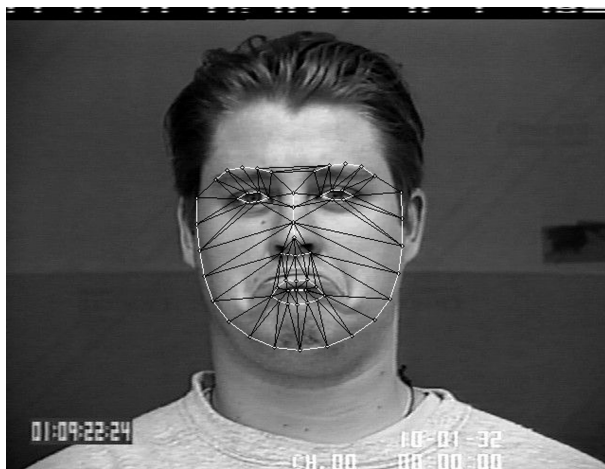
The following experiment was done in order to derive a relationship between the parameter of discrimination and the number of triangles (i.e. the number of features). The Fig. 5.5 shows a plot of Parameter of Discrimination (\mathcal{D}) against the



(a)



(b)



(c)

Fig. 5.3: Examples of images where the FaceTracker fails.

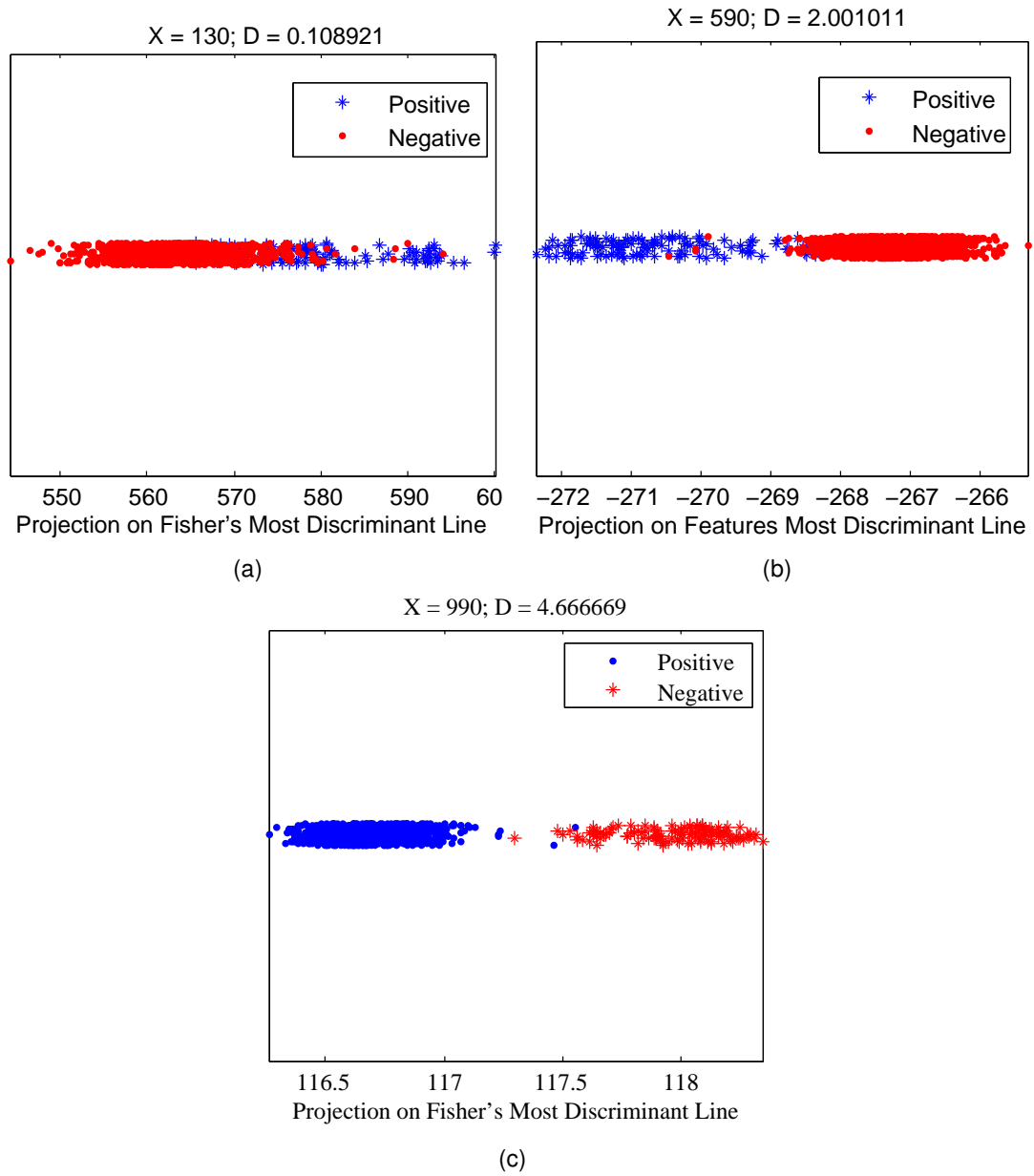


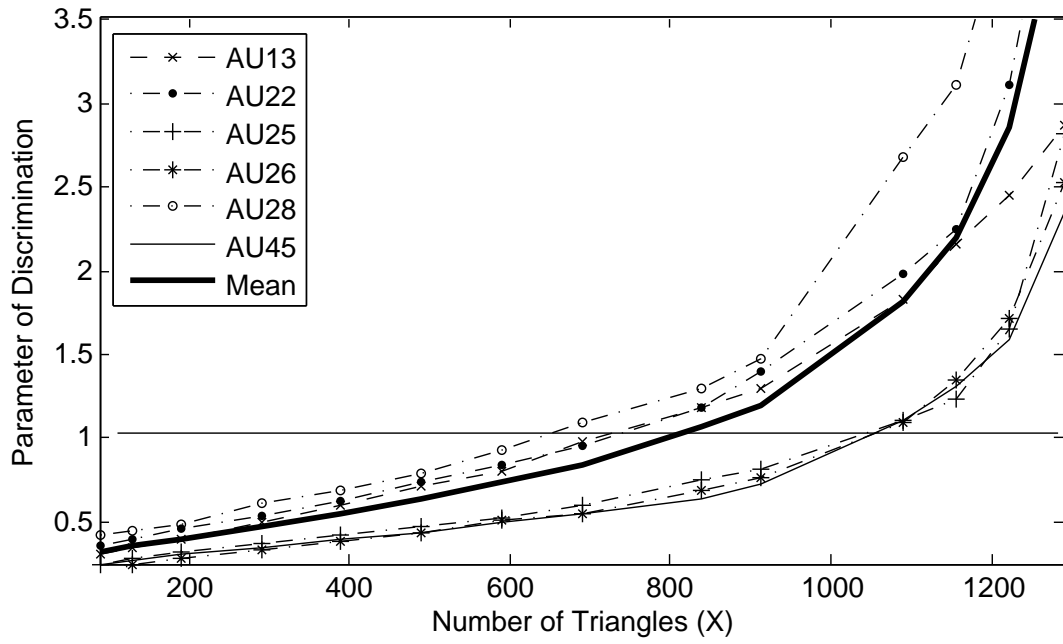
Fig. 5.4: Projection of the positive and negative features for AU10 on the 1st LDA eigenvector. \mathcal{D} represents the Parameter of Discrimination for a particular Number of Features (X).

Number of Triangles (X) used to create the CAPPX features. It can be noted from the plot that with increasing number of features, \mathcal{D} increases in an exponent-like fashion. This observation is coherent with the intuition mentioned earlier that with increasing number of triangles, the features retain more information useful for discriminating facial actions. It is also clear from the figure that for MMI database, if the number of triangles is about 1100, the value of \mathcal{D} exceeds 1.

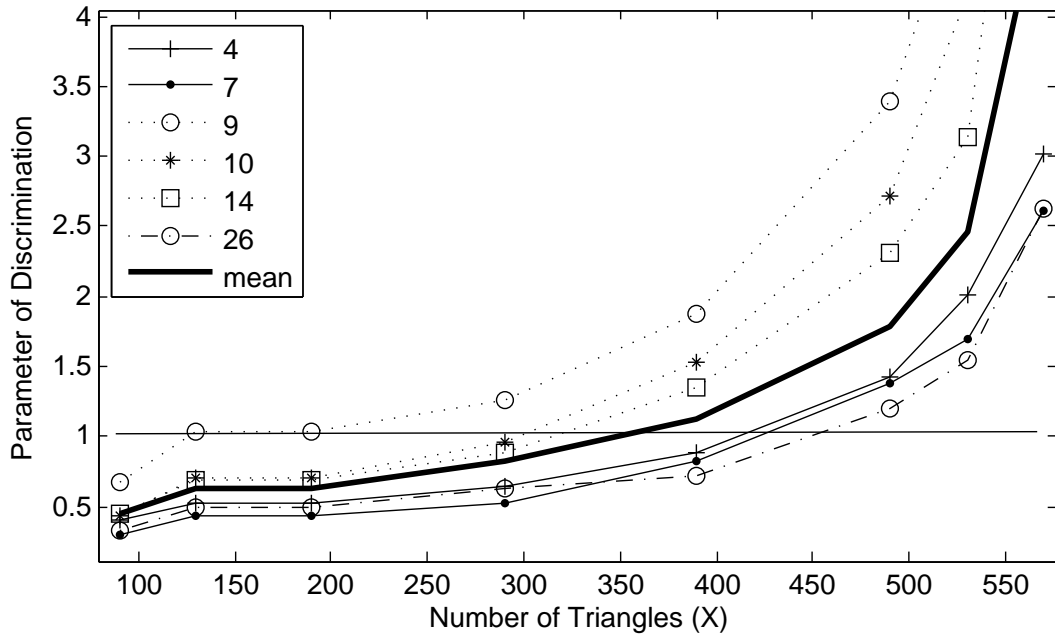
Similar experiment was done using the Extended Cohn-Kanade (CK+) database. In this case, only the apex frames were used to make the chart. However, the rate of increase is different than that of MMI database. CK+ requires less number of features than MMI to obtain a similar value of \mathcal{D} . In other words, the apex frames of CK+ database contain more information to distinguish among the presence or absence of an action unit than the middle frames of MMI database. A reason behind so might be due to the fact that expressions shown in CK+ are extremely exaggerated in the apex frames. On the other hand MMI dataset expresses a mild display of the action units. Another interesting observation is the AU 26 is least discriminative in both cases. This indicates a weakness of selected features to discriminate AU26. It may be due to the poor performance of the FaceTracker in the lower lips and cheek-chin area as demonstrated in Fig. 5.2.

5.2.3 Morphological and Appearance Properties of AUs

The effects of different AUs are reflected in both the landmark locations (i.e. shape) and appearance of face. The effects of some action units are more reflected in shape features while others are reflected in appearance features. For example, it is very intuitive that AU 1 (Inner brow raiser), AU 2 (Outer brow raiser), AU 27 (Mouth Stretch) etc. should reflect their existence in shape features. On the other hand AU 9 (Nose Wrinkler), AU14 (Dimpler), AU11 (Nasolabial Furrow Deepener) etc. should produce appearance changes. Such phenomenon is clearly reflected in discrimination parameters as shown in Fig. 5.6. The scatter plot in Fig. 5.6(a) shows the relative



(a)



(b)

Fig. 5.5: Plot of Parameter of Discrimination (\mathcal{D}) vs. Number of Features for (a) MMI Database and (b) Extended Cohn Kanade (CK+) Database. Three most discriminative and three least discriminative AUs are shown as well as the mean of all the AUs used in this work.

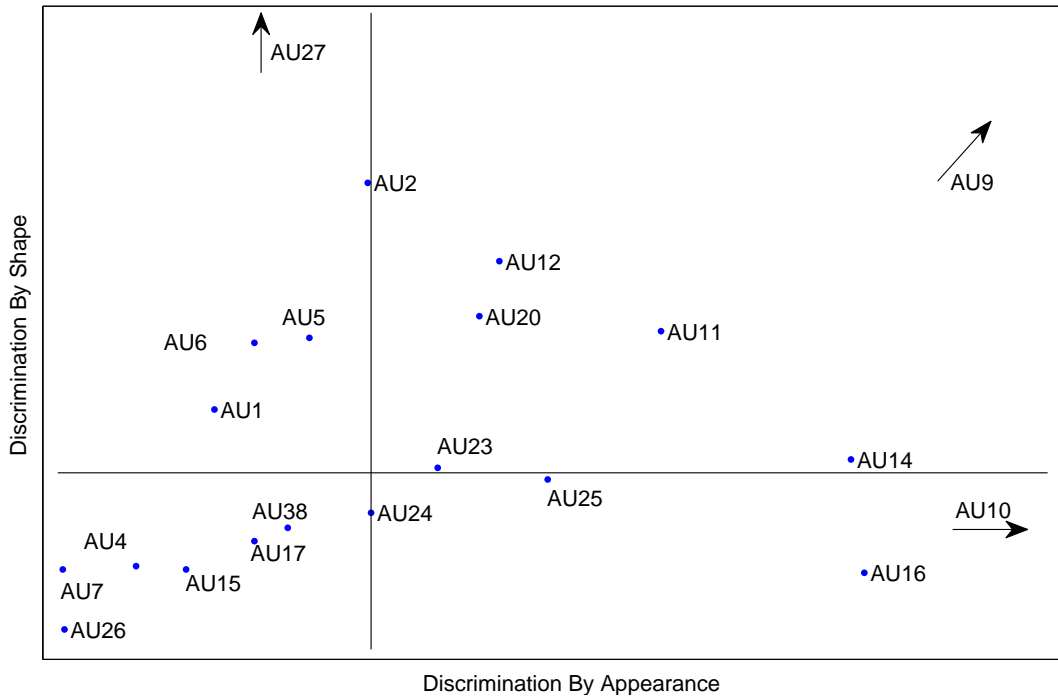
positions of different action units based on the values of \mathcal{D} for shape features (SPTS) and appearance features (CAPPX). The number of triangles used for calculating the appearance feature was 570. The horizontal and vertical lines represent the median.

From this plot, several properties of different action unit are readily visible. For example, some AUs are easy to discriminate by either of the two features (e.g. AUs in top right quadrant) while others are difficult (e.g. AUs in bottom left quadrant). Also some action units are more distinguishable by a certain kinds of features than the other (e.g. the AUs in top left and bottom right quadrants). Moreover, it can be noticed that the intuitions about the action units discussed earlier are reflected accordingly in the plot.

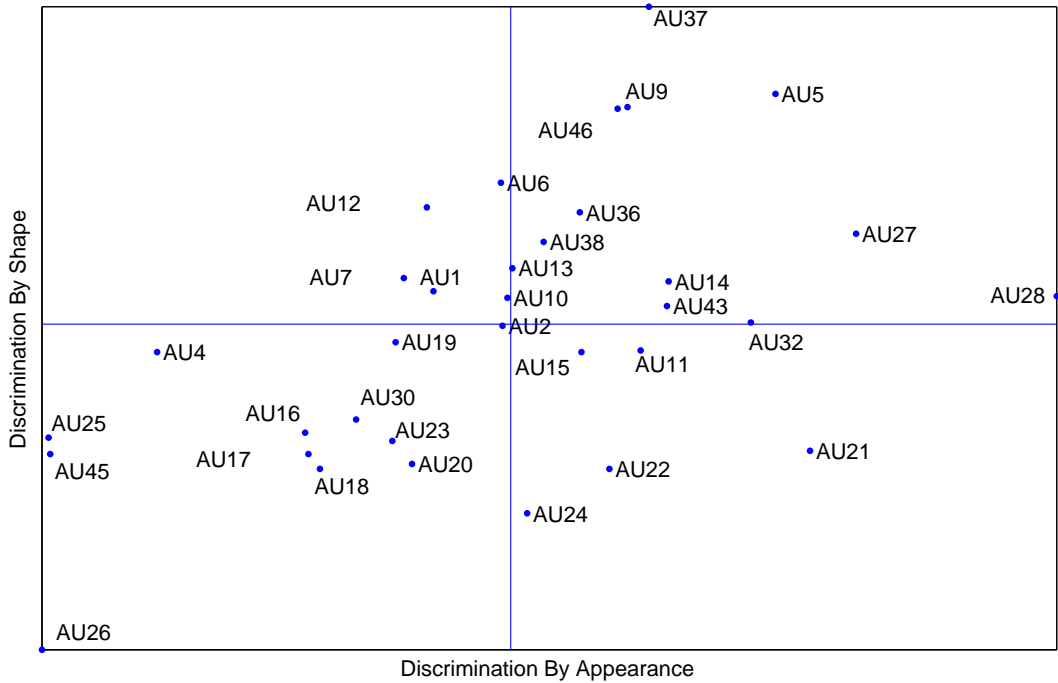
Fig. 5.6(b) shows the same plot for MMI database. An important fact can be observed from the two plots that AU25 (Lips part) and AU26 (Jaw drop) are not shown to be discriminable by shape although intuitively they should be so. This discrepancy is due to the poor performance of FaceTracker as discussed in Sec. 5.2.2. Moreover sometimes a particular AU also induces the occurrence of other AUs. In such cases, an appearance based AU can be more discriminative by non-appearance features or vice versa.

5.2.4 Effect of Hybridization on Parameter of Discrimination

In Fig. 5.7, \mathcal{D} for SPTS feature, CAPPX and their hybrid are shown for different action units. For CK+, CAPP530 and for MMI, CAPP1090 was used. These values were selected in such a way so that the value of \mathcal{D} becomes slightly greater than one for all the action units. From the figure it is to be noticed that SPTS features are not good enough to distinguish among the action units. When sufficiently triangulated, appearance features produce greater amount of discrimination. However, shape features have a great potential to increase the discriminating power when combined with the appearance features. This increment is more than the value of \mathcal{D} for shape features only. This effect is evident for both the CK+ and MMI database.

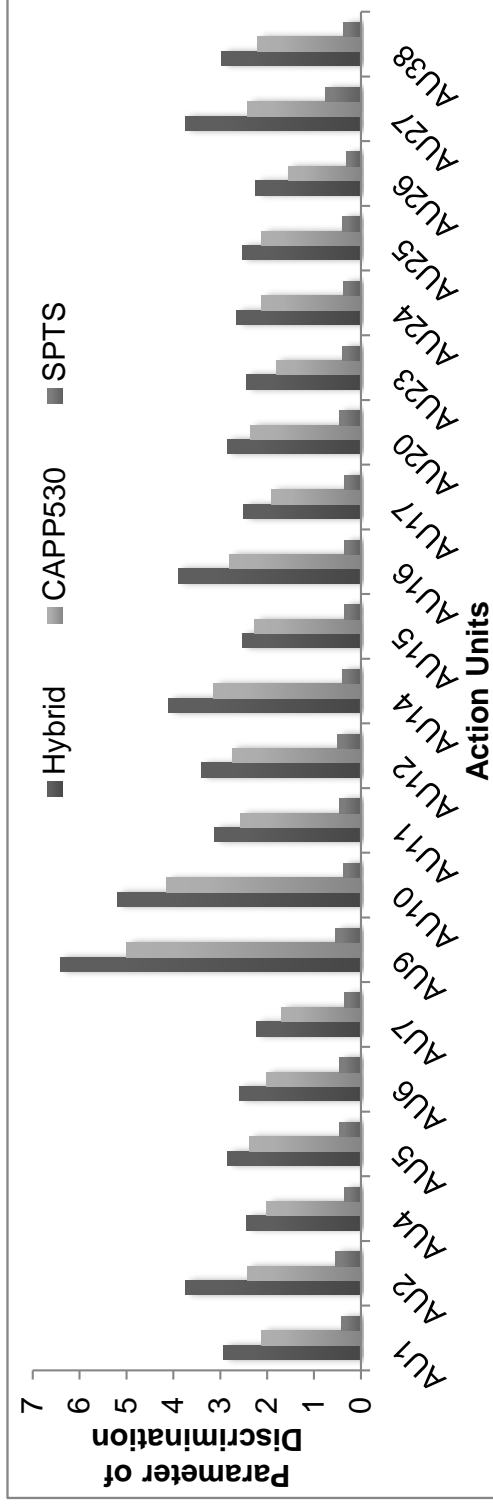


(a)

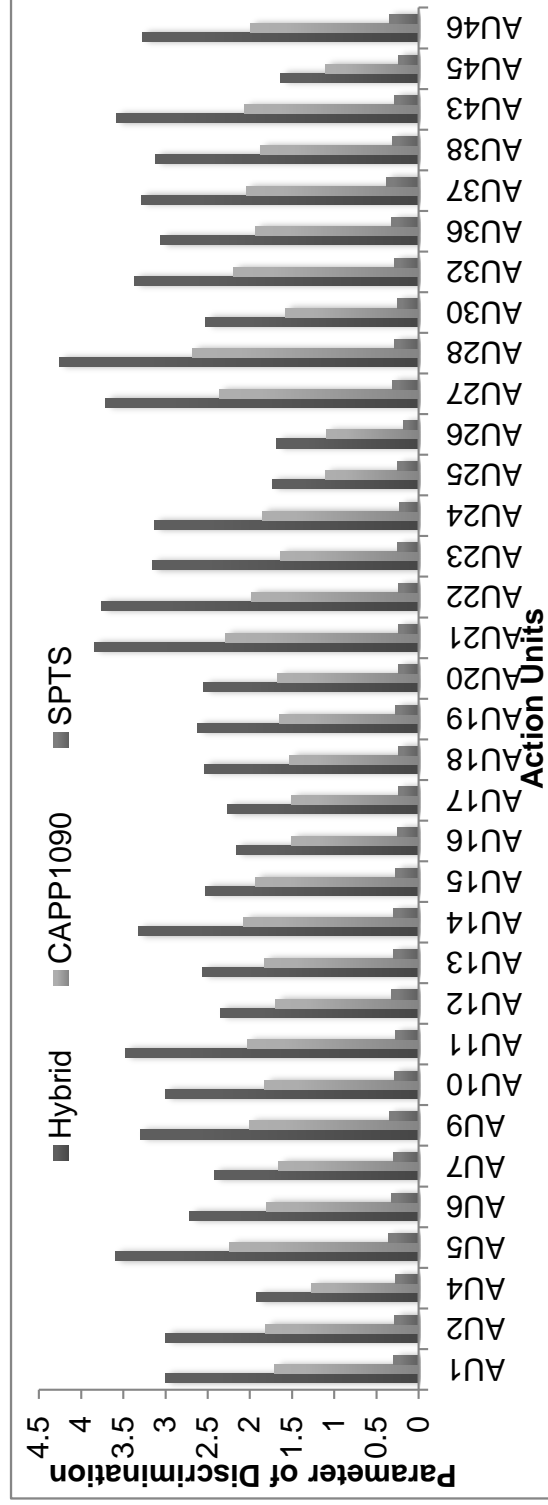


(b)

Fig. 5.6: Scatter plot of different AUs positioned based on the values of \mathcal{D}_s obtained from SPTS features and CAPP570 features. (a) shows the plot for CK+ database and (b) shows the same for MMI database.



(a)



(b)

Fig. 5.7: Parameter of Discrimination for different Features and Action Units. (a) shows the plot for CK+ (b) shows for MMI.

5.3 Prediction Level Evaluation

5.3.1 Receiver Operating Characteristics (ROC)

All prediction level evaluations are done based on the output predicted by the classifier located in last level of the system. A well established method for evaluating classification performance is Receiver Operating Characteristics also known as ROC. ROC shows a detailed view about the performance of a classifier in different possible conditions. It is a plot of True Positive Rate (TPR) versus False Positive Rate (FPR). TPR is a classifier performance representing what fraction of positive data points are correctly classified. On the other hand, FPR represents the fraction of negative samples that has been incorrectly classified as positive. Therefore, it is desirable for a good classifier to have a high true TPR and a low FPR. In other words, an ideally good classifier will cover the whole ROC space.

An example of an ROC curve is shown in Fig. 5.8. This curve was calculated using CAPP1090 features for AU9 from MMI database. Weka toolbox [53] was used for all the necessary processing. A boosting based meta classifier (AdaBoostM1) was used where the base classifier was a decision stub. All other default parameters in weka were used. Five fold cross validation was adopted. The area under this ROC curve (AUC) is 0.765. It should be noted that for an ideal ROC where TPR is 1 and FPR is 0 the area under ROC is 1. For more information on ROC and AUC please refer to [54].

5.3.2 AUC vs. AU: Effect of Features

The following experiment was designed in order to observe the effect of different features on Area Under ROC Curve (AUC). The AUC was calculated using decision stump based Ada Boost [55] classifier. Three different kinds of features were used in this experiment - CAPPX, SPTS and Hybrid. For CK+ database, 530 appearance features were used. On the other hand for MMI database, 1090 number of appearance features used. In all the cases a five fold cross validation is used.

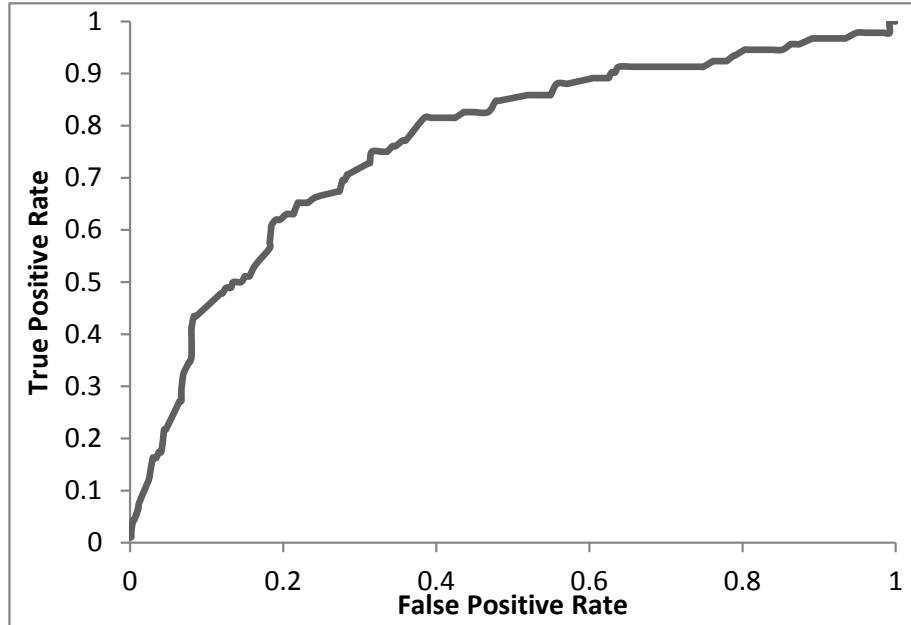
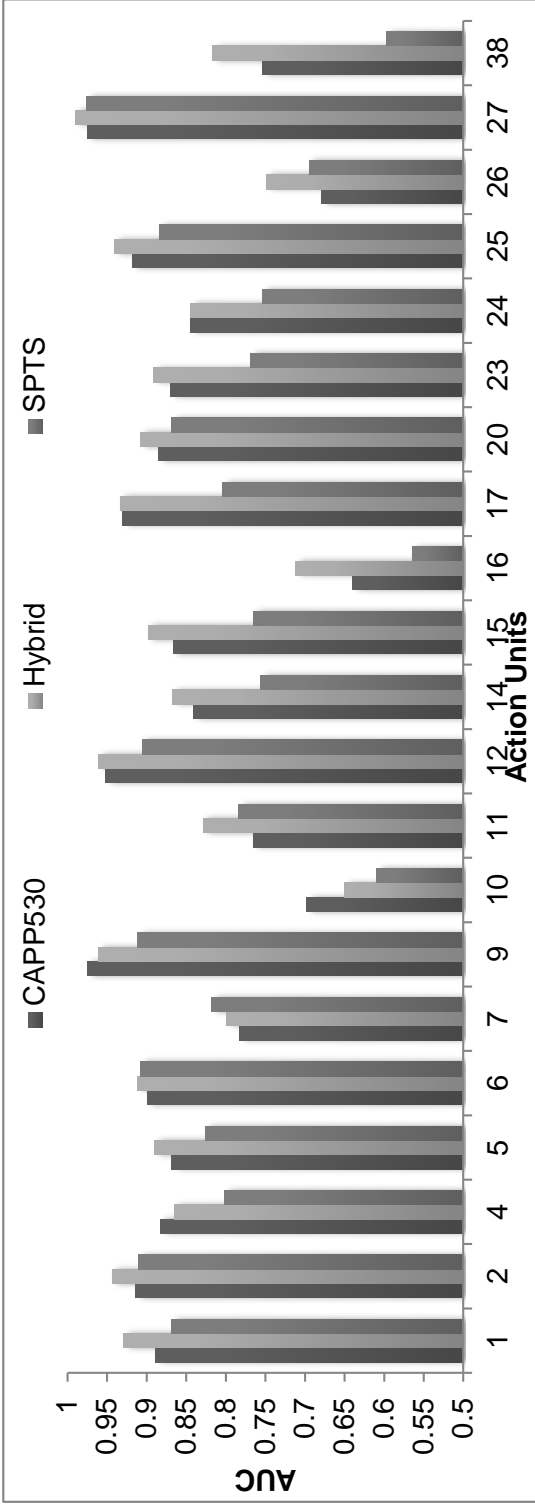
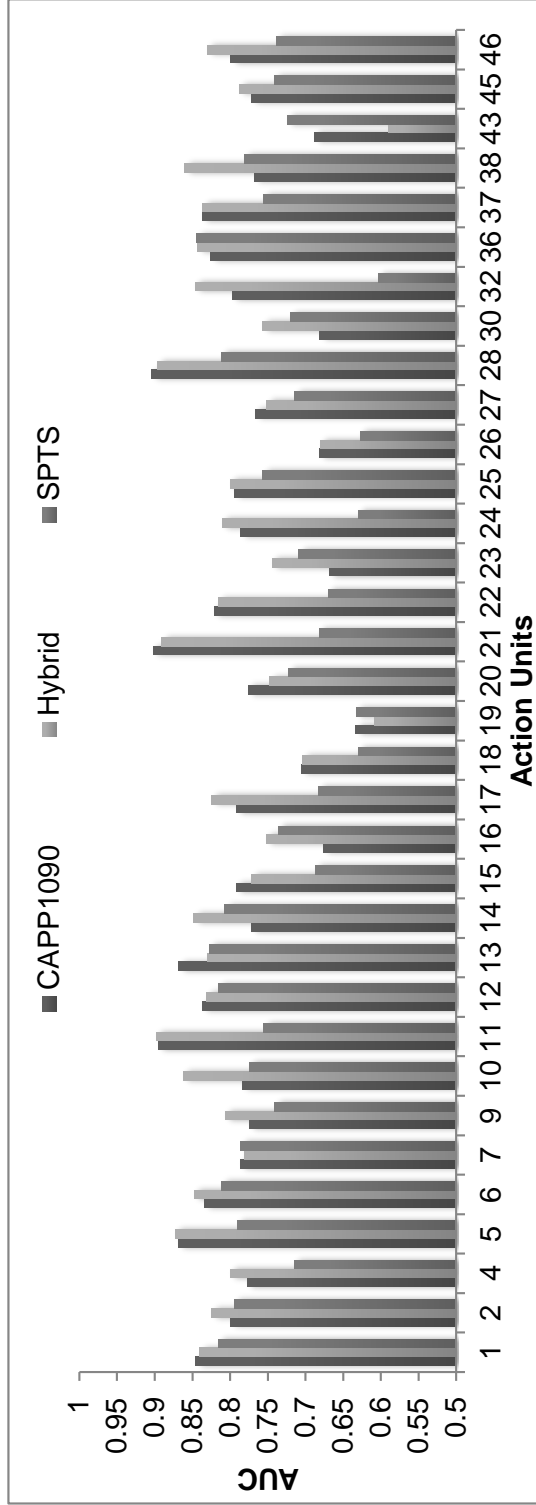


Fig. 5.8: An example of ROC Curve.

As shown in the results in Fig. 5.9 it can be noted that the SPTS features give a low classification performance than any other features. However, when this is merged with the appearance features (i.e. hybrid) it shows significant improvement over the CAPPX features. It should be noted that similar phenomenon was observed with parameter of discrimination, \mathcal{D} , which gives an impression that \mathcal{D} can provide some indication about the classification performances without actually doing the classification. This information is helpful for choosing better features in order to get good classification performance. However, the relationship between \mathcal{D} and AUC is probably not very straightforward because there are some action units which show decreased AUC with hybrid features which never happened in case of \mathcal{D} . This is because the AUC incorporates several sources of inefficiencies; for example, the classifier used and its parameters, the curse of dimensionality etc.



(a)



(b)

Fig. 5.9: AUC for different Features and Action Units. (a) shows the plot for CK+ (b) shows for MMI.

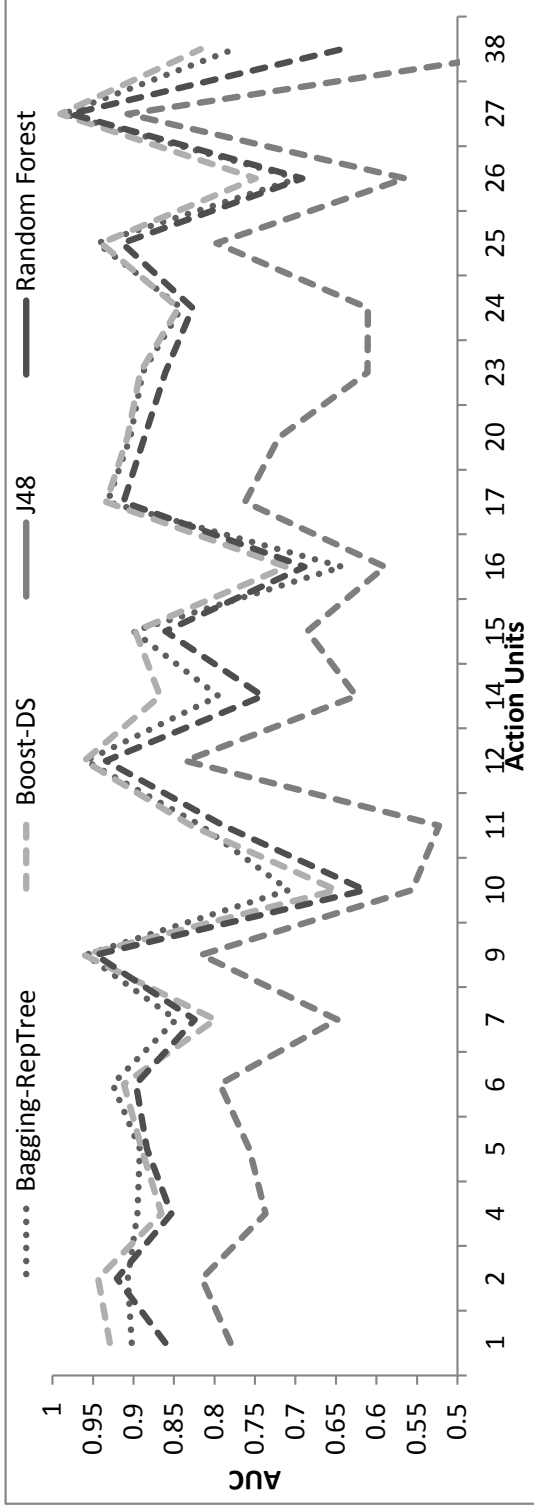
5.3.3 AUC vs. AU: Effect of Classifiers

It is mentioned in the previous section that the choice of classifiers might influence the AUC. In order to verify that assertion the following experiment was designed. Four different classifiers were used in this experiment: Decision stump based Ada-Boost classifier, A fast decision tree based bagging classifier, J48 tree based classifier and Random forest classifier. Hybrid feature was used with the same number of triangulation as in the previous section. The result of this experiment is shown in Fig. 5.10.

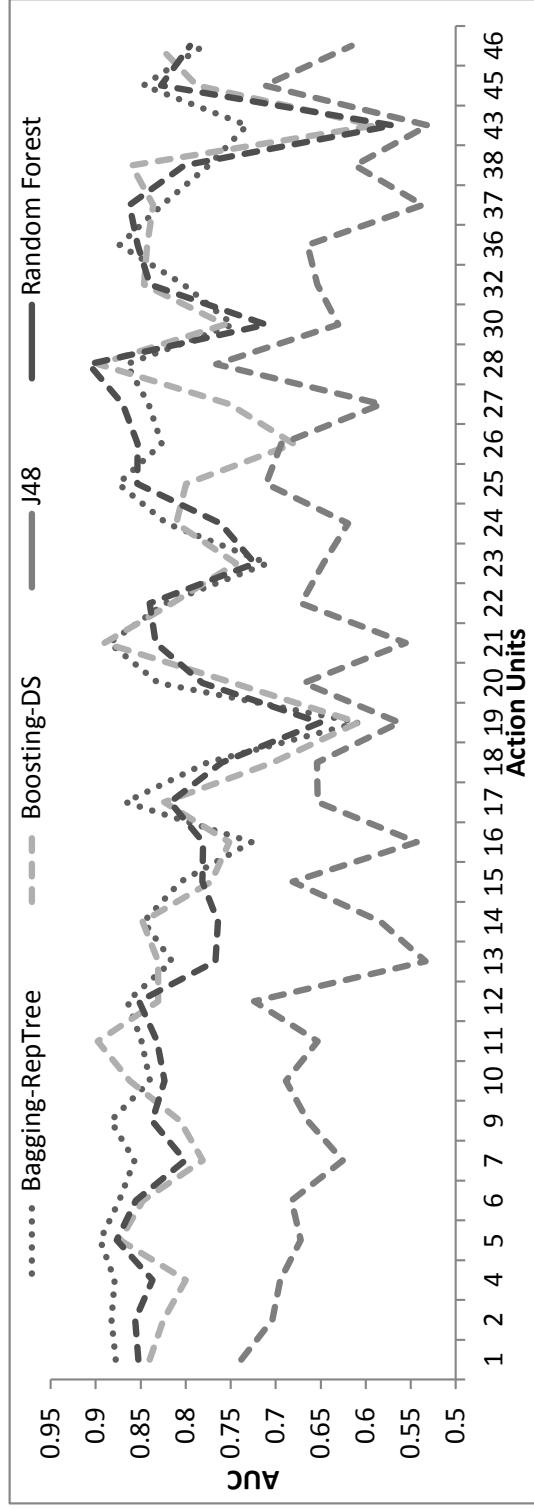
It is evident from the figure that all the four different classifiers show different performances although their input is identical. Among these classifiers, ensemble classifiers (bagging and boosting) show best performance in most of the cases. In certain occasions, random forest also provides the best classification performance. However, J48 tree based classifier is not good for most of the cases.

5.3.4 AUC vs. Number of Features

It is mentioned earlier in several occasions that the number of features in CAPPX has significant influence over the classification performance. Fig. 5.11 is computed using MMI database where the effect of number of features (X) on AUC is demonstrated. Fig. 5.11(b) shows that with increasing number of features the AUC is first increases. However, after a certain point it starts to decrease. This demonstrates the effect of curse of dimensionality. With increasing number of features, possible sample space grows exponentially which results in poor classification performance. Another interesting point can be noted in Fig. 5.11(a). It shows that for different AUs, AUC reaches maximum in a different amount of features. This indicates that for different action units, a different number of features are needed for optimal classification performance. Similar phenomenon is also evident in Fig. 5.12(b) where the calculations were made using CK+ database.

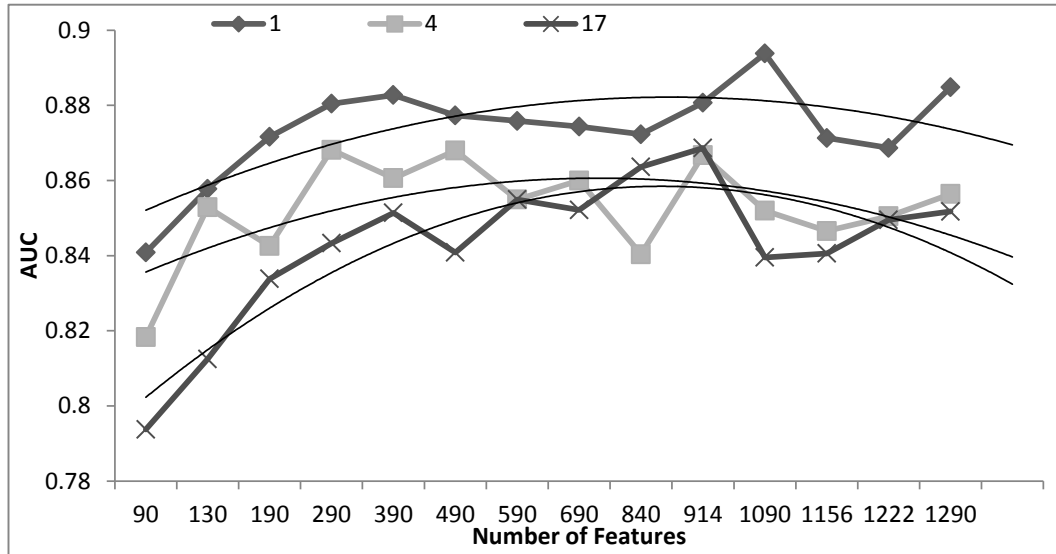


(a)

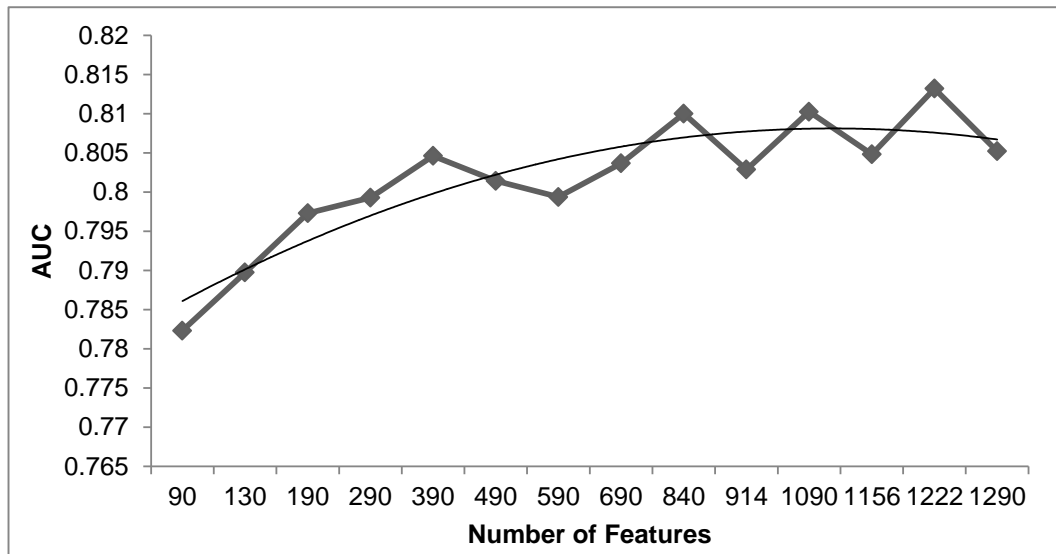


(b)

Fig. 5.10: AUC for different classifier. (a) shows the plot for CK+ (b) shows for MMI.

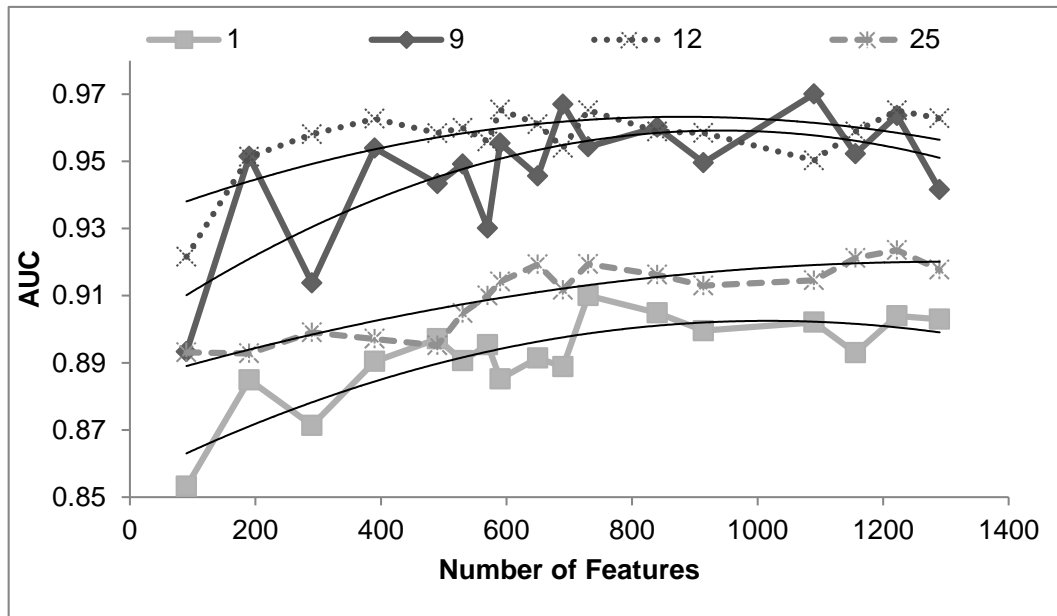


(a)

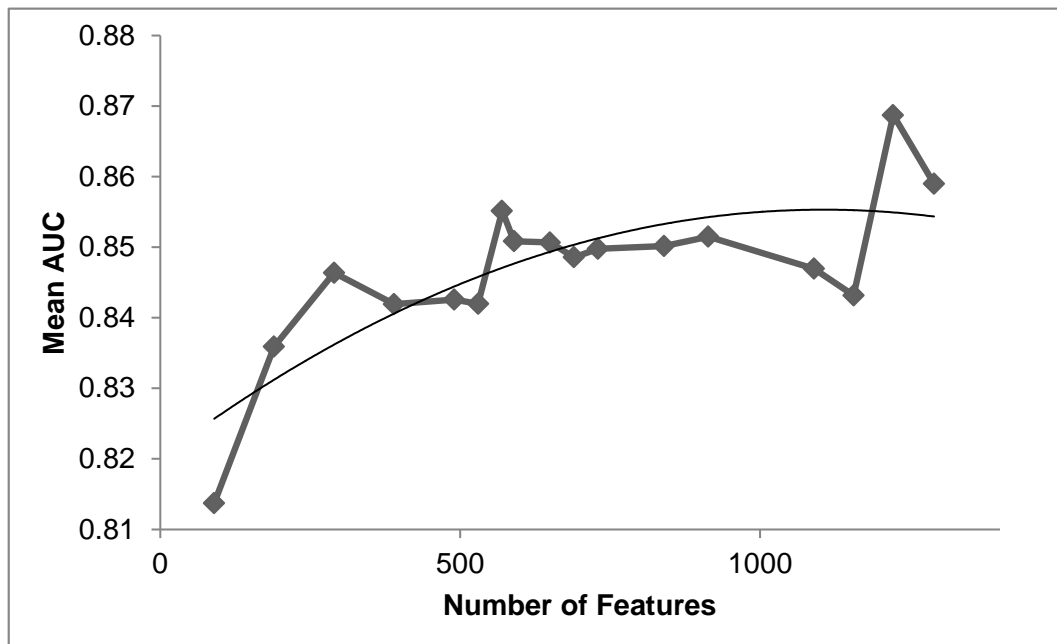


(b)

Fig. 5.11: Plots and 2nd order polynomial trend-lines of (a) AUC vs Number of Features for a few action units and (b) Mean AUC of all the action units vs Number of features. All the calculations are made using MMI database.

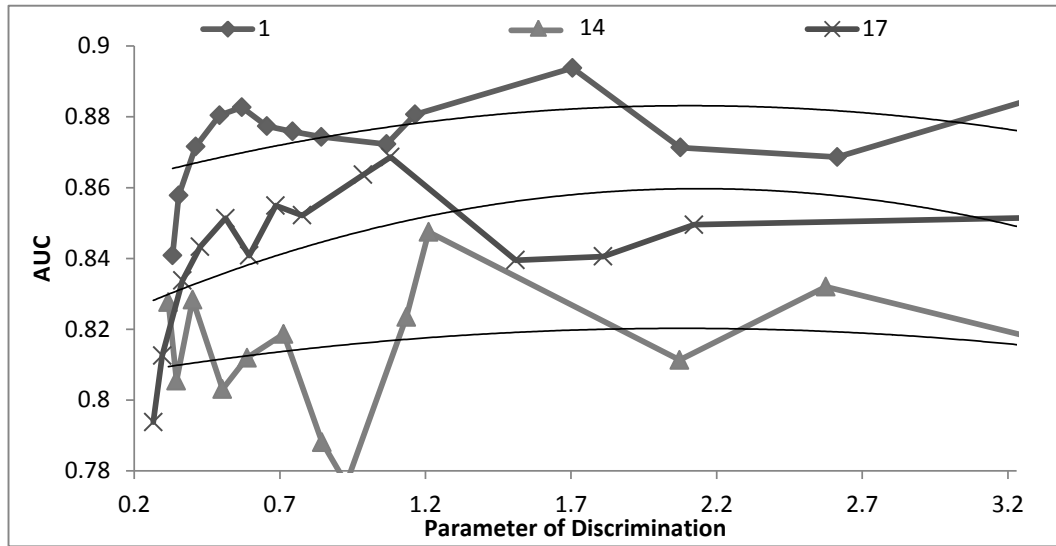


(a)

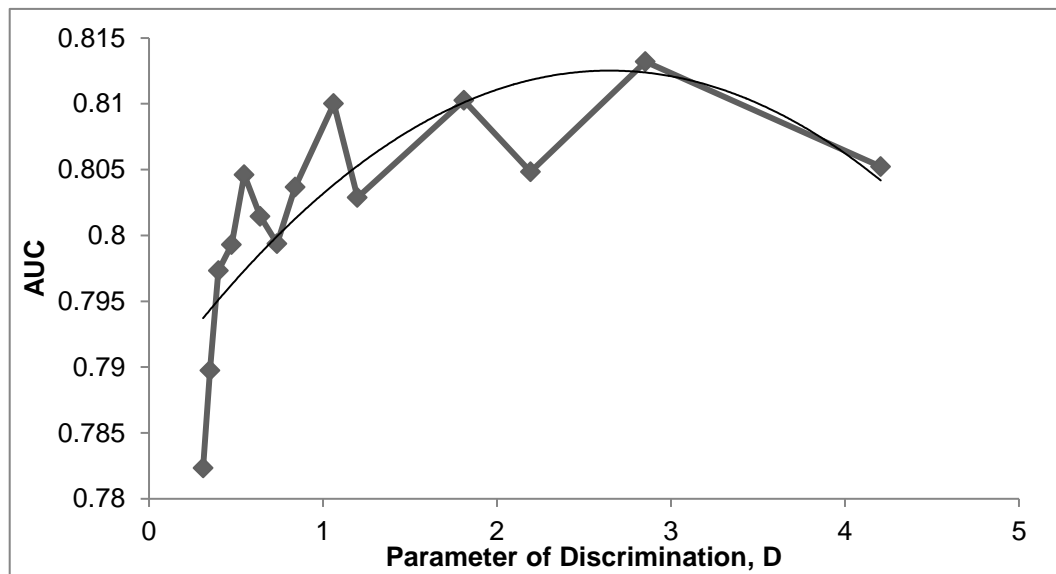


(b)

Fig. 5.12: Plots and 2nd order polynomial trend-lines of (a) AUC vs Number of Features for a few action units and (b) Mean AUC of all the action units vs Number of features. All the calculations are made using CK+ database.



(a)



(b)

Fig. 5.13: Plots and 2nd order polynomial trend-lines of (a) AUC vs Parameter of Discrimination for a few action units and (b) Mean AUC of all the action units vs Parameter of Discrimination. All the calculations are made using MMI database.

5.3.5 AUC vs. Parameter of Discrimination

The plot of AUC versus Parameter of Discrimination as shown in Fig. 5.13 demonstrates an important phenomenon. It is evident from the trend lines that with increasing values of \mathcal{D} , AUC increases and then after a threshold it starts decreasing again. Unlike the number of features, AUC reaches to its peak in a consistent manner when plot against \mathcal{D} . From the mean AUC plot as shown in Fig. 5.13(b) it can be noticed that average AUC reaches maximum when \mathcal{D} is about 2.7.

Chapter 6

Conclusion

The previous chapters have demonstrated a Three Level Evaluation approach for a simple action unit detection system. Primary contribution of this work is to acknowledge that a comprehensive evaluation is necessary to determine the utility of model based approaches. The key idea is to evaluate all the major components instead of the last component only. In addition to this, several problems of effective evaluation of model based approaches are identified and some solutions are also suggested.

A new metric to evaluate landmark detection systems has been proposed in this work which is named as Normalized Root Mean Squared Point Error (NRMS-PE). It has been shown that this new error metric is invariant to non-proportionate scaling of horizontal and vertical axes of images and also convenient to calculate than the classical error metric (RMS-PE). Furthermore, a new parameter has been proposed named the “Parameter of Discrimination”, \mathcal{D} , in order to calculate the quality of the extracted features. The efficacy of this parameter has been assessed through several experiments.

It has been shown that, with the help of these improved and newly proposed evaluation metrics, the three level evaluation approach reveals significantly more information than its traditional counterpart. This is useful for literary comparison of the strengths and weaknesses of different systems.

REFERENCES

- [1] Mwtoews. (2007, April) Normal distribution curve that illustrates standard deviations. Based (in concept) on figure by Jeremy Kemp, on 2005-02-09.
http://en.wikipedia.org/wiki/File:Standard_deviation_diagram.svg
- [2] J. Saragih, S. Lucey, and J. Cohn, "Deformable model fitting by regularized landmark mean-shift," *International Journal of Computer Vision*, pp. 1–16, 2011.
- [3] R. W. Picard, "Affective computing," M.I.T. Media Laboratory Perceptual Computing Section, Tech. Rep. 321, 1995.
- [4] A. Graesser, P. Chipman, B. Haynes, and A. Olney, "Autotutor: An intelligent tutoring system with mixed-initiative dialogue," *IEEE Transactions on Education*, vol. 48, no. 4, pp. 612–618, 2005.
- [5] B. McDaniel, S. Ó Mello, B. King, P. Chipman, K. Tapp, and A. Graesser, "Facial features for affective state detection in learning environments," *Proceedings of the 29th Annual Cognitive Science Society*, 2007, pp. 467–472.
- [6] A. Ashraf, S. Lucey, J. Cohn, T. Chen, Z. Ambadar, K. Prkachin, and P. Solomon, "The painful face - pain expression recognition using active appearance models," *Image Vision Comput.*, vol. 27, no. 12, pp. 1788–1796, 2009.
- [7] J. Cohn, T. Kruez, I. Matthews, Y. Yang, M. Nguyen, M. Padilla, F. Zhou, and F. De la Torre, "Detecting depression from facial actions and vocal prosody," *Affective Computing and Intelligent Interaction and Workshops, 2009*, 2009.
- [8] J. Russell, J. Bachorowski, and J. Fernández-Dols, "Facial and vocal expressions of emotion," *Annual Review of Psychology*, vol. 54, no. 1, pp. 329–349, 2003.
- [9] P. Ekman, "Facial expression and emotion." *American Psychologist*, vol. 48, no. 4, p. 384, 1993.
- [10] P. Ekman and W. Friesen, *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Palo Alto: Consulting Psychologists Press, 1978.
- [11] M. Bartlett, G. Littlewort, and M. Frank, "Fully automatic facial action recognition in spontaneous behavior," *Recognition, 2006.*, pp. 223–230, 2006.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1613024
- [12] T. F. Cootes and C. J. Taylor, "Active shape models – smart snakes," in *Proc. British Machine Vision Conference*, vol. 266275. Citeseer, 1992, pp. –. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.17.5163&rep=rep1&type=pdf>
- [13] B. Fasel and J. Luetttin, "Automatic facial expression analysis: a survey," *Pattern Recognition*, vol. 36, no. 1, pp. 259–275, 2003.
<http://linkinghub.elsevier.com/retrieve/pii/S0031320302000523>
- [14] Y. Chen and F. De la Torre, "Active conditional models," *IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, 2011.

- [15] T. Cootes, G. Edwards, and C. Taylor, "Active appearance models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 23, no. 6, pp. 681–685, 2001.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=927467
- [16] J. M. Saragih, S. Lucey, and J. F. Cohn, "Face alignment through subspace constrained mean-shifts," *Computer Vision, 2009 IEEE 12th International Conference on*, no. Clm. IEEE, Sept. 2010, pp. 1034–1041.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5459377
- [17] D. Cristinacce and T. F. Cootes, "Feature detection and tracking with constrained local models," *Proc. British Machine Vision Conference*, vol. 3. Citeseer, 2006, pp. 929–938.
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.108.6210&rep=rep1&type=pdf>
- [18] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3d faces," *Proceedings of the 26th annual conference on Computer graphics and interactive techniques - SIGGRAPH '99*, pp. 187–194, 1999.
<http://portal.acm.org/citation.cfm?doid=311535.311556>
- [19] J. Saragih, S. Lucey, and J. Cohn, "Subspace constrained mean-shift," Pittsburg, Tech. Rep., 2009. http://swing.adm.ri.cmu.edu/pub_files/2009/5/TR-09-15.pdf
- [20] I. Matthews and S. Baker, "Active appearance models revisited," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 135–164, 2004.
- [21] C. Hu, R. Feris, and M. Turk, "Active wavelet networks for face alignment," *British machine vision conference*. Citeseer, 2003, <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.4.167&rep=rep1&type=pdf>
- [22] J. Xiao, S. Baker, I. Matthews, and T. Kanade, "Real-time combined 2d+3d active appearance models," *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2004, pp. 535–542.
- [23] A. Lanitis, C. Taylor, and T. F. Cootes, "Automatic interpretation and coding of face images using flexible models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 743–756, July 1997.
<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=598231>
- [24] T. Kanade, B. Lucas, and An, "Iterative image registration technique with an application to stereo vision," *International Joint Conference on Artificial In*, pp. 674–679, 1981.
- [25] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *International Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, 2004.
<http://www.springerlink.com/index/T032H4PG050X2012.pdf>
- [26] P. Lucey, J. Cohn, S. Lucey, S. Sridharan, and K. Prkachin, "Automatically detecting action units from faces of pain: Comparing shape and appearance features," *Computer Vision and Pattern Recognition Workshops, 2009. CVPR Workshops 2009. IEEE Computer Society Conference on*. IEEE, 2009, pp. 12–18.











- [27] P. Lucey, J. F. Cohn, I. Matthews, S. Lucey, S. Sridharan, J. Howlett, and K. M. Prkachin, "Automatically detecting pain in video through facial action units," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 41, no. 3, pp. 664–674, June 2011.
- [28] R. Gross, I. Matthews, and S. Baker, "Generic vs. person specific active appearance models," *Image and Vision Computing*, vol. 23, no. 11, 2005.
- [29] D. Cristinacce and T. Cootes, "Facial feature detection using adaboost with shape constraints," *British Machine Vision Conference*, vol. 1. Citeseer, 2003, pp. 231–240.
- [30] D. Cristinacce and T. F. Cootes, "A comparison of shape constrained facial feature detectors," *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.*, pp. 375–380, May 2004.
<http://www.computer.org/portal/web/csdl/doi/10.1109/AFGR.2004.1301561>
- [31] D. Cristinacce and T. F. Cootes, "Boosted regression active shape models," *18th British Machine Vision Conference, Warwick, UK*. Citeseer, 2007, pp. 880–889.
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.66.7073&rep=rep1&type=pdf>
- [32] Y. Wang, S. Lucey, and J. F. Cohn, "Enforcing convexity for improved alignment with constrained local models," *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, vol. 2008. IEEE, June 2008, pp. 1–8.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4587808
- [33] J. Saragih and R. Gócke, "Learning aam fitting through simulation," *Pattern Recognition*, vol. 42, no. 11, pp. 2628–2636, Nov. 2009.
<http://linkinghub.elsevier.com/retrieve/pii/S0031320309001514>
- [34] J. Saragih. (2010, June) Facetracker. Webpage.
<http://web.mac.com/jsaragih/FaceTracker/FaceTracker.html>
- [35] S. W. Chew, P. J. Lucey, S. Lucey, J. Saragih, J. Cohn, and S. Sridharan, "Person-independent facial expression detection using constrained local models," *Proceedings of FG 2011 Facial Expression Recognition and Analysis Challenge*, no. March, 2011. <http://eprints.qut.edu.au/40354/>
- [36] P. Lucey, S. Lucey, and J. F. Cohn, "Registration invariant representations for expression detection," *2010 International Conference on Digital Image Computing: Techniques and Applications*, no. i. IEEE, Dec. 2010, pp. 255–261.
<http://www.computer.org/portal/web/csdl/doi/10.1109/DICTA.2010.53>
- [37] S. Lucey, I. Matthews, C. Hu, Z. Ambadar, F. De la Torre, and J. Cohn, "Aam derived face representations for robust facial action recognition," in *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*. IEEE, 2006, pp. 155–160. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1613014
- [38] S. Lucey, A. Ashraf, and J. Cohn, "Investigating spontaneous facial action recognition through aam representations of the face." Citeseer, 2007, pp. 395–406.
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.69.9620&rep=rep1&type=pdf>




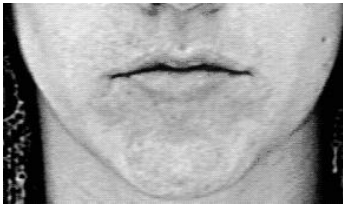





- [39] T. Brick, M. Hunter, and J. Cohn, "Get the facts fast: Automated face analysis benefits from the addition of velocity," *Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009. 3rd International Conference on*. IEEE, 2009, http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5349600
- [40] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, *The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression*. IEEE, June 2010. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5543262&abstractAccess=no&userType=inst
- [41] D. Cai, X. He, and J. Han, "Srda: An efficient algorithm for large-scale discriminant analysis," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–12, 2007.
- [42] R. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Human Genetics*, vol. 7, no. 2, pp. 179–188, 1936.
- [43] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie," *Image and Vision Computing*, vol. 28, no. 5, pp. 807–813, 2010.
- [44] T. Kanade, Y. Tian, and J. Cohn, "Comprehensive database for facial expression analysis," *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, 2000, pp. 46–53.
- [45] M. Valstar and M. Pantic, "Induced disgust, happiness and surprise: an addition to the mmi facial expression database," *The Workshop Programme*, 2010.
- [46] M. Pantic, M. Valstar, R. Rademaker, and L. Maat, "Web-based database for facial expression analysis," *2005 IEEE International Conference on Multimedia and Expo*. IEEE, 2005.
- [47] F. Bookstein, "Landmark methods for forms without landmarks: Localizing group differences in outline shape," *Mathematical Methods in Biomedical Image Analysis, IEEE Workshop on*, p. 0279, 1996.
- [48] M. Stegmann, "Active appearance models: Theory, extensions and cases," Master's thesis, 2000.
- [49] Wikipedia. Delaunay triangulation. Wikipedia, the free encyclopedia. http://en.wikipedia.org/wiki/Delaunay_triangulation
- [50] S. Laughlin, "A simple coding procedure enhances a neuron's information capacity," *Z. Naturforsch*, vol. 36, no. 9-10, pp. 910–912, 1981.
- [51] G. Bradski and A. Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, 2008.
- [52] J. Lee and M. Verleysen, *Nonlinear dimensionality reduction*. Springer Verlag, 2007. http://books.google.com/books?hl=en&lr=&id=vyVo4fO_9bQC&oi=fnd&pg=PP13&dq=Nonlinear+Dimensionality+Reduction&ots=NEpcmJwESH&sig=i38QBbJhP2cvI6VSFcWK5Ll42Q8










- [53] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. Witten, "The weka data mining software: an update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.
- [54] T. Fawcett, "Roc graphs: Notes and practical considerations for researchers," *Machine Learning*, vol. 31, no. HPL-2003-4, pp. 1–38, 2004.
- [55] Y. Freund and R. Schapire, "Experiments with a new boosting algorithm," in *MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE*. Citeseer, 1996, pp. 148–156.
- [56] T. Cootes, G. Edwards, and C. Taylor, "Active appearance models," *Computer Vision/ECCV'98*, p. 484, 1998.
- [57] C. Goodall, "Procrustes methods in the statistical analysis of shape," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 53, no. 2, pp. 285–339, 1991.
<http://www.jstor.org/stable/2345744>
- [58] L. Smith. (2002) A tutorial on principal components analysis.
<http://kybele.psych.cornell.edu/~edelman/Psych-465-Spring-2003/PCA-tutorial.pdf>
- [59] M. Bern and D. Eppstein, "Mesh generation and optimal triangulation," 1992.
- [60] J. R. Shewchuk, *Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator*, ser. Lecture Notes in Computer Science. Berlin: Springer-Verlag, May 1996, vol. 1148, pp. 203–222.
<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.40.9874>
- [61] S. Baker, R. Gross, I. Matthews, and T. Ishikawa, "Lucas-kanade 20 years on: A unifying framework: Part 2," *International Journal of Computer Vision*, vol. 56, no. 221-255, p. 6, 2002.
- [62] M. H. Nguyen and F. De la Torre, "Metric learning for image alignment," *International Journal of Computer Vision*, vol. 88, no. 1, pp. 69–84, 2010.
- [63] G. Hager and P. Belhumeur, "Efficient region tracking with parametric models of geometry and illumination," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 10, pp. 1025–1039, 1998.
- [64] R. Gross, I. Matthews, and S. Baker, "Active appearance models with occlusion," *Image and Vision Computing*, vol. 24, no. 1, pp. 593–604, 2006.






Appendix A

Names and Examples of Action Units

AU/AD	Description	Example image
1	Inner Brow Raiser	
2	Outer Brow Raiser	
4	Brow Lowerer	
5	Upper Lid Raiser	
6	Cheek Raiser	
7	Lid Tightener	
9	Nose Wrinkler	
10	Upper Lip Raiser	
11	Nasolabial Deepener	
12	Lip Corner Puller	

AU/AD	Description	Example image
13	Cheek Puffer	
14	Dimpler	
15	Lip Corner Depressor	
16	Lower Lip Depressor	
17	Chin Raiser	
18	Lip Puckerer	
19	Tongue Show	
20	Lip stretcher	
22	Lip Funneler	

AU/AD	Description	Example image
23	Lip Tightener	
24	Lip Pressor	
25	Lips part	
26	Jaw Drop	
27	Mouth Stretch	
28	Lip Suck	
30	Jaw Sideways	
32	Bite	
36	Blow	

AU/AD	Description	Example image
37	Puff	
38	Suck	
43	Eyes Closed	
45	Blink	
46	Wink	

This list of Action Units and associated pictures were taken from the “Automated Face Analysis” webpage of Computer Science department of The Carnegie Mellon University.

URL: <http://www.cs.cmu.edu/~face/index2.htm>

Pictures of Tongue Show, Jaw Sideways, Bite, Blow, Puff, Suck and Wink were taken from the FACS Manual.

Appendix B

Model Based Landmark Detection Techniques

Model based facial expression recognition systems can employ a variety of landmark detection techniques. Among them two popular methods are discussed below.

B.1 Active Appearance Model

Active Appearance Model (AAM) [56] is a mathematical model capable to account for various deformations in morphable objects. Using AAM, it is possible to parameterize such deformations in terms of some known variations. Building AAM requires ground truth annotation of some predefined landmark points in a set of sample pictures of a morphable object. Through the model building process some constraints are defined regarding the positions of landmarks with respect to other landmarks. In the fitting process, these constraints are utilized as prior knowledge in order to parameterize a new image of the object.

Let us assume that each of $s_1, s_2, s_3, \dots, s_n$ be $2l$ dimensional vector representing the x and y coordinates of l landmark points in n sample images of an object. These vectors are often called shape vectors of the deformable object. Let us also assume that s_0 be the mean shape vector. That is,

$$s_0 = \frac{1}{n} \sum_{i=1}^n s_i \quad (\text{B.1})$$

For building the shape model, first, Procrustes analysis [57, 47] is applied on the set of shapes to align them by removing the effect of global transformations. Then Principal Component Analysis (PCA)[47] is utilized in order to obtain a set of orthonormal eigenvectors representing the major variations of the landmark points. A good tutorial of PCA and its physical interpretation can be found in [58]. The

eigenvectors found from PCA are often called the modes of deformations of the object. The change of a new shape vector from the mean can be expressed as a linear combination of these eigenvectors as below.

$$\mathbf{s} - \mathbf{s}_0 = c_1 \mathbf{e}_1 + c_2 \mathbf{e}_2 \dots + c_r \mathbf{e}_r \quad (\text{B.2})$$

Where \mathbf{s}_0 denotes the mean shape vector and $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_r$ be the shape eigenvectors representing the modes of deformations in shape. The coefficients of shape eigenvectors $c_1, c_2 \dots c_r$ represent the amount of corresponding deformations. Once the eigenvectors are fixed, any shape variation of a deformable object can be described by these coefficients $c_1, c_2 \dots c_r$ which are called shape parameters.

On the other hand, a set of parameters representing appearance variations of the object are known as appearance parameters. In order to parameterize the appearance, all the image contents are first warped into a canonical base shape and re-sampled. Generally the mean shape, \mathbf{s}_0 is used as the canonical shape and a piecewise affine warp based on Delaunay triangulation [59, 60] is used for warping. Some photometric normalization is also performed for eliminating the effect of global changes in illumination [48]. Then the pixel intensities of the part of image inside the convex hull created by the landmark points can be processed in the same way as shape vectors for getting linear modes of variations in appearance. The variation from mean appearance can be described as following where $\mathbf{a}, \mathbf{a}_0, \mathbf{u}_i, \lambda_i$ represent appearance vector, mean appearance vector, i^{th} Eigenvector and Eigenvalues respectively.

$$\mathbf{a} - \mathbf{a}_0 = \lambda_1 \mathbf{u}_1 + \lambda_2 \mathbf{u}_2 \dots + \lambda_q \mathbf{u}_q \quad (\text{B.3})$$

The mean shape (\mathbf{s}_0), mean appearance (\mathbf{a}_0) and the eigenvectors (\mathbf{e}_i and \mathbf{u}_j for $i = 1,$

2, ... r and $j = 1, 2, \dots q$) constitute the mathematical model for describing the variations of a deformable object which is known as Active Appearance Model (AAM). For more detailed description of building AAM please refer to [48, 20].

Once an Active Appearance Model is built, it can be used to identify the shape and appearance of a deformable object in a non-annotated image; a process known as model fitting. Fitting Active Appearance Model is an optimization process where the model parameters are tuned in order to synthesize an image which is closest to the test image. Sum of squared errors of the pixel intensities between the test image $I(\mathbf{x}, \mathbf{y})$ and the synthesized image $I(\mathbf{x}, \mathbf{y}|\mathbf{c}, \lambda)$ is generally used as an objective function.

$$[\mathbf{c}, \lambda] = \underset{\mathbf{c}, \lambda}{\operatorname{argmin}} \sum_{\mathbf{x}, \mathbf{y}} [I(\mathbf{x}, \mathbf{y}) - I(\mathbf{x}, \mathbf{y}|\mathbf{c}, \lambda)]^2 \quad (\text{B.4})$$

This optimization is done in a process known as Inverse Compositional Image Alignment (ICIA). Detailed discussion of such process is outside the scope of this work. For more information please refer to [20].

Many works have been done to improve the performance of AAM and thus it has many different versions. Efforts have been made to use other useful objective functions [61, 62]. Sometimes, a convexity criterion is enforced into the objective function [32] in order to avoid local minima and faster convergence in the optimization process. A “Project Out” [63] method is often used for faster fitting of the appearance. Also, works have been done to make AAM robust against occlusion [64] and identity [28].

B.2 Constrained Local Model (CLM)

Although many works have been done on AAM, it fails to robustly detect landmark points in certain cases. AAM does not robustly work when the model is built with several people. In other words, AAM is subject dependent. Moreover, it gives poor fitting performance when illumination is varied widely.

CLM [19, 16] is a technique that has obtained considerable attention to AAM researchers because it has been found useful as a partial solution to the generalization aspects of AAM. Generally, in AAM, the appearance vector is constituted of all the pixel intensities inside the convex hull of annotated landmarks which is known as “Holistic Approach” . Contrastingly, a patch based method is adopted in CLM, where a small region of appearance around each landmark is considered for modeling. This makes the varying illumination problem much easier to solve because lighting in a smaller region is more homogeneous. Appearance variation inside a patch is much less than that inside a big region. Consequently, it is easier to model the appearance variation with a simple PCA based dimensionality reduction technique.

The fitting process of CLM is constituted of two major stages. In the first stage, an exhaustive local search is performed by some local detectors to estimate locations of the patches in the image. A number of local detectors can be used for this purpose such as: linear logistic regressors, Gaussian likelihood and the Haar-based boosted classifier etc. From the local classifiers a likelihood map, $p(l_i = \text{aligned}|I, x)$ is obtained for each landmark. Where l_i is a random variable indicating whether the i^{th} landmark has aligned or not. x is a 2D location in image I .

In second step, conditional independence among the landmarks are assumed and the following objective function is maximized with respect to \mathbf{p} .

$$p(l_i = \text{aligned}|I, \mathbf{p}) = \prod_{i=1}^n p(l_i = \text{aligned}|I, x_i) \quad (\text{B.5})$$

Here, \mathbf{p} represents the scale, rotation, translation and non-rigid movement parameters of the landmarks created in the model building phase. A varieties of different optimization strategies can utilized in this stage. For more discussion on

these please refer to [19]. CLM is found to be robust in detecting landmarks in different lighting conditions. It is also found to be person independent [35]. The landmark tracker used in this work is built on CLM technique.