

AUTOMATIC ANALYSIS OF HEAD AND FACIAL GESTURES IN VIDEO  
STREAMS

by

Hatice Çınar Akakın

BS, in Electrical and Electronic Engineering, Eskişehir Osmangazi University, 2000

MS, in Electrical and Electronic Engineering, Anadolu University, 2003

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

Graduate Program in Electrical and Electronics Engineering

Boğaziçi University

2010

## ACKNOWLEDGEMENTS

I am very grateful to my PhD supervisor Prof. Bülent Sankur for his infinite support and invaluable guidance during my PhD. I am fortunate to be one of his students. He is an intelligent supervisor and an excellent person.

I would like to thank Prof. Lale Akarun and Assoc. Prof. Burak Acar for their precious feedbacks and scientific comments about my PhD research. I would like to thank Assoc. Prof. Çiğdem Erođlu Erdem and Prof. Aşşın Ertüzün for being part of my defense jury. I have special thanks to Assoc. Prof. Murat Saraçlar not only for his scientific comments but also for his joyful attitude in BÜSİM. I would like to thank Prof. Atalay Barkana for travelling from Eskişehir to participate in my defense jury.

I would like to thank Prof. Ömer Nezh Gerek, Prof. Altuğ İftar and Prof. Atalay Barkana from Anadolu University Eskişehir for their guidance and support.

I would like to thank Dr. Albert Ali Salah and Arman Savran for his scientific contributions to my research with their collaborations. I would also like to thank İsmail Arı for his valuable contributions on BUHMAP dataset.

My special thanks go to my colleagues and friends Dr. Ebru Arısoy, Oya Çeliktutan, Dr. Helin Dutağacı, Fulya Kunter and İpek Şen for their precious supports and encouragements. They made me to endure to the difficulties of the PhD progress.

I would like to thank my friends and colleagues at BÜSİM. They made the lab warm and a pleasant environment. My thanks go to Dr. Ceyhun Burak Akgül, Dr. Ebru Arısoy, Sergül Aydöre, Doğaç Başaran, Doğan Can, Oya Çeliktutan, Dr. Cem Demirkır, Çağlayan Dicle, Erinç Dikici, Dr. Helin Dutağacı, Bilgin Esmel, Neslihan Gerek, Arman Savran, Temuçin Som, Ekin Şahin, and Sinan Yıldırım.

This thesis is dedicated to my beloved family and parents. I would like to thank my parents, my sisters and my cousins for everything. I am grateful for their endless love and support.

Finally I would like to express my gratitude to Tümer. Thank you for being my best friend and husband with your endless support. You help me to pursue my dreams and make them a part of real life. Çınar, my brightest sunshine, thank you very much for your understanding with a mom who spends most of her time for her research.

## ABSTRACT

# AUTOMATIC ANALYSIS OF HEAD AND FACIAL GESTURES IN VIDEO STREAMS

Automatic analysis of head gestures and facial expressions is a challenging research area and it has significant applications for intelligent human-computer interfaces. An important task is the automatic classification of non-verbal messages composed of facial signals where both facial expressions and head rotations are observed. This is a challenging task, because there is no definite grammar or code-book for mapping the non-verbal facial signals into a corresponding mental state. Furthermore, non-verbal facial signals and the observed emotions have dependency on personality, society, state of the mood and also the context in which they are displayed or observed. This thesis mainly addresses the three desired tasks for an effective visual information based automatic face and head gesture (FHG) analyzer. First we develop a fully automatic, robust and accurate 17-point facial landmark localizer based on local appearance information and structural information of landmarks. Second, we develop a multi-step facial landmark tracker in order to handle simultaneous head rotations and facial expressions. Thirdly, we analyze the mental states underlying facial behaviors by utilizing time series of the extracted features. We consider two data representation types, namely facial landmark trajectories and spatiotemporal evolution data of the face image during an emotional expression. Novel and different sets of features are extracted from these face representations for the automatic facial expression recognition. Features can be landmark coordinate time series, facial geometric features or appearance patches on expressive regions of the face. We use comparatively, feature sequence classifiers: Hidden Markov Models and Hidden Conditional Random Fields, and feature subspace methods: Independent Component Analysis, Non-negative Matrix Factorization and Discrete Cosine Transform on the spatiotemporal data with modified nearest neighbor classifier. Proposed algorithms improves the state of the art performance results for both posed and spontaneous databases.

## ÖZET

# VIDEO GÖRÜNTÜLERİNDEN KAFA VE YÜZ MİMİKLERİNİN OTOMATİK ANALİZİ

Video görüntülerinden kafa hareketlerinin ve yüz ifadelerinin otomatik analizi akıllı insan bilgisayar arayüzlerinde önemli uygulamaları olan zorlayıcı bir araştırma alanıdır. Kafa hareketlerinin ve yüz ifadelerinin olduğu yüz sinyallerinden oluşan sözsüz mesajların otomatik olarak sınıflandırılması önemli bir görevdir. Bu görev zorlayıcıdır çünkü sözsüz yüz sinyallerinden zihinsel durum geçişini sağlayacak bir sözlük ya da kod çizelgesi tanımlı değildir. Dahası, sözsüz yüz sinyallerinin oluşumu ve yorumlanması kişiye, topluma, o anki ruhsal duruma ve bulunulan ortama göre değişmektedir. Bu tez görsel bilgiye dayalı yüz ve kafa hareketleri analizcisi için gerekli olan başlıca üç temel görevin çözümünü ele alır. İlk olarak tamamen otomatik, dayanıklı ve hassas yüz görseline ve yüzün yapısal bilgisine dayalı çalışan on yedi yüz noktası bulan bir algoritma geliştirdik. İkinci olarak, eş zamanlı oluşan kafa hareketlerine ve yüz ifadelerine dayanıklı çok basamaklı bir yüz nirengi noktası izleme algoritması geliştirdik. Üçüncü olarak, izlenen nirengi noktalarına dayalı çıkarılan özniteliklerin zaman dizileri kullanılarak zihinsel durumun altında yatan yüz davranışlarının analizi ele alındı. İki veri gösterimi kullanıldı, bunlar yüz nirengi noktası koordinat gezinmeleri ile ifade esnasında oluşan yüz imgesinin zamanuzamsal gelişim verisidir. Bu yüz gösterimleri kullanılarak yüz ifadelerini tanımak için yeni ve birçok kümeden oluşan öznitelikler çıkarılmıştır. Çıkarılan öznitelikler nirengi koordinatlarından oluşan zaman dizileri, yüz geometrik öznitelikleri veya yüzün anlatımsal bölgelerinden çıkarılan görüntü öznitelikleridir. Saklı Markov Modeller ve Saklı Şartlı Rastsal Alanlar gibi dizi sınıflandırıcıları ile Bağımsız Bileşenler Analizi, Negatif Olmayan Matris Çarpınlarına Ayırma ve Kesikli Kosinüs Dönüşümü gibi çeşitli alt-uzay izdüşüm yöntemleri karşılaştırma amaçlı kullanılmıştır. Önerilen algoritmalar pozlu ve doğal veritabanları üzerinde bugüne kadar kaydedilen en iyi performans değerlerine sahiptir.

## TABLE OF CONTENTS

ACKNOWLEDGEMENTS . . . . .	iii
ABSTRACT . . . . .	v
ÖZET . . . . .	vi
LIST OF FIGURES . . . . .	x
LIST OF TABLES . . . . .	xviii
LIST OF SYMBOLS/ABBREVIATIONS . . . . .	xxii
1. INTRODUCTION . . . . .	1
1.1. Outline . . . . .	5
2. LITERATURE REVIEW . . . . .	7
2.1. Background on Facial Landmark Detection . . . . .	7
2.2. Background on Facial Landmark Tracking . . . . .	15
2.3. Background on Mental and Emotional State Analysis . . . . .	18
3. FACIAL LANDMARK DETECTION . . . . .	24
3.1. Coarse and Fine Level Fiducial Landmark Detection . . . . .	25
3.1.1. IGF Based Features . . . . .	26
3.1.2. DCT Based Features . . . . .	30
3.1.3. Support Vector Machine Based Classification . . . . .	32
3.1.4. Databases and Testing Methodology for IGF-and DCT-Based Features . . . . .	34
3.1.5. Performance of IGF based and DCT based Localizers on FRGC-I	35
3.2. Regularization with Probabilistic Graph Models . . . . .	40
3.2.1. Face Localization on 2D/3D Face Images . . . . .	40
3.2.2. Improvement with Probabilistic Graph Models . . . . .	41
3.2.3. Probabilistic Graph Model - I ( <i>PGM – I</i> ) . . . . .	45
3.2.4. Probabilistic Graph Model - II ( <i>PGM – II</i> ) . . . . .	48
3.2.5. Performance of <i>PGM – I</i> and <i>PGM – II</i> on FRGC-I and FRGC-II	50
3.3. Final Adjustments for Automatic Facial Landmark Localizer . . . . .	54
3.3.1. Improvement in Search Region . . . . .	54
3.3.2. Ancillary Landmark Initialization . . . . .	54

3.4. Conclusions . . . . .	58
4. MULTI-STEP FACIAL LANDMARK TRACKING ALGORITHM . . . . .	61
4.1. Kalman Prediction . . . . .	61
4.2. Template Library . . . . .	65
4.2.1. Refinement with Template . . . . .	65
4.2.2. Template Update . . . . .	66
4.3. PCA Regularization with Multi-Pose Shape Model . . . . .	73
4.4. Facial Landmark Tracking Results . . . . .	79
4.4.1. Video database (BUHMAP) . . . . .	79
4.4.2. Landmark tracking results . . . . .	80
4.4.3. Contribution of the Tracking Steps . . . . .	82
4.5. Conclusions . . . . .	85
5. CHARACTERISTICS AND FEATURES OF FACIAL EXPRESSIONS AND HEAD GESTURES . . . . .	87
5.1. Dynamics of the Facial expressions and Head Gestures . . . . .	88
5.2. Data Representation and Feature Types . . . . .	88
5.2.1. Facial Landmark Coordinates . . . . .	90
5.2.2. Face Geometric Features . . . . .	92
5.2.3. Appearance based Features . . . . .	93
5.3. Alternative Data Representation and Feature Types . . . . .	96
5.3.1. Spatiotemporal prism of face sequences . . . . .	97
5.3.2. Features from spatiotemporal face cube . . . . .	98
6. CLASSIFIERS FOR FACIAL EXPRESSION AND HEAD GESTURE RECOG- NITION . . . . .	100
6.1. Feature - Sequence Classifiers . . . . .	100
6.1.1. HMM . . . . .	100
6.1.2. HCRF . . . . .	101
6.2. Feature - Subspace Classifiers . . . . .	103
6.2.1. Modified Nearest - Neighbor Classification . . . . .	108
6.3. Decision level fusion of classifiers . . . . .	109
7. RECOGNITION RESULTS FOR HEAD GESTURE AND FACIAL EXPRES- SION VIDEOS . . . . .	113

7.1.	Performance Results for BUHMAP Database . . . . .	115
7.1.1.	Results of Classifier Fusion . . . . .	122
7.2.	Classification Methodology for LILir TwoTalk Database . . . . .	125
7.3.	Performance Results for the LILir TwoTalk Database . . . . .	127
7.4.	Performance Results for Cohn-Kanade Facial Expression Database . . . . .	137
8.	CONCLUSIONS . . . . .	141
8.1.	Contributions of the Thesis . . . . .	141
8.1.1.	Reliable Facial Landmark Detection . . . . .	141
8.1.2.	Robustness by graph-based regularizer . . . . .	141
8.1.3.	Facial Landmark Tracker . . . . .	142
8.2.	Facial Expression and Head Gesture Analyzer . . . . .	143
8.2.1.	Generalization to Use Cases . . . . .	144
8.3.	Future Work . . . . .	144
	REFERENCES . . . . .	147

## LIST OF FIGURES

Figure 2.1.	(a)Search using an ASM of a face, (b)Search using ASM of a face, given a poor starting point. (Figures are taken from Cootes study [1]) . . . . .	10
Figure 2.2.	(a)Depth maps representation, (b) Point cloud representation (X, Y, Z coordinate vectors respectively, as a function of the vector index). Figures are taken from [2]. . . . .	14
Figure 2.3.	Some of the upper face action units and their combinations [3] . . . . .	19
Figure 2.4.	Representation of six basic emotions on static images: happy, angry, surprised, sad, disgusted, afraid . . . . .	19
Figure 2.5.	A dynamic stimulu . . . . .	20
Figure 3.1.	Illustration steps of facial landmark initialization. (a) Face detection, (b) Fiducial landmark detection: DCT-features, SVM and Probabilistic Graph Matching refinement as in [4], (c) Ancillary landmark initialization: Adaptation of the 17-point mesh, (d) Refinement of ancillary landmarks via template search and SVM . . . . .	25
Figure 3.2.	(a)Representation of Gabor wavelet filters at five scales and eight orientations , (b)The magnitudes of the Gabor representations of a face image . . . . .	28
Figure 3.3.	Overview of the IGF method both for coarse (a) and fine (b) localization . . . . .	29

Figure 3.4.	Two dimensional DCT basis functions ( $N = 8$ ), 2D basis are generated by multiplying the 1D basis functions for $N=8$ . . . . .	31
Figure 3.5.	IOD histogram of FRGC-I database . . . . .	33
Figure 3.6.	Sample images from FRGC-I database . . . . .	35
Figure 3.7.	$0.1d$ : accepted error distance for localization, $d$ : IOD . . . . .	35
Figure 3.8.	Tuning of the DCT method: results of the fine localization for each landmark type. Legend: DCT8-20 16-54 should be read as: 8x8 coarse level DCT transform and 20 out of 64 coefficients are selected; 16x16 fine level DCT transform and 54 out of 256 coefficients are selected . . . . .	36
Figure 3.9.	Comparison of fine refinement performances of DCT, Lades [5] and Wiskott [6] . . . . .	37
Figure 3.10.	Comparison of fine refinement performances of DCT, IGF and IMoFA-L based methods on 2D and 3D data. Legend: IGF7-100 11-150 should be read as: 7x7 coarse level IGF vector and dimension reduced from 7x7x12 to 100 via PCA; 11x11 fine level IGF vector and dimension reduced from 11x11x8 to 150 via PCA. IMoFA 2D+3D is a combination of the conspicuity maps obtained separately from 2D Gabor features and 3D depth image. . . . .	39
Figure 3.11.	General flowchart of coarse level localization (a) Score map extraction, (b) Choosing Probabilistic Graph model (PGM) . . . . .	41

Figure 3.12. Score maps for (a) right eye outer corner, (b) right eye inner corner, (c) nose tip, (d) left mouth corner. These images can be interpreted as feature-ness map on the face. The peaks have been accentuated for visibility. . . . .	43
Figure 3.13. (a) Chosen candidate points (four highest scored locations) for some of the facial landmarks, (b) A plausible landmark triple of the right eye outer corner (REOC), right eye inner corner (REIC) and right mouth corner (RMC) . . . . .	44
Figure 3.14. Scatter diagram of the three reliable landmarks (right eye inner corner, left eye inner corner and left eye outer corner) and of the four remaining landmarks vis-à-vis the basis three reliable landmarks	47
Figure 3.15. Face graph for PGM-II and sample quadrilateral for the right eye outer corner with its three support features. . . . .	49
Figure 3.16. Graph-based correction. Improvements of 2D feature accuracy with PGM-I and of 3D feature accuracy with PGM-II. . . . .	51
Figure 3.17. Improvement with the refining stage after applying the PGM methods	51
Figure 3.18. Contribution of 2D + 3D. First column: The location errors with pure 3D techniques are plotted on the vertical; those with the 2D-aided 3D localization error on the horizontal. Second column: Similar results for pure 2D vis-à-vis 3D-aided 2D localization. . . . .	52
Figure 3.19. Performance of landmark localization on face images acquired in similar conditions: 2D-Fine test results with PGM-II (training set: FRGC-II (fall 2003 session), test set: FRGC-II (spring 2004 session) . . . . .	52

Figure 3.20.	Detected face region via boosted Haar feature based face detector [7] and the corresponding subregions for coarse level landmark search.	55
Figure 3.21.	Main steps of automatic facial landmarking method . . . . .	55
Figure 3.22.	Distribution of ancillary landmark points (shown with $\mathbf{a}$ ) anchored on fiducial landmark points (shown with $\mathbf{f}$ ) . . . . .	57
Figure 3.23.	Sample 2D face images from Bosphorus Database[8] . . . . .	58
Figure 4.1.	The general flow diagram of the proposed landmark tracking algorithm . . . . .	62
Figure 4.2.	52 manually annotated landmarks (dots), 17 landmarks tracked by our algorithm (crosses) . . . . .	62
Figure 4.3.	White rectangles represent the chosen templates for outer-eye corner in an image sequence . . . . .	66
Figure 4.4.	Template Library Update Algorithm . . . . .	67
Figure 4.5.	Illustration of similarity scores in search windows on a sample face image. Solid box: search window; dotted window: a template in a test position; dash-dot window: template in optimal position. . . .	68
Figure 4.6.	Template Matching Algorithm . . . . .	69
Figure 4.7.	Template library information for sample videos:(a) Head-shaking video, (b) Head up and eyebrow raise video, (x-axis : frame index, y-axis : index of the chosen template) . . . . .	70

Figure 4.8.	Template library information for sample videos:(a) Head forward and eyebrow raise video, (b) Smiling video (x-axis : frame index, y-axis : index of the chosen template) . . . . .	71
Figure 4.9.	Sample face images from Bosphorus Database [8] illustrating various head poses and facial actions . . . . .	74
Figure 4.10.	Tracked frames with different pose models: right yaw, frontal or near frontal, left yaw . . . . .	74
Figure 4.11.	Lines: generated face shapes ( $b_i$ parameters are limit points), Circles: mean face shape for the frontal faces, (a) nose middle point is below the midpoint of upper lip, (b) inner eye and eyebrow points are very close to each other, (c) midpoint of upper and lower lip are almost overlapping and above the lip corners (d) mouth corners are almost on the same x-coordinate with outer eye corners and eyes and eyebrows are far away from each other. . . . .	77
Figure 4.12.	Generated face shapes under perturbations of single modes for near frontal faces. The first mode affects the width of the face , the second mode affects the yaw angle of the head, the third and fourth modes affect the pitch angles and mouth states (open or close) and the fifth and sixth modes affect the eyebrow movements of the frontal face shape model. . . . .	78
Figure 4.13.	Contribution of tracking steps to the landmark tracking accuracy illustrated on a sample frame of a gesture video . . . . .	79
Figure 4.14.	Cumulative performance of landmark errors vis-à-vis their ground-truth data. (Eyes: mean of the 4 eye points, Nose : approximated nose tip using tracked nose points, Mouth: mean of 4 lip points, Eyebrows: mean of the pair of outer and inner eyebrow points) . . . . .	81

Figure 4.15.	Mean Euclidean distance for each gesture class (averaged overall 17 landmarks) and the grand mean (averaged over gestures and landmarks) . . . . .	82
Figure 4.16.	Tracked landmarks on sample image sequences; 1 <sup>st</sup> row - <b>G1</b> ; 2 <sup>nd</sup> row - <b>G2</b> ; 3 <sup>rd</sup> row - <b>G3</b> ; 4 <sup>th</sup> row - <b>G4</b> ; 5 <sup>th</sup> row - <b>G5</b> ; 6 <sup>th</sup> row - <b>G6</b> ; 7 <sup>th</sup> row - <b>G7</b> ; . . . . .	83
Figure 4.17.	Contribution of tracking steps to the tracking performance in terms of AUCDC (H-LR head shaking; H-U head up and eyebrow raise; H-F head forward and eyebrow raise) . . . . .	84
Figure 5.1.	A facial expression sequence belonged to surprised emotion with time index $k$ . $k = 1$ to $k = 2$ represents the neutral state, $k = 3$ to $k = 5$ represents the onset phase, $k = 5$ to $k = 10$ represents the apex phase and $k = 10$ to $k = 12$ represents the offset phase) . . .	89
Figure 5.2.	(a) Sequential features and feature-sequence classifiers, (b) Subspace features and feature-subspace classifier . . . . .	91
Figure 5.3.	Some of the distances that are used for geometric feature extraction	92
Figure 5.4.	Facial patches and the corresponding blocks defined on a sample image . . . . .	95
Figure 5.5.	Representation of $A$ matrix which is composed of DCT features of image patches . . . . .	96
Figure 5.6.	Facial landmarks to be detected and the cropped face region. The cropped area is dimensioned according to the EID . . . . .	97

Figure 5.7.	Illustration of a spatiotemporal prism over a three-frame instance. There is a total of 21 blocks of size 16x16 extracted from the video shot. The blocks overlap by 50%. . . . .	98
Figure 6.1.	Graphical structure of a simple HMM ( $Y$ is the output label, $s$ is the state label and $X$ is the observation sequence) . . . . .	101
Figure 6.2.	Graphical structure for HCRF ( $Y$ is output label, $s$ is the hidden state labels and $X$ is the observation sequence) . . . . .	102
Figure 6.3.	Comparison of time normalized feature sequence and hidden state sequence after HMM . . . . .	104
Figure 6.4.	Histogram of the length of the gesture videos in the database . . . . .	105
Figure 6.5.	(a) Spatiotemporal trajectory matrices $P$ as gray level intensity images. (b) The upper half corresponds to x-coordinates of the landmarks and lower half corresponds to y-coordinates of the landmarks. . . . .	106
Figure 6.6.	Min and median distances of a test feature vector from 2 different gesture classes. . . . .	108
Figure 6.7.	Assigning a class label to a test sample . . . . .	109
Figure 6.8.	Score normalization process for two sample classifier scores . . . . .	112
Figure 7.1.	(a) Mean and standard deviation of classifiers over 7 gestures . . . . .	123
Figure 7.2.	Temporal feature extraction and classification for a 104-frame long sample clip. . . . .	128

Figure 7.3.	Normalized difference scores for a sample test set and the corresponding ROC curve . . . . .	131
Figure 7.4.	ROC of decision fusion for the classification of four categories via temporal window based classification. . . . .	136
Figure 7.5.	ROC of decision fusion for the classification of four categories via clip based classification . . . . .	136

## LIST OF TABLES

Table 1.1.	Desired properties from an effective HCI system . . . . .	3
Table 2.1.	Summary of recent 2D facial landmarking algorithms . . . . .	11
Table 2.2.	Summary of recent 2D facial landmarking algorithms (Table 2.1 continued) . . . . .	12
Table 3.1.	Average coarse and fine localization errors in terms of Euclidean pixel distances on the original images . . . . .	38
Table 3.2.	Performance of different window sizes for fine refinement . . . . .	50
Table 4.1.	Percentage of templates usage (%) . . . . .	72
Table 4.2.	Head and facial gesture classes in BUHMAP DB [9] . . . . .	80
Table 5.1.	Face geometric features derived from tracked landmarks (Euclidean distances) . . . . .	94
Table 5.2.	Face geometric features derived from tracked landmarks (Euclidean distances) (Table 5.1 continued) . . . . .	95
Table 5.3.	Patches and the corresponding block sizes . . . . .	95
Table 6.1.	Dimensions of the subspace projection matrices for BUHMAP database ( $r$ is the number of training samples) . . . . .	107
Table 6.2.	Comparison of NN, LDA and MNN methods on NMF features of trajectory matrix $P$ . . . . .	109

Table 7.1.	Meaning of the non-manual signs recorded in BUHMAP videos for TSL . . . . .	114
Table 7.2.	Nonverbal communication categories of LILir TwoTalk Corpus [10]	114
Table 7.3.	Test set and performed experiments . . . . .	116
Table 7.4.	HMM with landmark trajectory features $(34,P)$ . . . . .	117
Table 7.5.	HCRF with with landmark trajectory features $(34,P)$ . . . . .	117
Table 7.6.	HMM with geometric features $(17,G)$ . . . . .	117
Table 7.7.	HCRF with geometric features $(17,G)$ , $w=1$ . . . . .	118
Table 7.8.	HMM with DCT-based appearance features (75 block DCT coefficients, $A$ ) . . . . .	118
Table 7.9.	HCRF with DCT-based appearance features (75 block DCT coefficients, $A$ ) . . . . .	118
Table 7.10.	MNN results with DCT features from landmark trajectory matrix $P$	119
Table 7.11.	MNN results with ICA features from landmark trajectory matrix $P$	120
Table 7.12.	MNN results with NMF features from landmark trajectory matrix $P$	120
Table 7.13.	MNN results with NMF features from geometric feature matrix $G$	120
Table 7.14.	MNN results with NMF features from DCT-based appearance matrix $A$ . . . . .	121

Table 7.15.	MNN results with ICA features from DCT-based appearance matrix $A_{121}$	
Table 7.16.	Designed classifiers . . . . .	121
Table 7.17.	Comparison of Fusion Techniques (Each time one classifier is removed - PV: Plurality Voting, BC: Borda Count, WBC: Weighted Borda Count, MIN: Minimum, MAX: Maximum, WSUM: Weighted Sum, PROD: Product) . . . . .	124
Table 7.18.	Best decision fusion results . . . . .	125
Table 7.19.	Calculated correlation coefficients using the average ratings for each category (taken from [11]) . . . . .	127
Table 7.20.	Data types, extracted features and the classifiers for the LILir database	129
Table 7.21.	Experiment sets for LILir Database . . . . .	130
Table 7.22.	Classification performances averaged over non-verbal message clips	133
Table 7.23.	Classification performances averaged over temporal windows . . . . .	133
Table 7.24.	Comparison of average classification performances over message categories for two classification paradigms . . . . .	133
Table 7.25.	Comparison of decision fusion results and Sheerman-Chase's results [11] . . . . .	135
Table 7.26.	Performance comparison of clip based decision fusion classification and human classification [11] . . . . .	137
Table 7.27.	Comparative results of individual classifiers . . . . .	138

Table 7.28.	Confusion matrix of 6-class facial expression recognition after decision fusion of five classifiers (overall recognition rate is 95.34) . . .	138
Table 7.29.	Comparison with other results from the literature. Since experimental conditions vary slightly, with each method we are reporting the number of subjects, the number of sequences and the number of classes analyzed(SI:Subject Independent, P:# of person, S:# of sequence, C:# of class) . . . . .	139

## LIST OF SYMBOLS/ABBREVIATIONS

$a$	Ancillary landmarks
$A$	Appearance based DCT coefficient matrix
$A_s$	State transition matrix
$b$	Shape parameters
$B$	Number of blocks in the face prism
$C$	Discrete coefficient matrix
$C$	Shape covariance matrix
$d$	Inter ocular distance
$D$	Spatiotemporal data matrix
$E$	Eigenvector matrix
$f$	Fiducial landmarks
$F$	Independent source signals
$G$	Geometric feature matrix
$G1$	Head shaking
$G2$	Head up
$G3$	Head forward
$G4$	Sadness
$G5$	Head up-down
$G6$	Happiness
$G7$	Happy up-down
$H$	NMF-based feature vectors for the training data
$j$	Landmark index
$k$	Time index
$l$	Length of video clips in seconds
$k_v$	Scale parameter
$m$	Number of reliable landmarks
$M$	Mixing matrix
$M^+$	Pseudo inverse of mixing matrix
$N$	Normal distribution

$NCC$	Normalized cross correlation
$P$	Landmark trajectory matrix
$p(x, s)$	Joint probability distribution function
$p(x s)$	Conditional probability distribution function
$Pro$	Subspace projection
$q$	Test measurement
$Q$	Training data matrix
$R$	Measurement noise covariance matrix
$S$	Intermediate hidden state for sequence classifiers
$\mathbf{S}$	Shape data matrix
$\bar{\mathbf{S}}$	Mean face shape
$t$	Time index
$T$	Number of frames for a video shot
$T_{test}$	Test template
$T_{library}$	Template library
$V$	Face prism
$v_{(k)}$	Measurement noise
$v$	Scale parameter for Gabor wavelet
$\mathbf{W}$	Basis matrix for NMF
$\mathbf{W}^+$	Pseudo inverse of basis matrix for NMF
$w$	Orientation parameter for Gabor wavelet
$w_{(k)}$	Process noise for Kalman filter
$x$	Landmark x-coordinate
$\hat{\mathbf{X}}$	Generated face shape
$X$	Observation sequence
$\chi$	Gabor feature vector
$X_{(k)}$	State vector
$y$	Landmark y-coordinate
$Z$	Independent Gabor feature vector
$\delta$	Dirac delta function

$\delta_x$	Displacement in x direction
$\delta_y$	Displacement in y direction
$\Delta_{AUCDC}$	Difference of areas under cumulative distribution curves
$\lambda$	Length
$\Lambda$	Mean length matrix
$\phi$	Mean relative angle
$\Phi$	Covariance matrix for the relative angle
$\rho$	Eigenvalue matrix
$\Psi$	Gabor wavelet
$\xi$	Optimal separating hyperplane parameter
$\varphi_w$	Orientation parameter
$\varsigma$	Process noise variance
$\zeta$	Process noise covariance matrix
AAM	Active appearance model
ARA	Average recognition accuracy
ASM	Active shape model
AU	Action unit
<i>AUCDC</i>	Area under cumulative distribution curve
BC	Borda Count
BUHMAP	Bogazici University Head Motion Analysis Project
CRF	Conditional Random Fields
DCT	Discrete Cosine Transform
EID	Distance between eye inner corners
FACS	Facial Action Coding System
FAU	Facial Action Unit
FER	Facial Expression Recognition
FHG	Face and Head Gesture
FRGC	Face Recognition Great Challenge
GWN	Gabor Wavelet Network
Happy U-D	head nodding with a smile

HCI	Human Computer Interaction
HCRF	Hidden Conditional Random Fields
Head U-D	Head nodding
H-F	Head forward eyebrow raise
H-LR	Head left-right
HMM	Hidden Markov Models
H-U	Head up eyebrow raise
ICA	Independent Components Analysis
IGF	Independent Gabor Features
Im	Facial image
IMOFA-L	Independent Mixture of Factor Analysis
IOD	Inter Ocular Distance
LDA	Linear Discriminant Analysis
LOSO	Leave one subject out
MAP	Maximum a posteriori estimation
MAX	Maximum
MIN	Minimum
MLP	Multi Layer Perceptron
MNN	Modified Nearest Neighbour
NCC	Normalized cross correlation
NMF	Non-negative Matrix Factorization
NN	Nearest Neighbour
OSH	Optimal Separating Hyperplane
PCA	Principal Components Analysis
PDM	Point Distribution Model
PGM	Probabilistic Graph Models
PROD	Product
PV	Plurality Voting
REIC	Right Eye Inner Corner
REOC	Right Eye Outer Corner
RMC	Right Mouth Corner

SVM	Support Vector Machines
TSL	Turkish Sign Language
TW	Temporal window
UND	University of Notre Dame
WBC	Weighted Borda Count
WSUM	Weighted sum

## 1. INTRODUCTION

The exploration of new human-computer interfaces has become a growing field in computer science, which aims to create more natural, intuitive, unobtrusive and efficient interfaces. This objective has become more relevant with the introduction of intelligent machines. These are increasingly more popular in the roles when interpretation of the user's affective states is required. Such interfaces can adapt and respond to the user's need better and can benefit from the knowledge of human-human communication [12, 13, 14].

Human face is a rich source of nonverbal information. Indeed, not only it is the source of identity information but it also provides clues to understand social feelings and can be instrumental in revealing mental states via social signals [15, 16]. Facial expressions form a significant part of human social interaction [17]. While communicating, we express ideas that are visualized in our minds by using words integrated with nonverbal behaviors. It is stated that about 55% of interpersonal feelings and attitudes, such as like and dislike, can be conveyed via facial expressions [18]. Therefore when the body language and verbal messages are used in complementary roles, our messages can be more clear and can be conveyed more accurately. Face then functions as a channel in communicating the emotional content of our messages. Gestures, eye and head movements, body movements, facial expressions and touch constitute the nonverbal message types of our body language. These non-verbal messages can be more instrumental than words in revealing our true mental states and feelings such as trust, agreement, enjoyment, hostility and worry.

An affective interface is defined as an interface that appeals to the emotional state of users and allows users to express themselves emotionally. We can say that, in human communication the non-verbal messages are interpreted in terms of affective states. The term of "Affective Computing" was introduced by R. Picard [19], a pioneering researcher on affective computing at MIT. It has some similarity with emotional intelligence but affective computing studies the single model or multi model

characteristics of physical and mental events occurring during verbal and nonverbal communication. Therefore the main task of affective computing is to provide an effective and intelligent interface between humans and machines [19, 20]. Consequently, empowering computers with the capability to recognize and to respond to nonverbal communication clues is important [13, 14, 21, 22].

Because of its importance, automatic human affect analysis has attracted the interest of many researchers in the last decades. Some of the current examples in affective interfaces include the following applications: (i) Fatigue and drowsiness detection [23, 24, 25], (ii) Prediction of frustration in intelligent tutors, (iii) Affective multimodal interfaces for office scenarios [26]. Applications of affective interfaces can be extended to potential commercial applications such as affect-sensitive systems for customer services, call centers, and game and entertainment industries [13].

As Kaliouby stated [27], for an effective and naturalistic Facial Expression Recognition (FER) system in a Human-Computer Interaction (HCI) context a number of characteristics should be satisfied as listed in Table 1.1. Even though these requirements are satisfied there may be still many challenging factors for automatically analyzing the facial expressions. For example, interpretation of emotional states of persons by observing changes in visual appearance is not an easy task not only for computers but also for humans. In fact, emotional states or feelings cannot be directly observed by another person. Only revealed emotional expression or “symptoms” can be exploited to infer the underlying mental state [28]. Another challenging problem is that sometimes different individuals exhibit different physiological responses to the same emotional state or similar physiological responses can be observed under different mental states. Therefore, interpretation of mental states is context dependent. For example, a smile can be elicited to show politeness, irony, joy, or greeting [13]. There is also one more challenge, that there is no definite grammar or code-book for mapping the affective states into a corresponding mental state category. Therefore, due to these challenges encountered in computer and cognitive sciences, interpretation of mental states is still an open problem for both disciplines.

Table 1.1. Desired properties from an effective HCI system

Characteristic	Criteria
Diversity of mental states	Supports basic emotions and complex mental states
Fully automated	No manual pre-process
Real time	No delayed responses from the computer
Rigid head motion	Capable of non-frontal poses
Continuous and asynchronous expressions	Able to process overlapping, asynchronous expressions
User-independent	Able to process novel subjects
Independent from neutral expression	Neutral expression not required
Talking heads and hand gestures	Supports natural communication streams (with speech and hand gestures)

Typical tasks to be expected from a visual-appearance based affective interface can be listed as follows: *(i)* Detection and localization of the face; *(ii)* Accurate facial feature detection and tracking; *(iii)* Facial expression analysis; *(iv)* Recognition of mental states from sequences involving face expressions and head movements. Despite advances in all of these tasks development of affective interfaces is still an open problem. Each listed task has its own limitations and challenges. This thesis focuses on each of these tasks except the face detection and localization. We consider face detection practically solved. For example, Viola and Jones [29] have introduced a very promising face detector which uses boosted cascade of simple classifiers based on Haar wavelet features. Many enhanced face detection and feature extraction algorithms are inspired or adapted from this approach [7, 30, 31, 32, 33]. All these variations of facial image analyses exploit the basic idea of Viola and Jones face detector or a modified version [34, 35, 36, 37].

There is room for improvement in automatic emotional state analysis for HCI systems. Most of the existing techniques developed for emotional state analysis systems

somewhat insufficient because of the following reasons: (i) The need for manual labeling of facial landmark initialization [38, 39], (ii) Use of static images only [36, 40], (iii) Inadaptability of the systems to head rotations (limited tolerances for yaw, tilt, and roll rotations) [30, 40], (iv) Limited to a few basic emotions [35, 37, 38, 40, 41]. Therefore, the main focus of this thesis is to address these drawbacks and bring in the necessary improvements.

We can list the major contributions and the novelties of this thesis as follows:

- (i) The first contribution of this thesis is a novel automatic facial landmark detector. We have developed a fully automatic, robust and accurate facial landmark localization algorithm based on local appearance information and structural information of landmarks. A novel method for structural correction method namely: Probabilistic Graph Models are introduced. Instances of detected landmarks are corners of eyebrows, eyes, mouth and nose (totally 17 facial points).
- (ii) The second contribution is an automatic and robust facial landmark tracker. We have developed a multi-step facial landmark tracking algorithm successfully in order to handle simultaneous head rotations and facial expressions.
- (iii) The third contribution is a fully automatic face expression and head gesture classification system based on the spatiotemporal features extracted from the visual recordings of non-verbal messages. Our scheme for the automatic analysis of a subject's FHG throughout video sequences, exploits the proposed automatic facial landmark detection and tracking algorithms. In FHG analysis, we consider two data representation types, namely facial landmark trajectories and spatiotemporal evolution data of the face during an emotion expression. Novel discriminative features are extracted from these face representations for the automatic facial expression recognition. The main focus of this study is to assess the benefits of the proposed feature extraction algorithms vis-à-vis alternative schemes in the literature with the ultimate goal of automatic emotional state classification.

To clarify the terminology, the following terms: “affective state”, “emotional state”, “mental state” and “non-verbal message” are used interchangeably within this

thesis.

## 1.1. Outline

This thesis is organized as follows: In Chapter 2, we review tasks to be performed within an automatic face expression and head gesture analysis system. A comprehensive literature survey of each task, namely facial landmark detection and localization, facial landmark tracking and facial expression and mental state analysis, is presented.

In Chapter 3, a novel automatic facial landmark detection algorithm is described. In the first tier seven fiducial facial landmark points are detected and then in the second tier ten ancillary landmarks are found. This scheme forms a 17-pt facial landmark detector with satisfactory localization performance. The performance statistics of the algorithm are described under realistic conditions.

In Chapter 4, we present the multi-step facial landmark tracking algorithm. The tracking algorithm is basically composed of appearance and model-based four successive steps namely: landmark location prediction using Kalman filter, block matching with template library, regularization with multi-pose shape models and a final refinement. A multi-step tracking algorithm is described in order to handle changes in the geometry of the tracked landmark points under head rotations (pitch and yaw angles up to  $\pm 45^\circ$ ) and facial expressions.

Chapter 5 investigates the inherent dynamics of the facial expressions and head gestures in order to extract effective and discriminative data representations types and features. Data representation types and the corresponding features are also explained in this chapter.

In Chapter 6, we describe two types of classifiers for inference of complex mental states from facial expression and head gesture video. First of them is the feature - sequence classifiers such as Hidden Markov Models (HMM) and Hidden Conditional Random Fields (HCRF). The other classifier type is the feature - subspace classifiers,

which are based on subspace projection methods such as Discrete Cosine Transforms (DCT), Independent Component Analysis (ICA) and Non-negative Matrix Factorization (NMF). A modified nearest neighbor (MNN) classifier is also presented for subspace-based features in this chapter.

Chapter 7 describes the video databases that are used for performance evaluation of the proposed FHG classifier. The organization of the test sets and experiments are presented in this chapter.

Finally, Chapter 8 concludes this thesis with a summary of our achievements and the future directions for our research.

## 2. LITERATURE REVIEW

In this chapter, we present the research studies from low level facial image analysis such as facial landmark detection and facial landmark tracking to high level interpretation of mental and emotional states of humans including gestures, facial expressions and complex mental states by analyzing the visual cues extracted from the facial images.

### 2.1. Background on Facial Landmark Detection

The accurate detection and localization of the face and of its features is instrumental for the successful performance of subsequent tasks in related computer vision applications. Many high-level vision applications such as facial feature tracking, facial expression analysis, and face recognition, require reliable landmark localization. Facial feature points are referred to in the literature as “salient points”, “anchor points”, or “facial landmarks”. The most frequently occurring fiduciary facial features are the four eye corners, the tip of the nose, and the two mouth corners. Additionally, the eyebrows, the bridge of the nose, the tip of the chin, and the nostrils are sometimes used as facial landmarks.

Facial feature detection is a challenging computer vision problem due to high inter-personal changes (gender, race), the intra-personal variability (pose, expression) and acquisition conditions (lighting, scale, facial accessories). The desiderata for facial feature localization list as follows: *i*) Accurate; *ii*) Precise within a few millimeters; *iii*) Computationally feasible since most systems must run in real time; *iv*) Robust against pose, illumination, expression and scale variations, as well as against occlusions and disturbance from facial accessories; *v*) Amenable to dynamic tracking.

In the literature there are various approaches for facial feature localization basically based on three approaches as appearance-based, geometry-based and structure-based. The majority of approaches use a preprocessing stage for initial coarse localization, using horizontal and vertical gray-level [42], or edge field projections, fol-

lowed by some histogram valley detection filter [43]. A second commonality between methods is that most use a coarse-to-fine localization to reduce the computational load [4, 44, 45, 46, 47, 48, 49, 50]. Some algorithms employ a skin colour-based scene segmentation to detect the face first [45, 51, 52], and further colour segmentation for lip detection [53, 54].

A common feature detection approach is to extend a whole face detection method, to search for smaller facial features at a higher resolution. For example Feris et al. [46] use Gabor Wavelet Networks (GWNs) to first find the approximate face region and then use smaller GWNs to look for 8 individual features, namely the corners of the eyes, the nostrils and mouth corners.

Appearance-based approaches aim to find basis vectors to represent the face and its facial features. Examples of transformations used are principal components analysis (PCA) [47, 55], Gabor wavelets [46, 48, 56], ICA [55, 57], DCT [58] and Gaussian derivative filters [44, 59]. These transform features capture and model facial features under statistical variability when selected and processed with machine learning techniques like boosted cascade detectors [31, 32, 33], support vector machines (SVM) [55, 60, 61], and multi-layer perceptrons (MLP) [47, 62].

Geometric-based methods use prior knowledge about the face position, and constrain the landmark search by heuristic rules that involve angles, distances, and areas [56, 58]. Structural information is an extension of geometric information that is used in validating localized features. For example, Wiskott et al. [6] analyze faces with a graph that models the relative positions of fiducial points, and a set of templates for each landmark for feature response comparison. They use an elastic bunch graph, where graph edges indicate the distances between the fiducial points; and graph nodes indicate Gabor wavelet responses. A number of templates (called the bunch) are used to test local feature responses. Smeraldi et al. [48] used retinotopic grid, where each node represents a vector composed of responses of the Gabor filters. Localization based on a saccadic search is completed by using the relative position information of the facial features.

Other approaches that use the structural information in addition to local similarity enable more flexible graph transforms to represent displacements of fiducial point positions, and search these positions to maximize feature responses under landmark configuration constraints [45, 63, 64]. For these models, the optimization process is plagued by local minima, which makes a good and often manual initialization necessary.

Active Shape Models (ASM) [63] are very popular approaches which combine the shape statistics and local feature detectors. The shape is learnt from a set of manually landmarked images using the shape statistics of Dryden and Mardia [65]. A profile model is trained on a linear patch normal to the model boundary through each model point. To improve the efficiency and robustness of the ASM algorithm, it is implemented in a multi-resolution framework. This involves first searching for the object in a coarse image, then refining the location in a series of successively finer resolution images. ASM needs sufficiently accurate starting position; otherwise it will converge to incorrect solutions. Figure 2.1-a illustrates a successful ASM search, but Figure 2.1-b demonstrates how the ASM can fail if the starting position is too far from the target. Since ASM is only searching along profiles around the current position, it cannot correct for large displacements from the correct position.

Active Appearance Model (AAM) [64] differs from the ASM in that instead of searching locally about each model point, it seeks to minimize the difference between a new image and one synthesized by the appearance model. The AAM combines shape and texture in a PCA space, then searches a new image iteratively by using the texture error to drive the model parameters. If a good enough initialization is given, the AAM converges to the solution, but is otherwise prone to the local minima. Christinacce et al. [32] also used a modified version of AAM to refine the predicted feature points. Xue et al. [66] proposed a Bayesian Shape Model which serves as a framework to solve the global and local deformations for facial feature extraction problem. The Bayesian Model (BM) expresses the matching of a deformable model to the object in a given image as a maximizing a posteriori (MAP) estimation problem. The coarse matching is adopted from the ASM's [63]. It is same as ASM except that, it uses edge profiles of

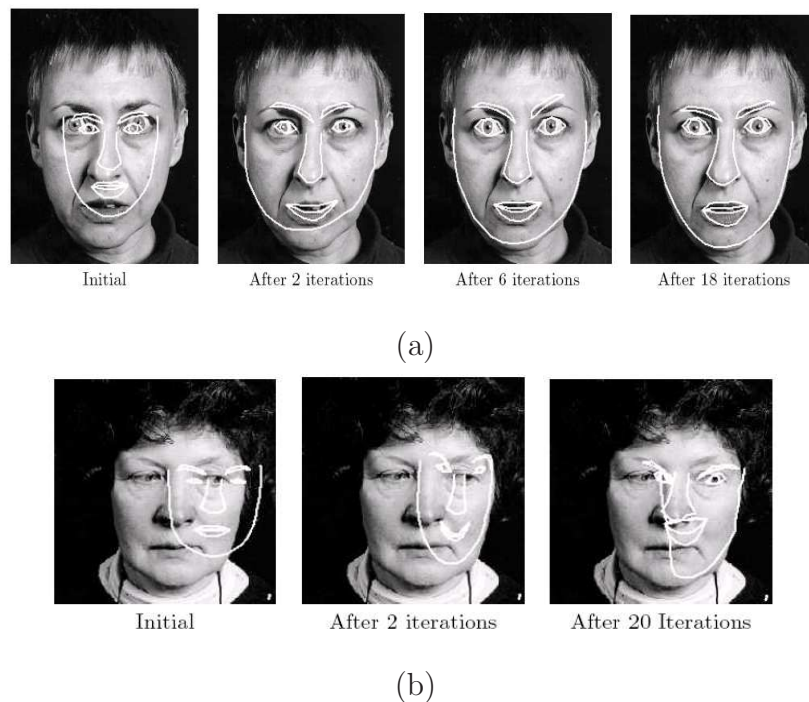


Figure 2.1. (a)Search using an ASM of a face, (b)Search using ASM of a face, given a poor starting point. (Figures are taken from Cootes study [1])

the boundary points in the tracking algorithm. Additionally in fine matching stage the deformation between image contour and the prototype contour is taken into account.

As a comparison of ASM and AAM, it is asserted that the ASM is faster and achieves more accurate feature point location than the AAM [1]. However, as it explicitly minimizes texture errors, the AAM gives a better match to the image texture. As a statistical model, both ASM and AAM gives the reconstructed object model that matches the probe image best. For that reason, it may not be able to match to a new image accurately if the variations of shape and appearance are not wide enough.

Table 2.1 summarizes some of the facial feature extraction methods in 2D face images.

Salah et al. [50] also used the combination of appearance model and structural model based method, called IMoFA-L, for robust facial landmark localization. In their approach face images are convolved with Gabor wavelets with 8 orientations and a single scale. Then the Gabor features of each channel is modeled via a generative

Table 2.1. Summary of recent 2D facial landmarking algorithms

Cristinacce et al. [32]	Assumed given	Boosted Haar wavelet-like features and classifiers	Training data: 1055 images / Test Data: BIOID dataset [67]
Smeraldi et al. [48]	30 dimensional Gabor response of each point + SVM	Gabor responses of the complete retinal field + SVM	M2VTS & XM2VTS [68]
Feris et al. [46]	GWN representation of faces	Template matching using Hierarchical Gabor Wavelet Network (GWN) representation of features	Yale and FERET Face Databases [69]
Ryu [47]	Vertical and horizontal projections of face edge map	PCA on coordinates of the feature edge map + MLP for template matching	ORL
Shih [56]	Edge projections + geometric model of facial features	Not present	Feret Face Database [69]
Arca et al. [44]	Color segmentation (Skin and lip) + SVM	Geometrical heuristics	1180 images of XM2VTS & 400 images of UniMiDb databases

Table 2.2. Summary of recent 2D facial landmarking algorithms (Table 2.1 continued)

Zobel et al. [58]	Geometrical heuristics on DCT coded images + Probabilistic model, based on coupled structured representation of feature locations	Not present	335 images of 20 subjects
Gourier et al. [59]	Gaussian derivatives $(G_x, G_y, G_{xx}, G_{xy}, G_{yy})$ + clustering to 10 centroid	Not present	
Antonini et al. [55]	Corner detection	Feature extraction using PCA and ICA projections of windows surrounding the corner points + SVM for template matching	BANCA database [70]
Pantic et al. [31]	horizontal and vertical intensity projections on partitioned face region	GentleBoost classification of 48 Gabor banks and gray scale values	Cohn-Kanade Database [71]
Eckhardt et al. [33]	face detection with multiscale sliding-window based GentleBoost detector	features are localized given the face context with multiscale sliding-window based GentleBoost detectors	Feret [69], BIOID [67] and Genki-4K [33] Databases
Zhu et al. [72]	A 28 point face mesh is imposed on the face based on Adaboost detected eye locations	Gabor coefficient vectors are searched via the fast phase-based displacement estimation	Not present

factor analysis mixture (IMoFA). Then feature similarity score maps are generated. Highest scored points are selected as the coarse locations of the landmarks and the fine-tuning is performed on the higher resolution 2D images using the 3D range image. A structural correction step is exploited to detect and correct mislocated landmarks.

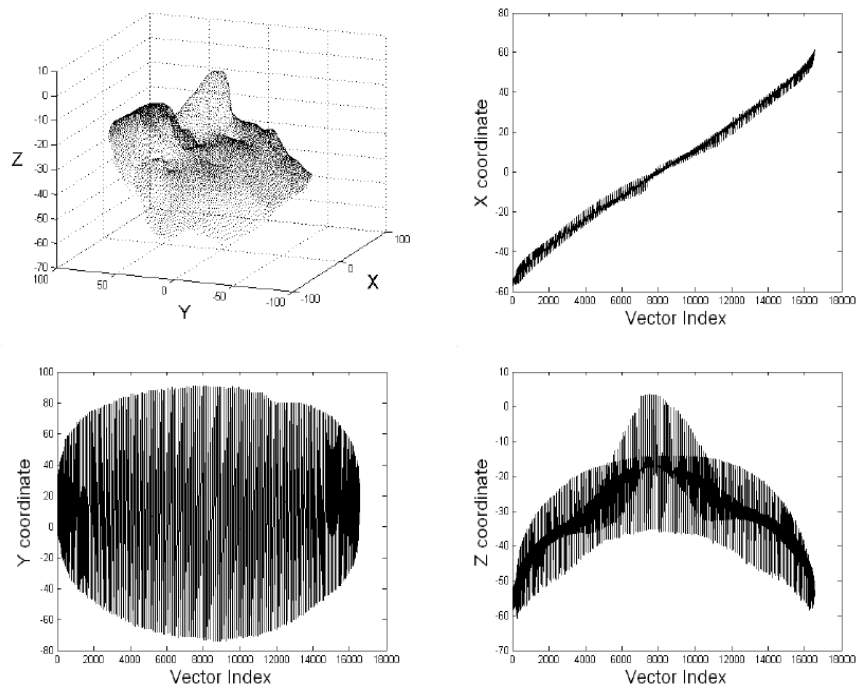
In study [73] a method for eye and mouth detection and eye center and mouth corner localization, based on geometrical information, was presented. The edge map of the detected face region was calculated. The distance vector field of the face was extracted by assigning to every facial image pixel a vector pointing to the closest edge pixel. The x and y components of these vectors were used to detect the eyes and mouth regions. Gray-level intensity information was used for eye center localization based on the fact that iris of the eye is the darkest area in the eye region, whereas the hue channel of the lip area was used for the detection of the mouth corners.

2D landmarking methods are mostly sensitive to pose variations in that they may fail for yaw degrees beyond  $\pm 30^\circ$  and illumination changes effect the gray scale (or RGB) values of the pixels nonlinearly. Recently 3D sensors, which provide range images with registered 2D color images, have become more widespread in facial feature analysis applications. Utilizing 3D face images make systems robust to illumination and pose variations. Therefore 3D information can be effective in order to improve the performance of face landmarking methods on both 2D and 3D images. Three different representation schemes can be extracted from the 3D range data such as depth map and point clouds as illustrated in Figure 2.2. Major 3D facial landmarking techniques are mostly structure-based. They combine the local appearance-based (2D) or local shape based (3D) features with parametric model of feature locations [4, 45, 74].

Lu et al. [75] assume the nose as the highest peak along the mid-line (the cross-section between the facial surface and the symmetry plane). A 3D statistical feature location model is used to reduce the search region. Shape index from 3D and corner-ness response from the intensity image are combined to extract the facial features. In [51], face is segmented by using skin color information, and range image is used to remove the background and to detect the nose location. 3D distances of facial points



(a)



(b)

Figure 2.2. (a)Depth maps representation, (b) Point cloud representation ( $X$ ,  $Y$ ,  $Z$  coordinate vectors respectively, as a function of the vector index). Figures are taken from [2].

are used to constraint the search area. Wang et al. [76] extract signature points from 3D range image and Gabor filter responses from 2D image for each image point. A feature vector containing both 2D and 3D features is used for matching. In [45], a statistical model of the inter-anchor point distances on 3D faces is used to estimate the locations of the feature points as bounding boxes. The algorithm initially detects the top of the head and estimates the locations of the bounding boxes. Exact locations of the feature points are found by using shape indexes as local shape characteristics. Romero et al. [77] propose two methods for 3D facial landmark localization. In the first method, structural graph matching algorithm eliminates the unlikely candidates using a “distance to local plane” node property and an “Euclidean distance” arc property. Best supported triplet combination, which corresponds to inner eye corners and nose tip, is selected by an exhaustive search. In the second method initially the nose tip is localized, using feature descriptors in a cascade filter. The inner eye corners are then localized relative to the nose tip. SSR descriptor, which are obtained by sampling the RBF function in the locality of a candidate facial vertex in a pose-invariant way, gives the best localization results.

## 2.2. Background on Facial Landmark Tracking

Facial landmark tracking is a challenging research area because rigid head movements, non-rigid facial surface deformations due to expressions, varying illumination conditions and occlusions cause significant changes on the appearance and configuration of tracked landmarks. Methods in the literature on facial landmark tracking fall into two categories of appearance-based [78, 79, 80] and model-based approaches [4, 81, 82, 83, 84]. The appearance-based approaches are general-purpose point trackers without the prior knowledge of the intent. Facial landmarks are tracked by locally searching for the best matching position, around which the appearance is most similar to the one in the previous frame [78]. Feris and Cesar Junior [79] represent face template as a linear combination of 52 continuous 2D odd-Gabor wavelet functions with different scale, translation and orientation parameters. The face image is affinely repositioned in the subsequent frames and the facial landmarks are effectively tracked when subjected to the same affine transformation. However the appearance-based ap-

proaches are susceptible to tracking errors due to face orientations, occlusions and strong facial expressions, that is in situations where the facial landmark appearance changes drastically. Buenaposada et al. [80] introduce a subspace representation of facial appearance that can be automatically trained and separates facial expressions from illumination variations. The appearance of a face is represented by the addition of two approximately independent linear subspaces modelling facial expressions and illumination respectively. The system requires two types of image sequences. One of them is facial expression sequence subject to all possible illuminations and the other sequence adopts all facial expressions under one particular illumination in the other.

In contrast, model-based approaches concentrate on explicitly modeling the shape of face in addition to appearance information, and consequently landmarks can benefit from some holistic information. The extensive literature on model-based facial landmark detection and tracking algorithms such as ASM [63], AAM [64] is a witness to the viability of the latter approach.

McKenna et al. [81] proposed an approach to track rigid and non-rigid facial motion based on a point distribution model (PDM) and Gabor wavelets. The prototypical model-based approach was the ASM approach [63]. This statistical model for deformable objects allowed shapes deformation only within the constraints of the learned patterns via PCA. The AAM [64] approach combines constraints of both shape variation and texture variation. Thus the best match between the model and the target image is achieved, whenever discrepancies in both the shape and texture residuals are minimized. Dornaika and Davoine [83] proposed a framework that utilizes online appearance models for 3D face and facial feature tracking. A hierarchical multi-state pose-dependent approach for facial feature tracking was proposed by Tong and et al. [85]. In their work, a multi-state statistical face shape model was used to characterize both the global shape of human face and the local structural details of each facial component across different facial expressions. In the first stage, the global feature points are matched; in the second stage, multi-state local shape models (open, closed and tightly closed states for the mouth, open and closed states for the eyes and one state for eyebrow and nose components) are utilized to handle facial expression changes.

Cristinacce and Cootes [82] utilized a set of feature templates for landmark detection and tracking with a shape constrained search. For each training image feature templates, the cropped rectangular patches around manually annotated facial landmarks, and the corresponding shape (landmark configuration) parameters were stored. For an unseen test image, current shape parameters were obtained and compared with the stored training shape parameters. Feature templates of the  $K$  closest matching shapes were compared with the texture sampled from the test image. The best matching training feature templates were then used to form detectors for each facial feature. The parameters of a statistical shape model are optimized to maximize the sum of responses.

Kanaujia et al. [84] proposed a framework to detect facial emblems such as head nodding, head shaking and eye blinking by tracking facial landmarks. Facial landmarks were tracked using specified ASM based on NMF instead of PCA. Multi-pose shape models are learned for different aspects of frontal, head rotated left, right, down and up in order to handle large poses variations. It was stated that the landmark configuration subspace learned via NMF can represent localized shape deformations better for facial expression tracking.

Tsalakanidou et al. [86] designed a 3D face tracker by employing 2D and 3D images recorded by a structured light sensor. 81 facial points were tracked by using appearance information and a global 2D+3D ASM trained via 3D facial point coordinates. In addition special trackers were developed for the mouth and the eyebrows using local 3D ASMs in order to model local appearance changes. They computed 23 geometric, appearance and surface deformation measurements using the estimated positions of the 81 landmarks. These measurements were used to recognize four facial expressions (disgust, happy, sad, surprise) and 11 facial action units using a rule-based approach.

Each of these two paradigms has its own strength and limitations and they may perform with mixed success in real world applications. For example, appearance based models suffer from significant changes in the face due to large head rotations, strong

facial expressions, occlusions and varying illumination. Model-based approaches compensate for these adversities utilizing prior knowledge of face for effective landmark tracking. However model-based approaches require more training data in order to compensate large head rotations and facial expressions for a robust and accurate tracking.

Our proposed method represents an improved hybrid method between model-based and appearance-based paradigms for robust and accurate tracking of facial landmarks on face video streams. First, a multi-pose landmark appearance deformable shape model is trained in order to handle shape variations due to varying head rotations. Then facial expressions are handled by a multi-state library of landmarks appearance-based templates which adapts to the scene.

### 2.3. Background on Mental and Emotional State Analysis

Most of the work in the literature on facial expression analysis is focused on the six basic emotions, i.e., happiness, surprise, sadness, fear, anger and disgust [22, 40, 87, 88, 89, 90]. The majority of facial expression recognition systems attempt to identify Facial Action units (FAUs) [27, 85, 86, 87, 88, 91, 92] based on Facial Action Coding System (FACS) [93]. Ekman and Friesen [93] proposed the FACS system for describing prototypic expressions of emotions by action units (AUs) or their combinations. In FACS, the facial behavior is decomposed into 46 action units, each of which is anatomically related to the individual facial muscles. Although they only define a small number of distinctive AUs, different combinations of AUs can be sufficient for accurately detecting and measuring a large number of facial expressions. Some of the upper face action units are illustrated in Figure 2.3. However, AUs may not provide a direct mapping between facial actions and mental states.

Facial expression recognition (FER) can be carried out by either static images or dynamic image sequences. Static stimuli consist of still images showing the only the apex or peak of an expression (Figure 2.4). Faces in the real world, however, are rarely static. It is known that human visual system has learned to detect and decode expressions in dynamic situations. Temporal features, encoding the change over










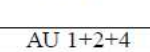
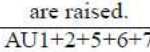
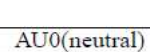
AU 1	AU 2	AU 4
		
Inner portion of the brows is raised.	Outer portion of the brows is raised.	Brows lowered and drawn together
AU 5	AU 6	AU 7
		
Upper eyelids are raised.	Cheeks are raised.	Lower eyelids are raised.
AU 1+4	AU 4+5	AU 1+2
		
Medial portion of the brows is raised and pulled together.	Brows lowered and drawn together and upper eyelids are raised.	Inner and outer portions of the brows are raised.
AU 1+2+4	AU1+2+5+6+7	AU0(neutral)
		
Brows are pulled together and upward.	Brow, eyelids, and cheek are raised.	Eyes, brow, and cheek are relaxed.

Figure 2.3. Some of the upper face action units and their combinations [3]



Figure 2.4. Representation of six basic emotions on static images: happy, angry, surprised, sad, disgusted, afraid

consecutive frames or the change with respect to a neutral frame is used to determine the underlying expression. As Pantic [94] stated, it is a growing research subject in cognitive sciences that the dynamics of facial expressions is crucial for the interpretation of human facial behavior [95, 96, 97, 98].

Dynamic situation is merely a collection of static snapshots between two emotions typically that start with a neutral face and end with a peak emotion as shown in Figure 2.5.

An alternative paradigm to FACS based FER systems is to model the appearance

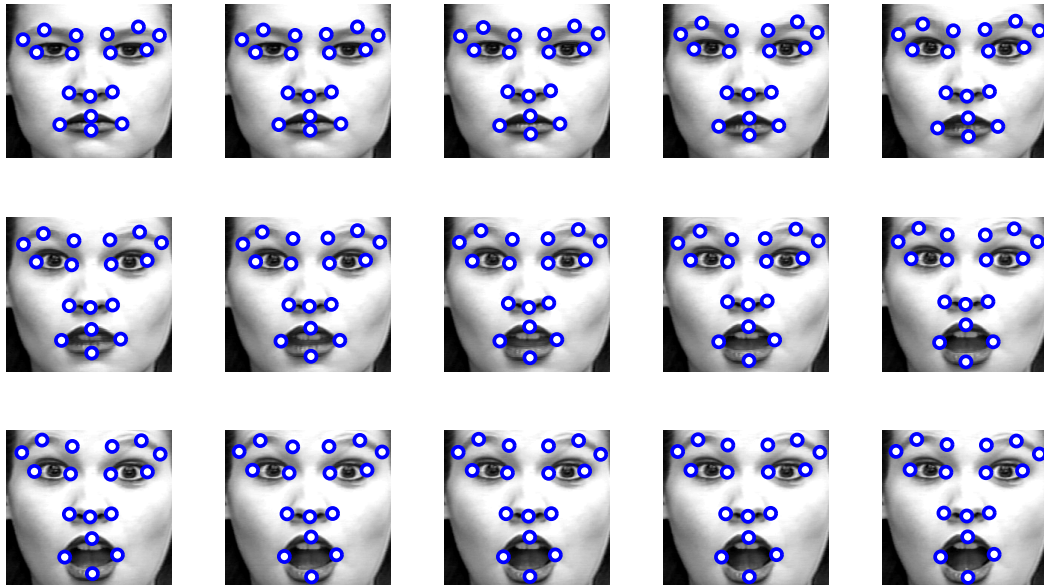


Figure 2.5. A dynamic stimulus

changes of the whole face or selected subregions by extracting discriminative features. For example, Tian [37] investigated the effects of different image resolutions for facial expression recognition by using geometric features and appearance-based features. Geometric features were extracted by tracking facial features which represented the shape and location of facial components e.g., mouth, eyes, brows, nose etc. Features were obtained by Gabor filtering applied on the difference images between neutral and expression faces. Similar to Tian's study [37], Bartlett et al. [35] also used Gabor filters for appearance based feature extraction from the still images. They obtained their best recognition results by selecting a subset of Gabor filters using AdaBoost and then training Support Vector Machines on the outputs of the filters selected by AdaBoost.

In addition to Gabor filters, which are robust to illumination changes and detect face edges on multiple scales and with different orientations, Local Binary Patterns (LBP) [40], Volumetric Local Binary Patterns (VLBP) [38] and Haar-like features [30] were also used for facial expression recognition and promising results were obtained.

In Zhao's study [38] the face was treated as a 3D volumetric data and motion and appearance were jointly modeled by VLBP. Finally SVM classifiers were trained using the extracted VLBP features. Yang et al. [30] used dynamic Haar-like features to cap-

ture the temporal characteristics of facial expressions. They further trained Adaboost on the encoded dynamic features. They manually aligned the faces as a preprocessing step. Shan [40] studied facial representation based on LBP features for facial expression recognition. They examined different machine learning methods, including template matching, SVM, Linear Discriminant Analysis (LDA), and the linear programming technique on LBP features. They obtained their best results with Boosted-LBP by learning the most discriminative LBP features with AdaBoost, and the recognition performance of different classifiers were improved by using the Boosted-LBP features.

However beyond the basic emotions, there are hundreds of complex mental states, e.g., confused, thinking and interested etc., that are visually identifiable. The automatic classification of complex mental states from the face videos is a significantly more challenging task as compared to the basic emotions. First, certain mental states hidden from the observer can only be inferred by analyzing the behavior of that person. Second, while basic emotions are mostly identifiable solely from facial action units, complex mental states additionally involve purposeful head gestures and eye-gaze direction. Finally, whereas basic emotions are identifiable from a small number of frames or even still images, complex mental states can only be recognized by analyzing the temporal dependencies across consecutive facial and head displays. Thus modeling complex mental states requires multi-level temporal abstractions [27].

Head displays, sometimes called as emblems [27, 84] fulfill a semantic function and provide conversational feedback. Examples of emblems are head nodding (head up and down) and head shaking (head swinging left and right) with or without accompanying facial expressions. In social interactions head and facial displays may convey a message, provide conversational feedback, and form a communicative tool [17, 18]. For example, head nod is an affirmative cue, frequently used throughout the world to indicate understanding, approval and agreement [17, 18, 99, 100, 101]. On the other hand, head shake is almost a universal sign of disapproval, disbelief, and negation [17, 18, 99, 100, 101]. Prediction of frustration and human fatigue detection problems were analyzed by integrating information from various sensory information [23, 25, 26].

- Fatigue and drowsiness detection [23, 24, 25]: Vural and et al. proposed a system for automatic detection of driver drowsiness from video [24]. Their project revealed a potential association between head roll and driver drowsiness, and the coupling of head roll with steering motion during drowsiness. It is obvious that incorporating automatic driver fatigue and drowsiness detection systems into vehicles may prevent accidents.
- Prediction of frustration [26]: Kapoor and et al. proposed a system to predict frustration of the subjects involved in a problem solving activity. The environment includes sensors that can measure video from face, postural movement from the chair, skin conductance (wireless sensor on non-dominant hand), and pressure applied to the mouse. There were two buttons prominent at the top of the screen that users could click on: “Im frustrated” and “I need some help”. The data observed through the sensors during the course of interaction are classified into “pre-frustration” or “not pre-frustration” behavior based on probabilistic machine learning techniques.
- Affective multimodal interface for office scenarios Maat and Pantic [102] developed an intelligent system, called Gaze-X, to support affective multimodal human-computer interaction where the users actions and emotions are modeled and then used to adapt the HCI and support the user in his or her activity. To support concepts of concurrency, modularity/scalability, persistency, and mobility, Gaze-X has been built as an agent-based system where different agents are responsible for different parts of the processing. A usability study conducted in an office scenario with a number of users indicates that Gaze-X is perceived as effective, easy to use, useful, and affectively qualitative [102].

There are relatively few papers in the literature addressing the FHG detection issue. In Kang et al. [100], location of eyes is detected and tracked in video sequence, and the resulting trajectory is used to recognize head shake and head nod gestures using HMMs. Somewhat similarly, Kapoor and Picard [99] used an active camera with infrared LEDs to track pupils. The position of pupils are used as observations by a discrete HMM pattern analyzer to detect head nods/shakes. Morency et al. [101] investigated how dialog context from an embodied conversational agent can improve

visual recognition of user gestures such as head nod and head shakes. For recognizing these gestures, they tracked head position and rotation, then computed head velocity vector and used SVM classifiers.

### 3. FACIAL LANDMARK DETECTION

The main task of this chapter is the automatic facial landmark initialization. We use a two-tier architecture for facial landmark initialization [4, 50, 49], where we first localize seven fiducial landmarks, and then in the second tier ten ancillary landmarks are found as shown in Fig. 3.1. The face is located using a modified version of Viola-Jones face detector [7]. Afterwards, the first tier detects seven facial landmark points, i.e., four eye corners, two mouth corners and the nose tip, via Support Vector Machines (SVM) trained separately on DCT features of each landmark. The estimated positions of landmarks are improved via a probabilistic graph model, which on the one hand eliminates false alarms (outlier estimates) and on the other hand increases the precision of landmark locations [4]. Since the classification of facial expressions would need more than seven landmarks, the next tier estimates ten more landmarks. These are inner corner, outer corner and middle point of the eyebrows, outer nostril corners and middle points of the upper and lower lips. We denote them as ancillary simply because their localization is dependent upon the first seven fiducial ones. Instead of brute-force searching for all ten ancillary facial points on the entire image, we place a seventeen-point face mesh by anchoring the seven corresponding nodes of the mesh on the actual fiducials. The landmark detection algorithm consists of the following steps:

- (i) Detect the face using modified Viola-Jones algorithm [7]
- (ii) Locate the seven fiducial facial landmarks (eye corners, nose point and mouth corners) using templates extracted during a training session [4]
- (iii) Initialize ancillary landmark positions by imposing seventeen node face mesh anchored on fiducial facial landmarks.
- (iv) Refine the positions of ancillary landmarks.

The details of these steps will be explained in the following sections.

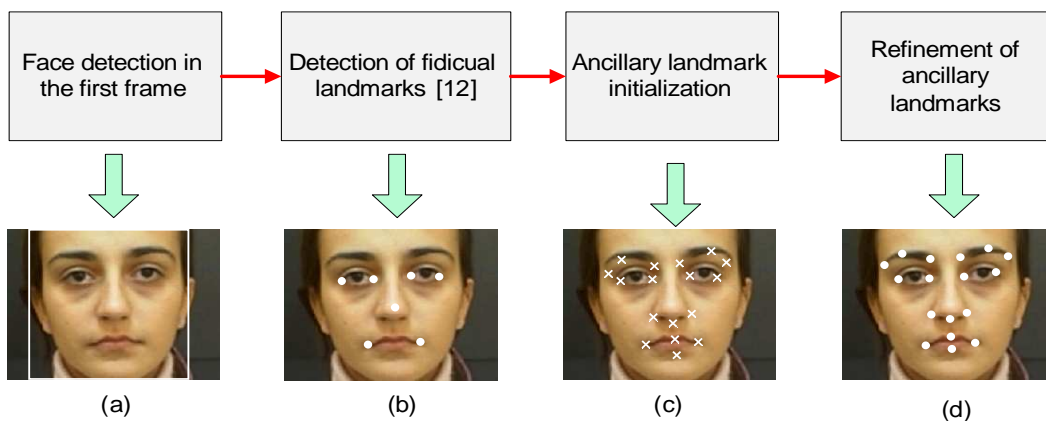


Figure 3.1. Illustration steps of facial landmark initialization. (a) Face detection, (b) Fiducial landmark detection: DCT-features, SVM and Probabilistic Graph Matching refinement as in [4], (c) Ancillary landmark initialization: Adaptation of the 17-point mesh, (d) Refinement of ancillary landmarks via template search and SVM

### 3.1. Coarse and Fine Level Fiducial Landmark Detection

In this part of the thesis we investigate local facial feature detectors both for 2D face images and 3D range information. It is meant gray-level intensity image by the term 2D and scene depth information by the term 3D in this chapter. If both 2D intensity image and 3D range data is available then hybrid schemes, where gray-level appearance information plays a predominant role and it is assisted by the 3D information, can be used to achieve more efficient facial landmark localization results.

Mainly two different techniques are proposed to compose discriminative feature vectors of the local facial patches: The first one is the Independent Gabor Features (IGF), where Gabor features are transformed via PCA and ICA in succession, and then classified with binary SVM classifiers (Section 3.1.1). In the second method, we use DCT coefficients as an alternative, again coupled with SVM classifier for feature localization (Section 3.1.2). Similar to many other facial feature localization algorithms, our methods employ a two-stage coarse-to-fine landmarking approach: In order to build a computationally efficient system, we start by searching potential landmark zones on downsampled 2D images.

Note that no illumination compensation method is employed during landmark detection algorithm. It is known that, Gabor features are less sensitive to illumination changes [103] and the DCT coefficients without DC term (the first coefficient) can be assumed to be illumination invariant under homogeneously changing lighting conditions. DC term represents the average intensity of the image block. It should be also taken into account that local features are less sensitive to illumination changes than global ones. Therefore the local DCT features and the local Gabor features that we utilized in this thesis are less sensitive to illumination changes.

### 3.1.1. IGF Based Features

IGF has been previously used by Liu and Wechsler for face recognition [104]. These features form a derivative of Gabor feature vectors, computed in different scales and orientations. Gabor kernels have such advantages as being tunable for good spatial localization and high frequency selectivity [31, 36, 48, 79, 104]. Furthermore, Daugman [105] suggested that the receptive field responses of simple cells can be described by the family of 2D Gabor wavelets. Gabor filters model the responses of the receptive fields of the orientation-selective simple cells in the human visual cortex [105]. Gabor wavelets are self-similar since all wavelets can be generated from one mother wavelet by dilation and rotation. To extract useful features from an image, a set of Gabor wavelets  $\{\sigma, k_v, \varphi_w\}$  with different frequencies, orientations and scales should be arranged. In spatial domain, Gabor wavelet is a complex exponential modulated by a Gaussian function as defined by the formula given in Equation 3.1.

$$\Psi_j(\vec{x}) = \frac{\vec{k}_j \vec{k}_j^T}{\sigma^2} e^{-\frac{\vec{k}_j \vec{k}_j^T \vec{x} \vec{x}^T}{2\sigma^2}} [e^{i\vec{k}_j \vec{x}} - e^{\frac{-\sigma^2}{2}}] \quad (3.1)$$

where  $\vec{x}$  define the pixel position in the spatial domain,  $k_v$  the scale parameter,  $\varphi_w$  the orientation of the Gabor wavelet, and  $\sigma$  the standard deviation of the Gaussian function along the x and y-axes.

$$\vec{k}_j = (k_{jx}, k_{jy}) = (k_v \cos \varphi_w, k_v \sin \varphi_w) \quad (3.2)$$

$$k_v = 2^{-\frac{v+2}{2}} \pi, \quad (3.3)$$

$$\varphi_w = w \frac{\pi}{8} \quad (3.4)$$

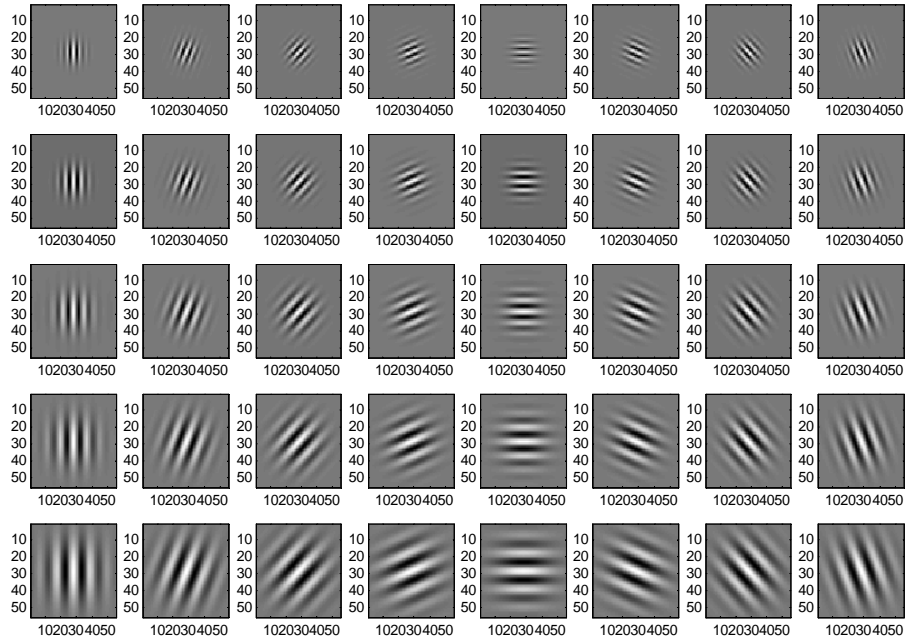
where  $v$  indicates scale and  $w$  indicates orientation.

The first factor in the Gabor kernel represents the Gaussian envelope and the second factor represents the complex sinusoidal function, known as the carrier. The term,  $e^{\left(\frac{-\sigma^2}{2}\right)}$ , of the complex sinusoidal compensates for the DC value.

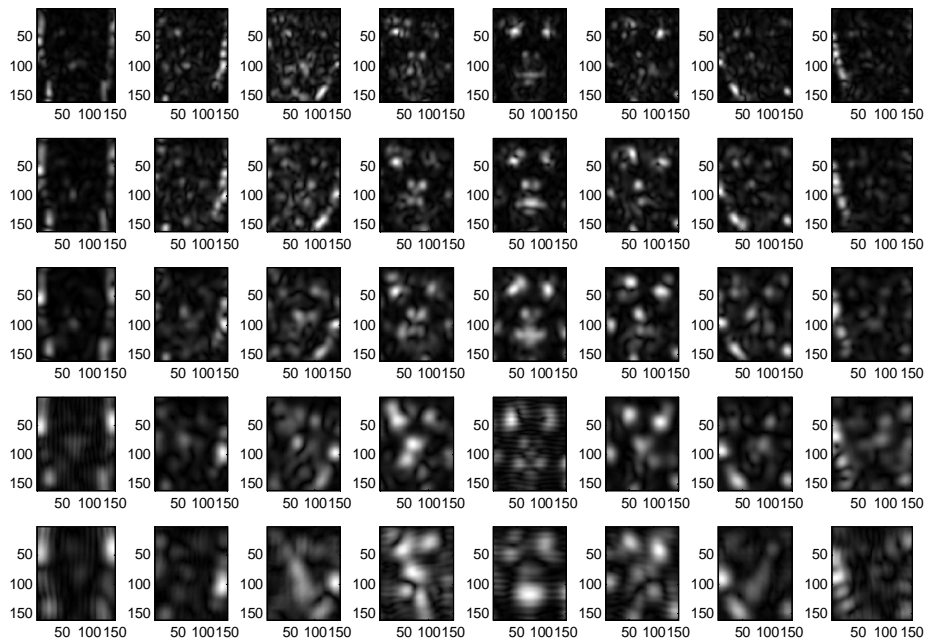
The Gabor representation of an image can be derived from the convolution of the image and the Gabor wavelets. Gabor wavelet representations at five scales ( $v \in \{0, \dots, 4\}$ ) and eight orientations ( $w \in \{1, \dots, 8\}$ ) are illustrated in Figure 3.2-a and the magnitude of the corresponding Gabor filtered image is illustrated in Figure 3.2-b.

The general block diagram of the IGF method is given in Figure 3.3. The first step is to extract Gabor feature vectors at defined scales and orientations from the given image. The dimension of the Gabor feature vector is very high as multiple scales and orientations are adopted. A useful method is the dimensionality reduction of these feature vectors via PCA. Therefore the second step is to employ PCA to the Gabor feature vectors. Then the third step continues with ICA process, which takes the higher-order statistics into account. These three steps constitute the IGF method.

In IGF approach, two Gabor kernel sets are designed. The first Gabor set is used for coarse localization on downsampled images, composed of three different scales, i.e.,  $v \in \{0, 1, 2\}$  and four orientations,  $w \in \{0, 2, 4, 6\}$ . The second Gabor set includes eight transforms, with two different scales, i.e.,  $v \in \{0, 1\}$  and four orientations,  $w \in \{0, 2, 4, 6\}$ . From the Gabor transformed face, we crop patches (i.e. feature generation windows) around each pixel of the search window. Each resulting Gabor feature is z-normalized by subtracting the component mean and dividing by its standard deviation. Finally, the component vectors over the grid are juxtaposed to form a larger feature



(a)



(b)

Figure 3.2. (a)Representation of Gabor wavelet filters at five scales and eight orientations , (b)The magnitudes of the Gabor representations of a face image

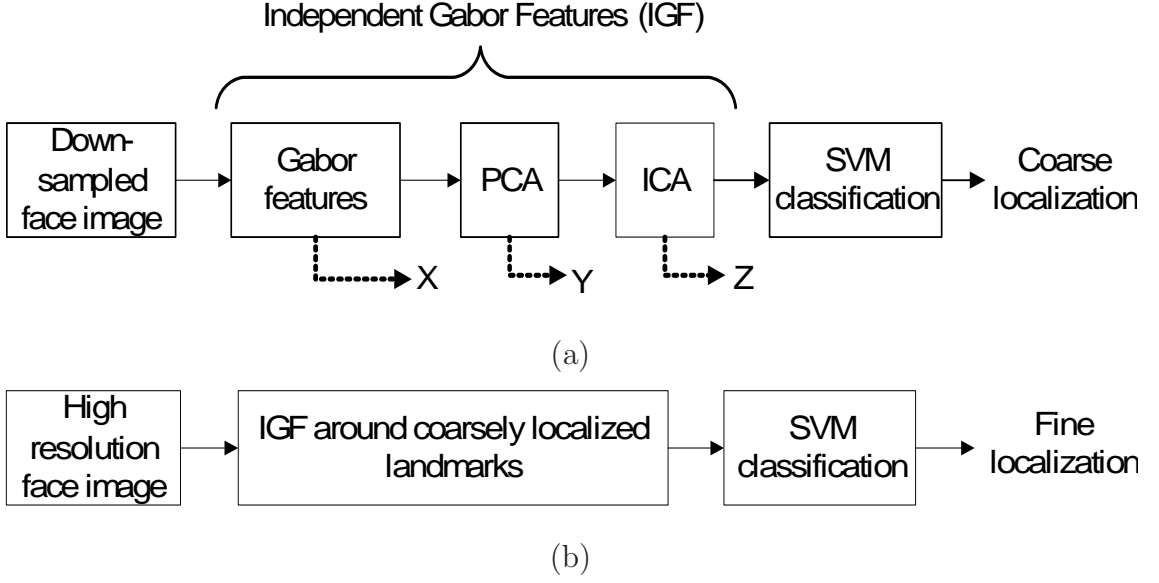


Figure 3.3. Overview of the IGF method both for coarse (a) and fine (b) localization vector  $\chi$ . The dimensionality is reduced to dimension  $n \ll N$  via PCA projection as

$$Y = E^t \chi \quad (3.5)$$

where  $E = [E_1 E_2 \dots E_n]$  is the  $N$ -by- $n$  eigenvector matrix corresponding to the  $n$  largest eigenvectors of the covariance matrix of  $\chi$ . The ICA method, which expands PCA by considering higher order statistics, was previously employed to derive independent Gabor features that were successful in human face recognition [104]. Other applications of ICA for face recognition can be found [106, 107]. The independent Gabor feature vector  $Z$  is obtained by multiplying the PCA-transformed features with  $W$ , the demixing matrix obtained by the FastICA algorithm [108]:

$$Z = WY \quad (3.6)$$

The de-mixing matrix  $W$  is obtained by maximizing some contrast function of the source data, which are assumed to be statistically independent and non-Gaussian. The derived ICA transformation matrix  $W$  is a combination of whitening, rotation, and normalization transformations [109].

In the coarse level feature localization,  $7 \times 7$  patches are cropped around each

search point and 100-dimensional IGF vectors are obtained by applying the PCA projection on  $7 \times 7 \times 12$  dimensional Gabor feature vectors. Note that, in the coarse level, search points correspond to each pixel of Gabor filtered low resolution face image. In the fine level,  $11 \times 11$  patches are cropped around the candidate points (coarsely localized landmark locations from coarse localization part) and 150-dimensional IGF vectors are obtained via PCA dimension reduction on  $11 \times 11 \times 8$  dimensional Gabor feature vectors. Computed feature vectors are fed to the SVM classifiers [110, 111].

### 3.1.2. DCT Based Features

DCT coefficients can capture the statistical shape variations and can be a faster alternative for facial landmark detection, compared to local IGF analysis. Therefore, in this approach we used DCT coefficients as discriminative local features [112, 113]. DCT is a model-based approach which transforms a vector into a space where the basis vectors are real-valued cosine signals.

At the candidate facial feature points on facial image  $Im$ , the  $C(\eta, \kappa)$  matrix containing DCT coefficients is computed as follows:

$$C(\eta, \kappa) = \alpha(\eta, )\alpha(\kappa) \sum_{y=0}^{K-1} \sum_{x=0}^{K-1} Im(y, x)\beta(y, x, \eta, \kappa) \quad (3.7)$$

$$\text{for } \eta, = 0, 1, \dots, K - 1 \text{ and } \kappa = 0, 1, \dots, K - 1,$$

where,

$$\alpha(\eta) = \begin{cases} \sqrt{\frac{1}{K}} & \text{for } \eta = 0, \\ \sqrt{\frac{2}{K}} & \text{for } \eta = 1, 2, \dots, K - 1, \end{cases} \quad (3.8)$$

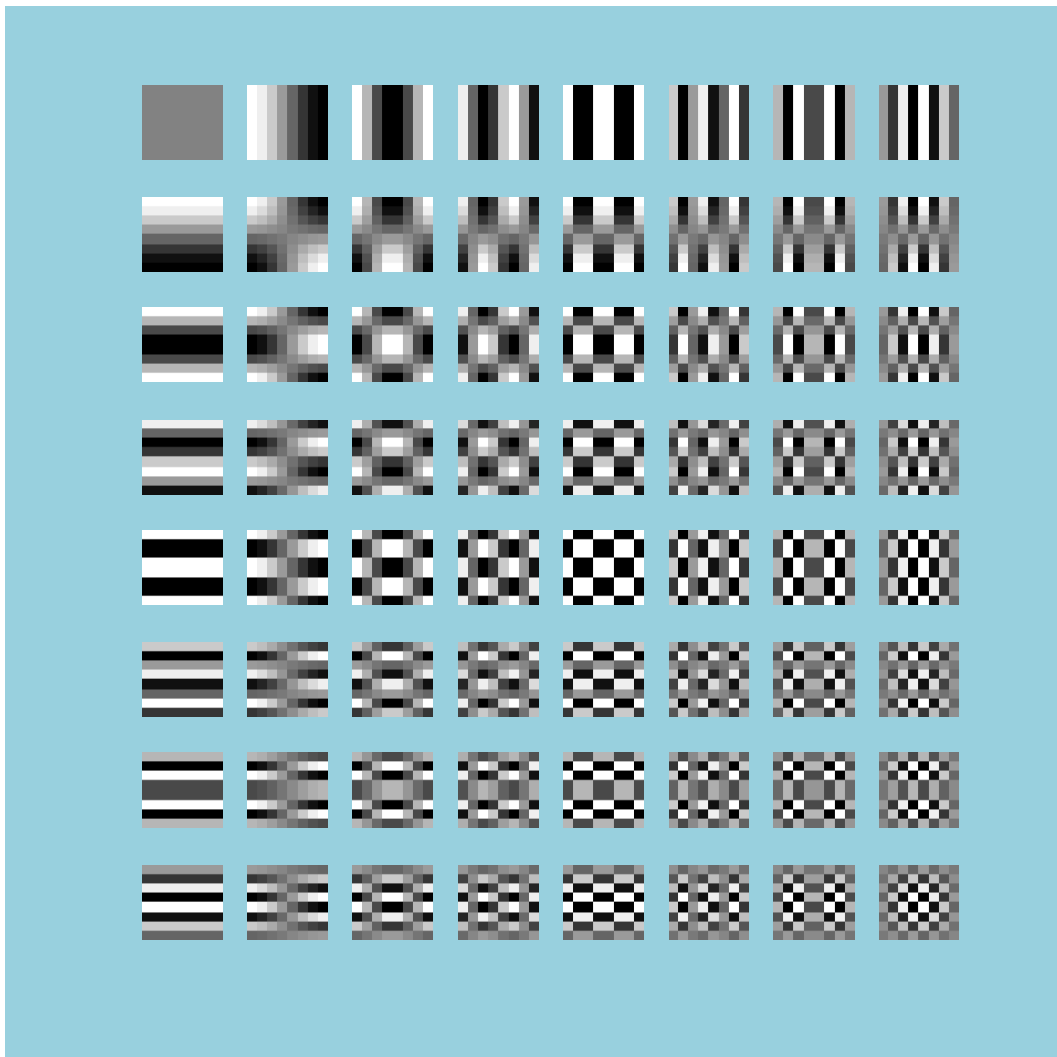


Figure 3.4. Two dimensional DCT basis functions ( $N = 8$ ), 2D basis are generated by multiplying the 1D basis functions for  $N=8$ .

and

$$\beta(y, x, \eta, \kappa) = \cos \left| \frac{(2y+1)\eta\pi}{2K} \right| \cos \left| \frac{(2x+1)\kappa\pi}{2K} \right|. \quad (3.9)$$

Two dimensional DCT basis functions generated by multiplying the one dimensional DCT basis functions for  $N = 8$  are illustrated in Figure 3.4. Some useful properties of DCT can be summarized as follows:

- (i) Easy to compute: When compared with Discrete Fourier Transform, DCT can be performed with real numbers.

- (ii) Data independency: It is not necessary to prepare a representative set of training data to compute a subspace. Since DCT has data independent basis functions.
- (iii) Energy compactness: If the signal is highly correlated and smoothly varying in the spatial domain, the DCT summarizes most of the information in few low frequency coefficients.
- (iv) Frequency information: It provides frequency information by converting signals from time-domain to frequency-domain to decorrelate the data.

Once the DCT coefficients are computed for a given image patch then the coefficients are ordered according to a zigzag pattern, in agreement with the amount of information stored in them. The first coefficient (DC value) is removed, since it only represents the average intensity value of the given image patch. The remaining (AC) coefficients denote the intensity changes or gray-level shape variations over the image patch. To analyze the effect of DCT coefficients both in coarse and fine stages, different number of DCT coefficients are used to form the DCT feature vector. In the coarse localization part we compute 8x8 DCT blocks from the whole face image except the background. For the fine localization, 16x16 DCT blocks are extracted for each point in a window centered at the coarsely estimated location. We use a search window of size 19x19, and obtain 361 candidate points for each facial landmark. This search window enables to improve the accuracy of the coarse landmark locations up to  $\pm 9$  pixels, which approximately corresponds to 9 % of average IOD distance for FRGC-I database. Figure 3.5 illustrates the histogram of inter ocular distances for FRGC-I database.

### 3.1.3. Support Vector Machine Based Classification

For the IGF and DCT based methods, we use SVM classifiers. SVMs belong to the class of maximum margin classifiers, such that they find a decision hyperplane for a two-class classification problem by maximizing the margin, which is the distance between the hyperplane and the closest data points of each class in the training set that are called support vectors. This linear classifier is termed the optimal separating hyperplane (OSH). Assuming linearly separable data, a separating hyperplane  $\xi x + b = 0$  exists.

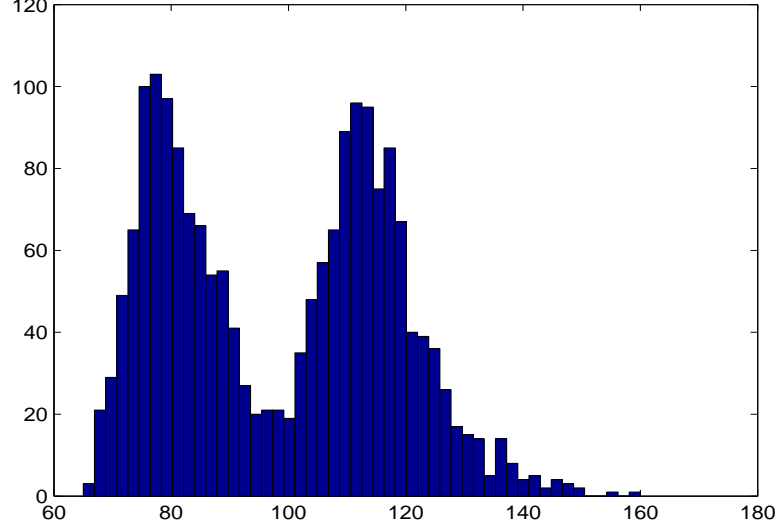


Figure 3.5. IOD histogram of FRGC-I database

The set of vectors is optimally separated without misclassification when the margin is maximal. The optimal plane must satisfy the condition  $y_i(\xi \cdot x + b) \geq 1$ . The distance between the hyperplane and the points is

$$d(\xi) = \frac{|\xi \cdot x + b|}{\|\xi\|} \geq \frac{1}{\|\xi\|} \quad (3.10)$$

therefore the optimal plane is obtained by minimizing  $\frac{1}{2}\xi^T \xi$  subject to  $y_i(\xi \cdot x + b) \geq 1$ . The optimization problem can be solved by the Lagrange function,

$$L(\xi, b, \alpha) = \frac{1}{2}\xi^T \xi - \sum_{i=1}^N \alpha_i [y_i(\xi \cdot x + b) - 1] \quad (3.11)$$

where  $\alpha_i$  are Lagrange multipliers. After solving the optimization problem, the OSH has the form:

$$f(x) = \sum_{i=1}^N \alpha_i y_i x_i^T + b \quad (3.12)$$

In the case of data, which are not linearly separable, we can project the data into a higherdimensional space in the hope of finding a linear OSH there. This is done by

replacing the inner product  $x_i^T x_k$  with a kernel function  $K(x_i, x_j)$  that satisfies Mercer conditions [110], thus allowing fast computations in the low-dimensional space, rather than the new, high dimensional space:

$$f(x) = \sum_{i=1}^N \alpha_i K(x, x_i) + b \quad (3.13)$$

One of the most frequently used kernel functions is the polynomial kernel  $K(x, y) = (1 + x \cdot y)^d$ , where  $d$  is the degree of the polynomial. In this study  $d$  is chosen as five. SVM classifiers are trained for both coarse and fine localization stages. Note that libsvm code [111] for Matlab was used in the experiments.

#### 3.1.4. Databases and Testing Methodology for IGF-and DCT-Based Features

In this section we have employed the University of Notre Dame (UND) database [114] for performance evaluation of the proposed facial landmark detection algorithms. The first part of the UND database is called FRGC-I and it consists of 942 2D images and the corresponding registered 3D point cloud data, both at resolution 480x640. Figure 3.6 shows sample images from FRGC-I database. The ground truth is created by manually landmarking seven points, that is, four eye corners, nose tip and mouth corners. The data are randomly split into three disjoint parts, i.e. the training (707 samples), and test sets (235 samples). To evaluate the performance of the feature localizers we have used a normalized distance, by dividing localization error, measured as Euclidean distance in terms of pixels, to the inter-ocular distance (distance between left and right eye pupils - IOD). A landmark is considered correctly detected if its deviation from the true landmark position is less than a given threshold, called the acceptance threshold (Figure 3.7).

All proposed algorithms proceed by facial feature localization on down-sampled (60x80) face images, followed by a refinement on the corresponding high-resolution (480x640) images. In the refinement stage, the search proceeds with a 19x19 window



Figure 3.6. Sample images from FRGC-I database

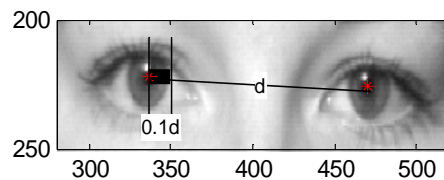


Figure 3.7.  $0.1d$  : accepted error distance for localization,  $d$  : IOD

from around the coarse localization results.

### 3.1.5. Performance of IGF based and DCT based Localizers on FRGC-I

The performance of the IGF-based and DCT-based facial landmark localizers are illustrated in Figure 3.8 and Figure 3.9 for the seven landmarks. Notice that the performance is given in terms of normalized distance. The horizontal axis indicates the distance threshold beyond which the location is assumed to be incorrect, whereas the vertical axis shows the correct localization percentage. The performance of the DCT-based algorithm is given in Figure 3.8, where the effect of the chosen number of DCT coefficients is illustrated. In the coarse level,  $8 \times 8$  DCT blocks are calculated at each facial point. Three tests were performed with 20, 35 and 42 coefficients, respectively, coefficients are always selected from the upper triangle of the DCT matrix.

In the fine stage,  $16 \times 16$  DCT blocks are calculated around each candidate facial landmark, from which choices of 54, 135 and 177 coefficients were tested. As we increase

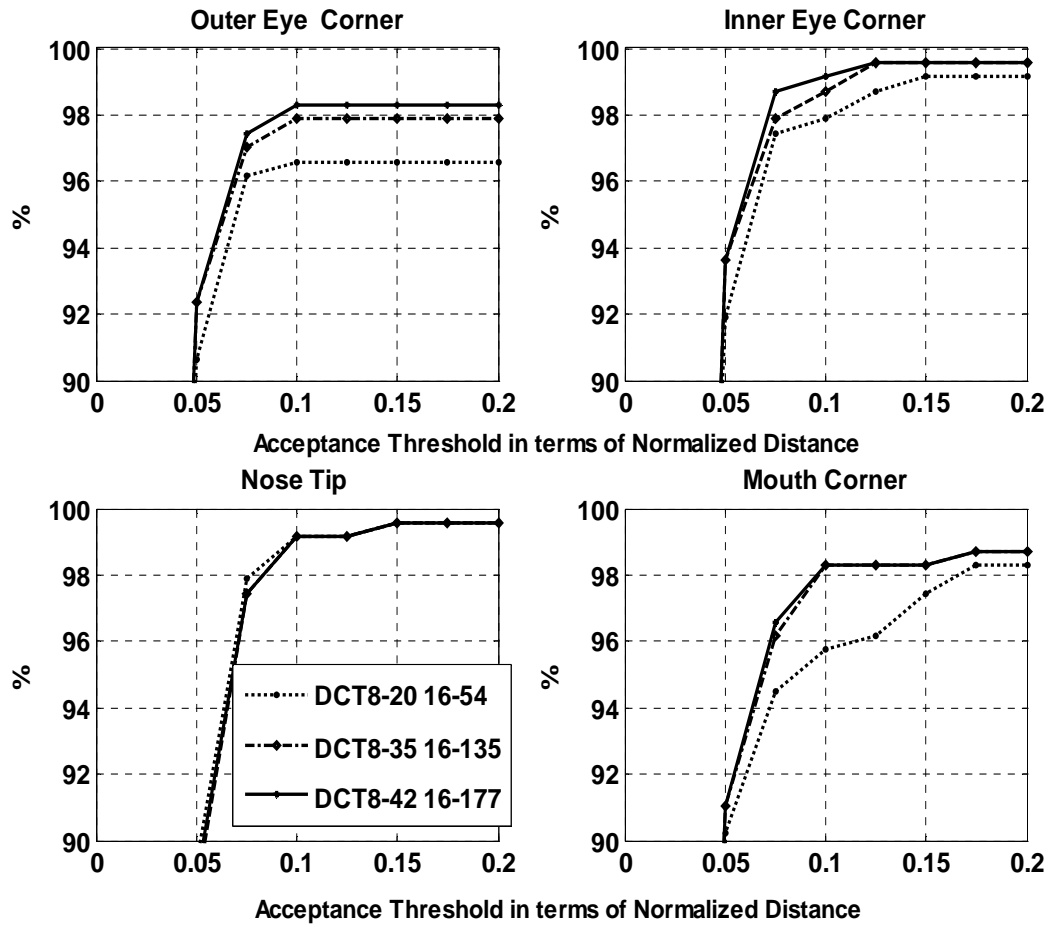


Figure 3.8. Tuning of the DCT method: results of the fine localization for each landmark type. Legend: DCT8-20 16-54 should be read as: 8x8 coarse level DCT transform and 20 out of 64 coefficients are selected; 16x16 fine level DCT transform and 54 out of 256 coefficients are selected

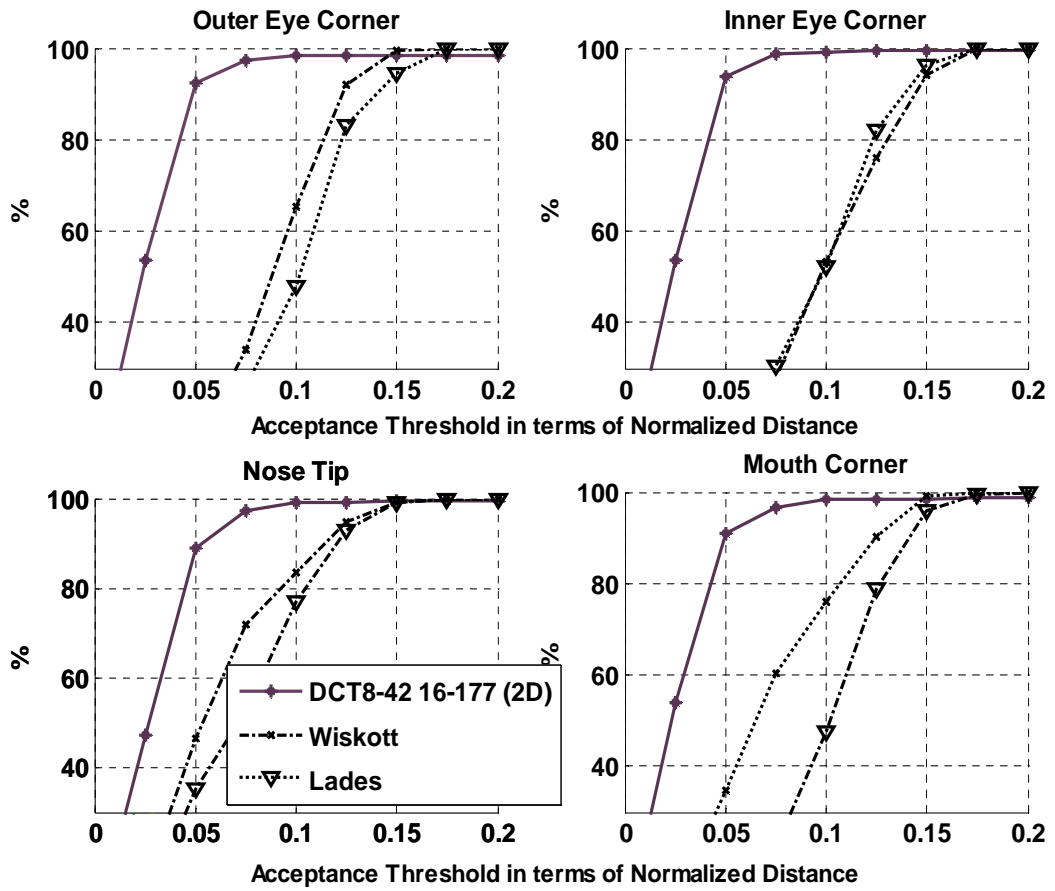


Figure 3.9. Comparison of fine refinement performances of DCT, Lades [5] and Wiskott [6]

Table 3.1. Average coarse and fine localization errors in terms of Euclidean pixel distances on the original images

Method	DCT-8-42	IGF 7-100	DCT-8-42
	DCT-16-177	IGF 11-150	DCT-16-177
	SVM (2D)	SVM (2D)	SVM (3D)
Coarse Localization (60x80 face images)	5.04 pixels	6.08 pixels	10.48 pixels
Fine Localization (480x640 face images)	2.80 pixels	4.36 pixels	7.81 pixels

the number of DCT coefficients, the localization accuracy increases as well, albeit with diminishing returns. This improvement is valid for all landmarks (See Figure 3.8) except for the nose tip. Increasing the number of DCT coefficients does not improve the performance of the DCT method for the nose tip. In Table 3.1 we show the average coarse and fine landmark localization errors for IMoFA-L (2D+3D), DCT (2D and 3D) and IGF (2D) methods.

All coarse methods except DCT on depth images give comparable localization results. Although the depth images result in a poorer average localization, the nose tip and inner eye corners are better localized, as the discriminative ravine and peak areas on the depth images correspond to the nose tip and inner eye corners. In fine localization part we compare our DCT algorithm with Ladess [5] and Wiskotts [6] Bunch-based methods. Figure 3.9 illustrates the comparison results. Lades and Wiskott methods are applied to search window which is located around the true landmark positions, our DCT algorithm, which is applied to search window located around the coarse landmark positions outperforms the Lades [5]and Wiskott [6] local feature analysis schemes.

In Figure 3.10, we can see the fine localization accuracy of the proposed algorithms for facial feature localization. The inner eye corners and mouth corners are relatively tougher to localize. There is a significant difference in the localization performance of different algorithms for different landmark types. However, the DCT method, appropriately tuned, seems to be the winner.

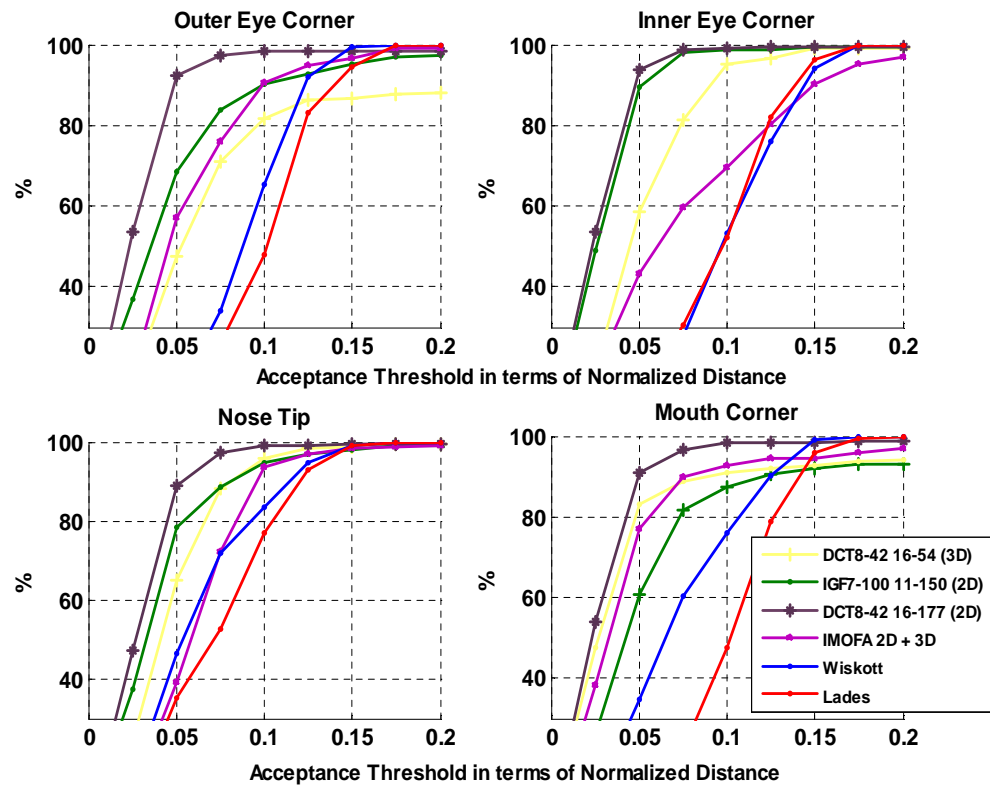


Figure 3.10. Comparison of fine refinement performances of DCT, IGF and IMoFA-L based methods on 2D and 3D data. Legend: IGF7-100 11-150 should be read as: 7x7 coarse level IGF vector and dimension reduced from 7x7x12 to 100 via PCA; 11x11 fine level IGF vector and dimension reduced from 11x11x8 to 150 via PCA. IMoFA 2D+3D is a combination of the conspicuity maps obtained separately from 2D Gabor features and 3D depth image.

### 3.2. Regularization with Probabilistic Graph Models

Challenging conditions, especially illumination and pose variations decrease the accuracy and robustness of facial feature landmarking. We can deal with these challenges on the one hand by resorting to graph-based methods that incorporate some anthropometrical information and on the other hand by using jointly 2D and 3D face data. In this section, we evaluate the contributions of graph-based methods and of joint usage of 2D and 3D information to the accuracy of facial feature localization.

This section will describe how to improve a baseline landmarker with the aid of probabilistic graphs. A detection scheme is robust if its performance difference is negligible between datasets that differ substantially in imaging conditions, such as illumination, pose, expression and accessories. For example, facial feature extraction in 2D images suffers heavily if illumination conditions change; in 3D if facial expressions are pronounced. More specifically, the problem is that if the facial feature detectors are trained on one database, but tested on a quite different database, the detection performance is observed to decrease dramatically. The main focus of this section is to explore the following paths:

- Improve the performance of 2D landmarking methods with the aid of graphs and anthropometrical knowledge.
- Improve similarly the performance of 3D landmarking reliable landmarks to compensate for the less reliable remaining subset.
- Do landmarking cooperatively with 2D and 3D data.

This way we expect to eliminate false positives and missing features.

#### 3.2.1. Face Localization on 2D/3D Face Images

Given a 2D test image, we first carry out the face detection and localization task. This can be performed either by some boosted cascade classifier, a modified version of Viola-Jones algorithm [7]. An alternative scheme would have been face blob based

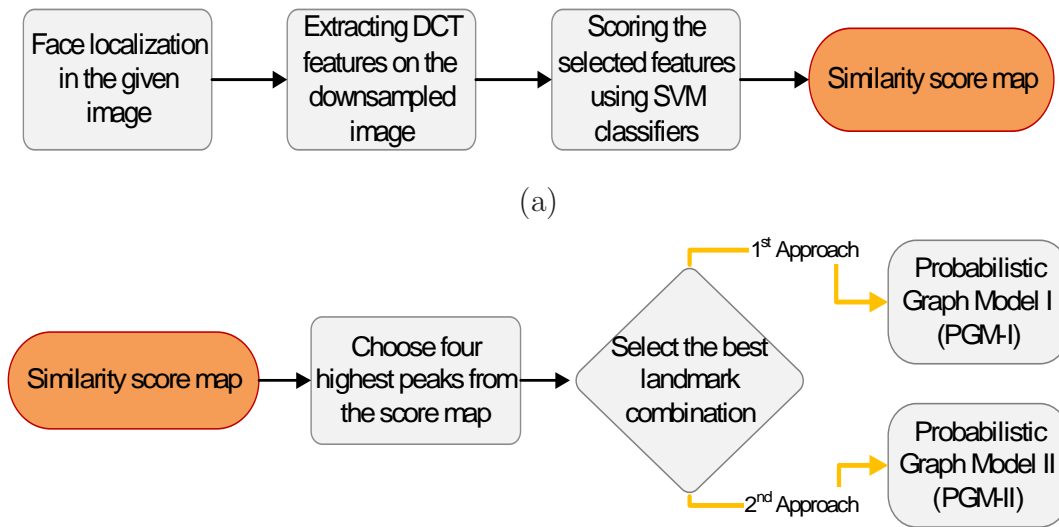


Figure 3.11. General flowchart of coarse level localization (a) Score map extraction, (b) Choosing Probabilistic Graph model (PGM)

detection on skin color followed by some post-processing [51]. In 3D, face localization is somewhat less problematic since the 3D face is expected to be located in the narrow slab of space that the range resolution of the sensor allows. The most protruding (nearest) point within this slab is assumed to be the nose tip for nearly frontal faces [75]. Finally 2D plus 3D data can be used for face detection. We test the output of the 2D face detector against the 3D detector. If there are multiple 2D face candidates, then we consider the intersection of 2D face masks with the 3D one, and pick up the one with the largest intersection. If there is no intersection, we consider that the 2D face detector has failed.

### 3.2.2. Improvement with Probabilistic Graph Models

The coarse level facial landmark points are identified with a two-tiered search scheme as shown in Figure 3.11. Briefly, the steps of the algorithm are as follows:

- (i) After the face is localized via a face detector, the face image is resampled till the IOD becomes 15 pixels. For FRGC-I database, the images are downsampled eight-fold, from its original resolution of 480x640 to 60x80. By this way, the average IOD for downsampled FRGC-I images becomes approximately 15 macropixels. We call the pixels of the lower resolution image as macropixels. This preprocessing

not only helps to reduce the computational complexity, but it enables the 8x8 DCT templates to do a more effective search.

- (ii) DCT templates, trained separately for each of the facial landmarks scan the face. We select the four highest peaks in the corresponding matching score maps as the possible candidates for the facial landmarks. Note that, the DCT templates are based on the 55% of the lower indexed DCT coefficients.
- (iii) Choosing  $k$  highest peaks of the score map guarantees that correct landmark points are not missed. Notice that we do not allow any two candidate points within a distance of  $t$  pixels ( $t = 1.5$  macropixels) in order to avoid replications.

Score maps for a sample test image is given in Figure 3.12. Score maps basically represent the score outputs of the SVM classifiers trained for different facial landmarks. Choosing  $k$  highest peaks means that choosing the  $k$  highest scored outputs of the classifier as the candidate landmark locations for the requested facial landmark. In our experiments  $k$  is chosen as 4.

After these initial features have been located two important tasks remain:

- To select the right combination of seven landmarks out of the extracted twenty-eight candidates. This combinatorial problem must be solved efficiently under real-time constraints.
- Since in general appearance-based facial feature detectors are not perfectly reliable, the detector may fail to respond at true feature location (missing feature) and respond at erroneous locations (false alarm). Therefore, the missing feature and the false-alarm feature problems must be solved. An example is given in Figure 3.13 where right mouth and right eye corners are missing and there are three false alarms for each landmark.

To this effect, we propose a structure-based framework. Thus we effectively use a two-tier model, where the first tier generates landmark candidates and the second tier verifies the most plausible subset. We develop two second tier algorithms: *Probabilistic Graph Model-I* (*PGM - I*), which is a heavy duty algorithm tuned to eliminate the

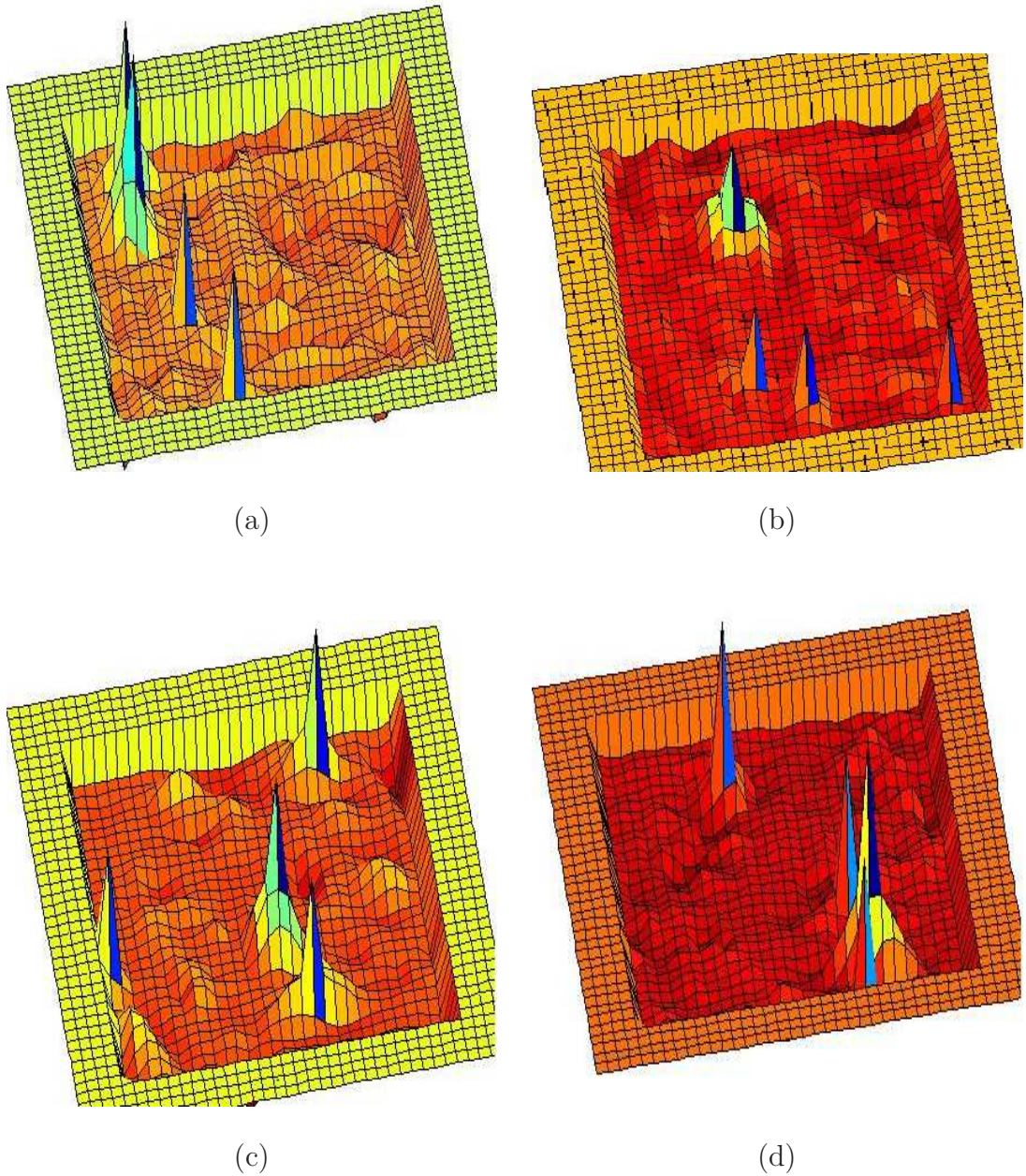


Figure 3.12. Score maps for (a) right eye outer corner, (b) right eye inner corner, (c) nose tip, (d) left mouth corner. These images can be interpreted as feature-ness map on the face. The peaks have been accentuated for visibility.

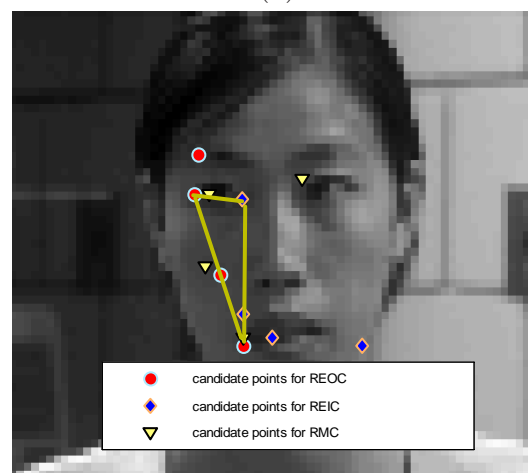
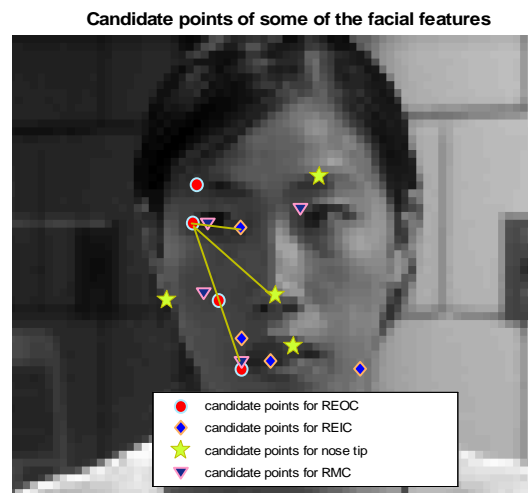


Figure 3.13. (a) Chosen candidate points (four highest scored locations) for some of the facial landmarks, (b) A plausible landmark triple of the right eye outer corner (REOC), right eye inner corner (REIC) and right mouth corner (RMC)

false alarms and to recuperate missing landmarks by estimating their location based on a subset of reliable landmarks. Missing landmark situation occurs when all chosen candidates are false alarms. In fact it is expected that one of the four chosen candidates will be the true location of that landmark. *Probabilistic Graph Model-II (PGM – 2)* is a light-duty algorithm tuned to improve the location estimates by fusing information of its companion landmarks. *PGM – I* is designed for more difficult face scenarios as compared to *PGM – II*.

### 3.2.3. Probabilistic Graph Model - I (*PGM – I*)

The *Probabilistic Graph Model-I* is especially suitable for challenging conditions where there are many false alarms. Our algorithm is a two-step algorithm; where the first step finds a subset of the most reliable landmarks, and then estimates the positions of the missing landmarks. It thus aims to recover the missing features and eliminate the false positives based on a minority of reliable features. The statistical model on which *PGM – I* is based consists of the normalized distances and angles of inter-feature points, learned during a training session.

Using these anthropomorphic data, we try to establish the best fitting combination of seven feature candidate points out of the twenty-eight ones declared by landmark detectors. This combinatorial search problem is initiated by selecting first the most reliable point from among the four candidates outputted from each detector. This gives us seven landmarks, one for each facial feature, though they may all still not be the correct ones. Then we consider all reliable combinations of landmarks and assign a figure of merit to each such reliable combination. In one instance, these reliable landmark points (call it the  $i^{th}$  configuration) might be the right eye outer corner, right eye inner corner and left mouth corner. Obviously, the threesome combinations of the seven landmarks result in  $\binom{7}{m}$  or 35 different sets when  $m=3$ . The figure of merit for the  $i^{th}$  configuration is calculated using the anthropometrical prior knowledge. This knowledge is given by mean length and mean relative angle  $(\bar{\lambda}_{i,j}, \bar{\phi}_{i,k})$  between the feature landmark points and their respective variances  $(\Lambda_{i,j}^k, \Phi_{i,k}^2)$ . In other words, the

angles and the lengths between feature points are regarded as Gaussian distributed random variables ( mean and covariance matrices  $(\bar{\phi}_i, \Phi)$  and  $(\bar{\lambda}_i, \Lambda_i)$  respectively). Given a combination of  $m$  reliable landmarks for  $m = 3, \dots, 7$ , the configuration energy is calculated as:

$$E(i) = \sum_{j=1}^m \frac{(\lambda_{i,j} - \lambda_{i,j}^2)^2}{\Lambda_{i,j}^2} + \sum_{k=1}^m \frac{(\phi_{i,k} - \phi_{i,k}^2)^2}{\Phi_{i,k}^2}, \quad \text{for } i = 1, \dots, \binom{7}{m}. \quad (3.14)$$

The stored energy in the graph is due to the spring forces between the actual location of the landmarks and their model location. Finally, we have to determine the rest  $7 - m$  of the seven landmarks, that is, to continue the prior example. For example when the right eye outer corner, right eye inner corner and left mouth corner are determined as the reliable landmarks then we have to locate the left eye outer corner, left eye inner corner and right mouth corner and the nose tip. There are  $4^m$  possible subsets to be considered. Recall that the model location information is obtained separately for each of the initial  $\binom{7}{m}$  configurations. Thus we compute a total of  $\binom{7}{m} * 4^m$  energy configurations as in Equation 3.14.

Given a set of most reliable triangles, the locations of the remaining landmarks can be estimated via a back-projection method as in Salah et al. study [74]. Our proposed algorithm is in a similar vein, but utilizes the  $\binom{7}{m}$  different distribution models for the  $m$  reliable and the remaining  $7 - m$  less reliable landmarks. As in [74], the triangle formed by the reliable landmarks is normalized; translated to origin, scaled to a fixed length and rotated so that each configuration becomes independent of the pose and scale of the face. Thus each configuration triangle will have its own set of normalization parameters. Obviously normalization with the same parameters is applied to the landmark points to be estimated. Given the most reliable points, the possible positions of the remaining landmarks are highly constrained as shown in Figure 3.14. Therefore this distribution models will allow us to estimate the coarse positions of the remaining landmarks.

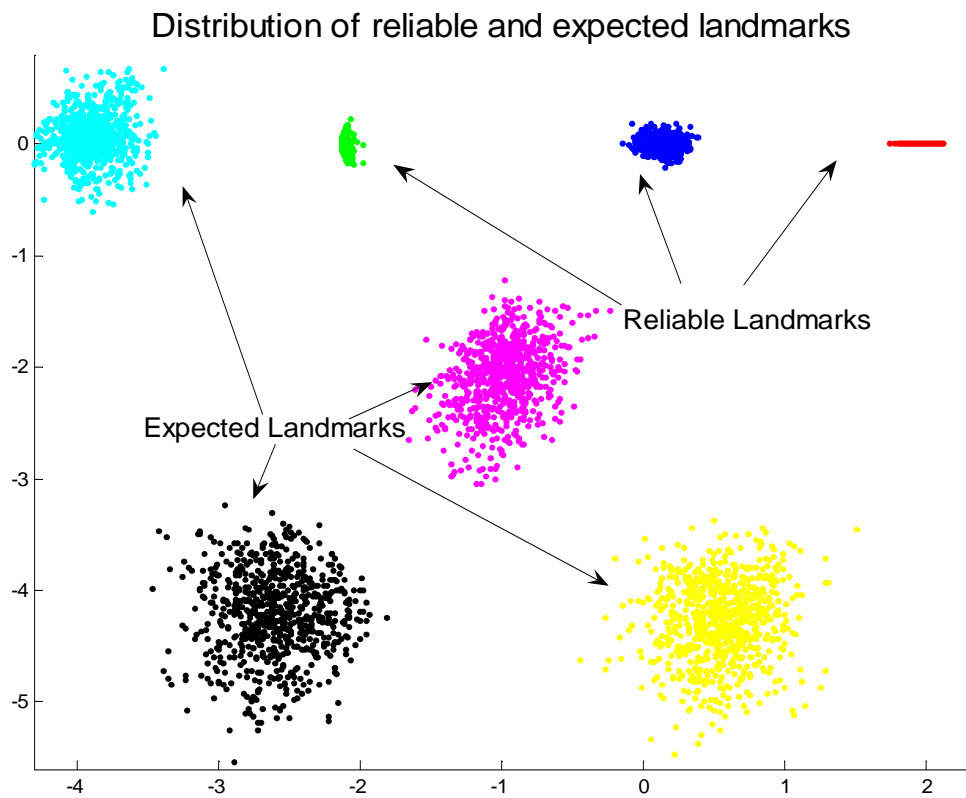


Figure 3.14. Scatter diagram of the three reliable landmarks (right eye inner corner, left eye inner corner and left eye outer corner) and of the four remaining landmarks vis-à-vis the basis three reliable landmarks

Figure 3.14 illustrates one such distribution model, where the dispersions of the three reliable points and those of the four remaining landmarks are shown. Each distribution is regarded as a Gaussian cloud, though this assumption is less valid for the reliable landmarks. In test part, once we establish the best fitting triple of feature landmarks, then the remaining four feature points are simply estimated using the means of the respective spatial distributions. These center points in the normalized coordinates are then subjected to the inverse normalization operation to establish their positions on the actual face. Note that when the number of reliable landmarks  $m = 3$  then the *PGM – I* scheme utilizes  $35 \times 43 = 2240$  possible searches to find the best combination.

### 3.2.4. Probabilistic Graph Model - II (*PGM – II*)

This model is designed for scenes where there is a small probability of missing feature and/or false alarms. Instead it represents a systematic search algorithm for the seven feature landmarks among the four peaks of the similarity score map. This graph approach is found appropriate to use with 3D range (depth) datasets and with 2D datasets for which the training and test conditions do not differ very much. A set of landmark points and their graph model are illustrated in Figure 3.15. In this algorithm, facial feature landmarks are not determined solely on their matching scores, but each is determined based on the harmony with companion features. For instance, the right eye outer corner needs to be in concordance with the positions of the right eye inner corner, nose tip and right mouth corner. These companion features constitute the support set for that feature. Obviously, the support set could have been constituted of all the remaining features. This not only will increase the computational load, but more importantly, it is more crucial to seek for coordination between features that are inherently tightly related to each other.

In *PGM – II*, we set out to test all possible quadrilaterals for each feature. Since we are dealing with quadrilaterals, we have  $4^4$  configurations for each feature, the four alternatives from the feature that we are trying to localize and  $4 \times 4 \times 4$  other candidates

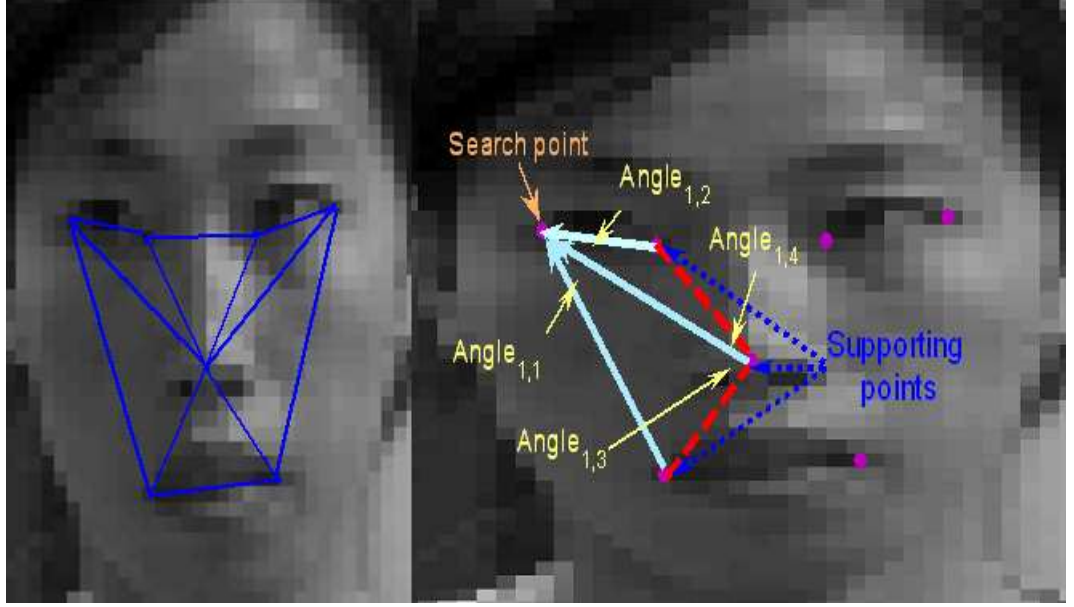


Figure 3.15. Face graph for PGM-II and sample quadrilateral for the right eye outer corner with its three support features.

from its support set. We calculate the stress energy for each such configuration:

$$E(i) = \sum_{j=1}^3 \frac{(\lambda_{i,j} - \lambda_{i,j}^2)^2}{\Lambda_{i,j}^2} + \sum_{k=1}^4 \frac{(\phi_{i,k} - \phi_{i,k}^2)^2}{\Phi_{i,k}^2}, \quad \text{for } i = 1, \dots, 4^4. \quad (3.15)$$

The first term in Equation 3.15 represents the sum of the deviations of the distances between support sets and the searched landmark and the second term represents the deviation of angles defined within the peripheral as shown in Figure 3.15. The feature landmark is declared to be the one with the lowest energy. This exercise is repeated for each landmark except the nose, which has a different neighborhood in the graph. Since every quadrilateral has the nose as one of the vertices, we simply chose it as the landmark point that is part of a quadrilateral with lowest energy. Thus the *PGM-II* scheme runs  $6x4^4 = 1526$  searches.

The coarse level localizer, described in Section 3.2.2 operates on the downsampled face images. *PGM-I* and *PGM-II* stages are followed by a refinement step executed on the original high-resolution image. In this refinement stage, the search proceeds with a 21x21 window from around the coarse localization points. This selected search window enables to improve the accuracy of the coarse locations up to  $\pm 10$  pixels which

Table 3.2. Performance of different window sizes for fine refinement

Window size	15x15	21x21	27x27
Average localization error in pixels	4.22	4.15	4.18

approximately corresponds to 0.1 of average inter ocular distance for FRGC-I database. The method is quite similar to the one used for coarse level, we use again DCT-based feature descriptors and an SVM classifier for fine level and choose the point with the highest score as the final location of the searched facial point [49]. Table 3.2 summarizes the performance of different window sizes for fine refinement in terms of average Euclidean distance to the groundtruth. It is observed that 21x21 window size is optimum when its compared with 15x15 and 27x27 window sizes.

### 3.2.5. Performance of *PGM – I* and *PGM – II* on FRGC-I and FRGC-II

We have used FRGC-I dataset for testing and FRGC -II dataset for training [114]. Each image in the datasets includes 2D intensity image and its corresponding 3D range data. FRGC-I images were captured under studio lights with uniform lighting, FRGC-II consists of images with facial expressions and non-uniform lighting. All the 2D and 3D faces are registered and possess point-to-point correspondences. Furthermore, all the faces in the datasets were marked manually with landmarks as groundtruth.

The performance of the facial landmark localizers with *PGM – I* and *PGM – II* is shown in Figures 3.16-3.19. In Figures 3.16,3.17 and 3.19 the horizontal axis denotes the error threshold as a percentage of inter-ocular eye distance, and the vertical axis denotes the percentage of test points that achieve to remain below this error threshold. Figure 3.19 illustrates the contribution of 2D+3D multimodality versus pure 2D or pure 3D. This graph can be interpreted such that more points closer to the x-axis represents better fine tuning with the corresponding multimodality. At the end of this section, the following observations are obtained:

- (i) Role of PGMs: When feature localizers are trained on one dataset and tested on another with substantially different imaging conditions, the performance is very

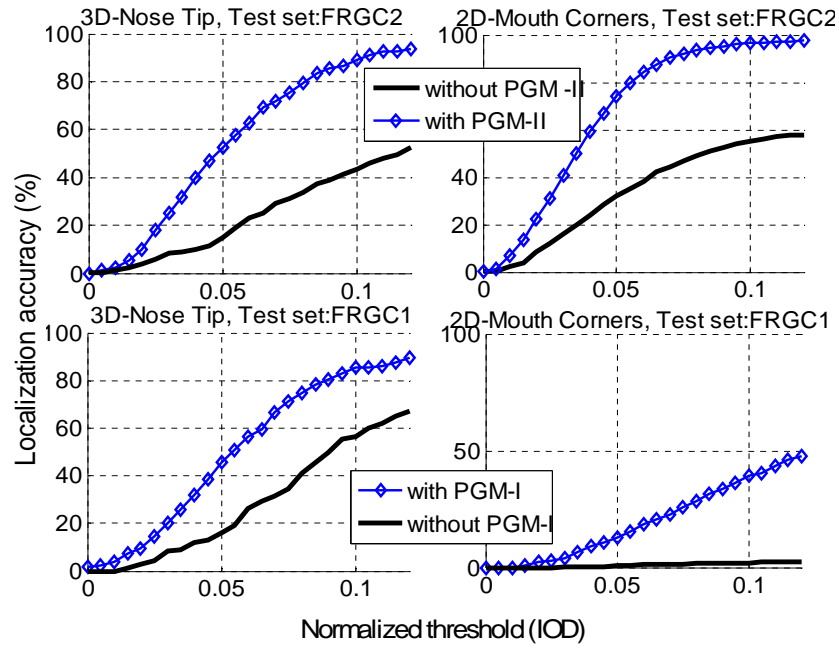


Figure 3.16. Graph-based correction. Improvements of 2D feature accuracy with PGM-I and of 3D feature accuracy with PGM-II.

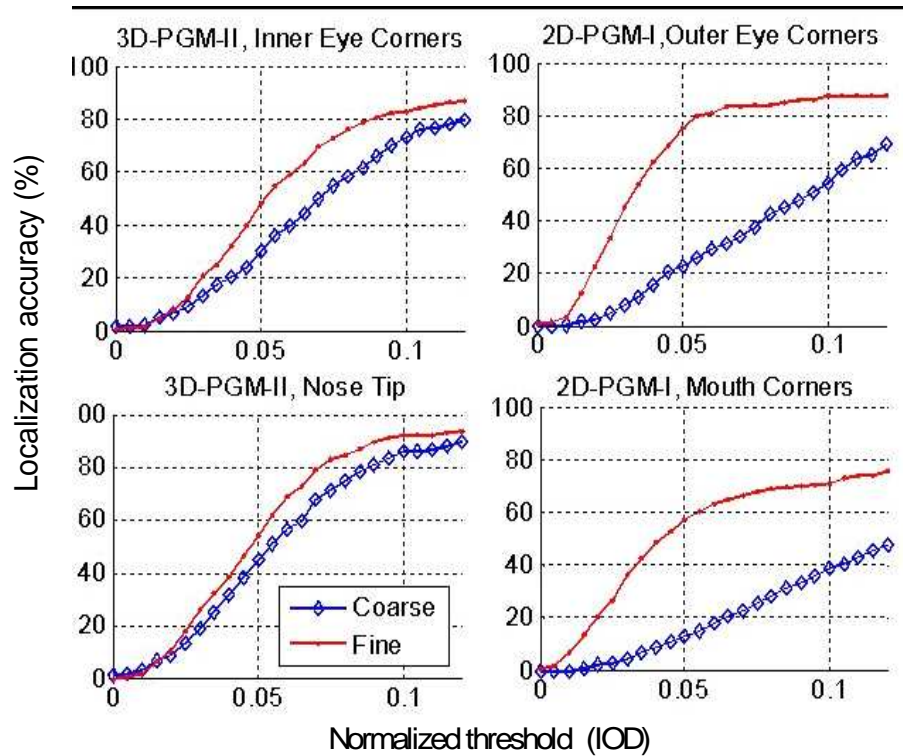


Figure 3.17. Improvement with the refining stage after applying the PGM methods

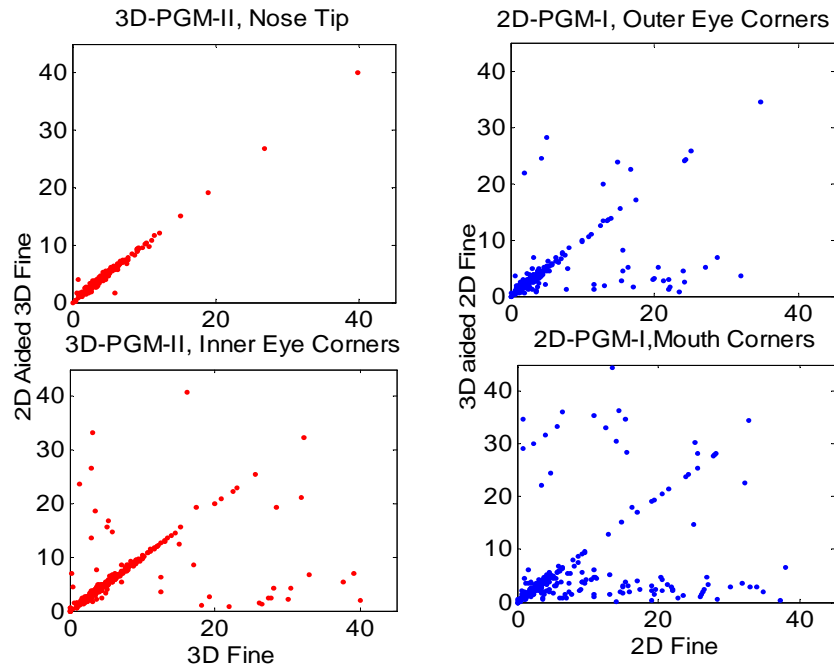


Figure 3.18. Contribution of 2D + 3D. First column: The location errors with pure 3D techniques are plotted on the vertical; those with the 2D-aided 3D localization error on the horizontal. Second column: Similar results for pure 2D vis-à-vis 3D-aided 2D localization.

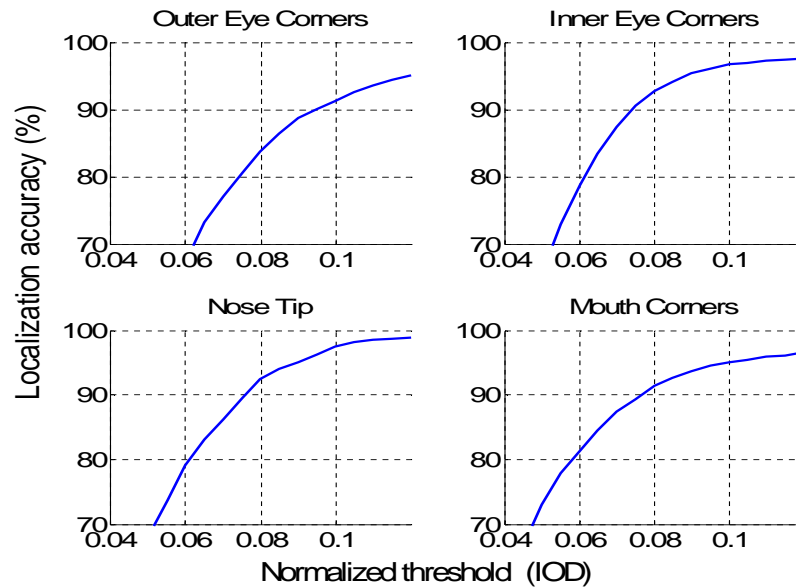


Figure 3.19. Performance of landmark localization on face images acquired in similar conditions: 2D-Fine test results with PGM-II (training set: FRGC-II (fall 2003 session), test set: FRGC-II (spring 2004 session))

poor, down to 20-30%, as shown in Figure 3.16 (line: without PGM; diamond polymark: with PGM), while if training and test datasets have similar illuminations the performance is very satisfactory close to 100% scores [49]. Figure 3.16 shows the improvement when PGM-I was used with 2D data and when PGM-II was used with 3D data.

- (ii) Role of refining: After the coarse localization, there exists still some room for improvement. In Figure 3.17, we show the accuracy improvement with the refining stage. Note that, if PGMs had not improved the initializations sufficiently, stage two refining gains would have been much less.
- (iii) Role of multimodality: The cooperation between 2D and 3D schemes is as follows: we separately estimate the landmark locations with 2D and 3D data, and then nonlinearly average as follows: If their Euclidean distance exceeds a threshold value (2 macropixels) then both points are sent to fine level tuning and select the fine level point with highest score. If, on the other hand, they do not differ much, then calculate the average of the coordinates of the 2D and 3D localizers, and proceed with this result to finetuning. We can see in Figure 3.18 that the 3D-aided 2D or 2D aided 3D methods are only marginally better than pure 2D coarse-2D fine or 3D coarse-3D fine methods. In this figure, the location errors with pure 2D (or 3D) techniques are plotted on the horizontal; those with the 3D-aided 2D localization errors (or 2D-aided 3D) on the vertical. The fact that there are more points below the diagonal signifies that the errors have diminished.
- (iv) Feature characteristics: There are non-negligible differences in the accuracy of detection of various features. For 2D images, the outer eye corners are most reliable, followed by mouth corners, nose tip and inner eye corners. The unexpected difference between inner and outer eye corners can only be explained by the sharp shadows affecting inner eye corners in one dataset, but not in the other dataset. In 3D, the nose tip fares the best, followed by the inner and outer eye corners, and lastly the mouth corners. It is again understandable that mouth corners are least reliable in 3D images due to expression variations.

### 3.3. Final Adjustments for Automatic Facial Landmark Localizer

#### 3.3.1. Improvement in Search Region

When we analyze the results obtained in Section 3.1 and Section 3.2, it is observed that coarse level localization (landmark localization in downsampled resolution) is very important for an accurate facial landmark localization in fine level. Since the localization performance of fine tuning at higher resolution images is highly dependent to the coarse localization performance. Coarse level facial landmark localization performed in Sections 3.1 and 3.2 is based on scanning the whole face region by using feature templates, or SVM classifiers, trained separately for each of the facial landmarks. Scanning the whole face region without using any priori anthropometric information is a redundant process. For example searching an eye corner template within the lower part of the face region may lead to false alarms. If the coordinates of the face region in test images are known then searching the facial landmarks only within the related subregions obtained by using the anthropometric knowledge will improve the localization accuracy in addition to a decrease in the time complexity. In Figure 3.20, the selected search windows based on the detected face region via boosted Haar feature based method [7] are illustrated. In order to extract these subregions some heuristic rules based on the anthropometric knowledge is utilized.

It is obvious that, using these selected windows for searching the related facial landmarks will improve both the accuracy and the speed of the algorithm. Therefore in the rest of this thesis, these selected subregions and *PGM – II* structure will be utilized within the automatic facial landmark detector in order to achieve real-time and more accurate results.

#### 3.3.2. Ancillary Landmark Initialization

In this subsection, the number of detected facial landmarks will be extended from seven to seventeen. Figure 3.21 shows the main steps of the 17 point facial landmark initialization algorithm. In Sections 3.1 and 3.2, facial landmark detectors

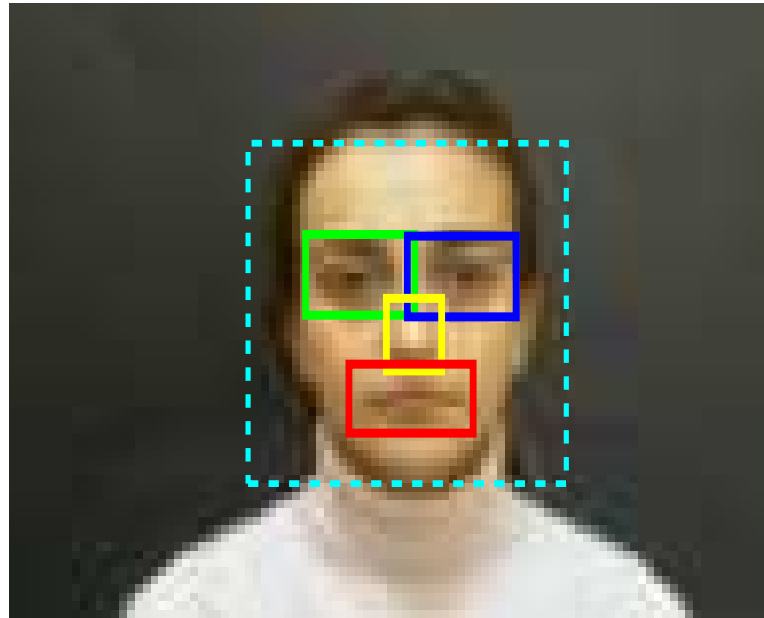


Figure 3.20. Detected face region via boosted Haar feature based face detector [7] and the corresponding subregions for coarse level landmark search.

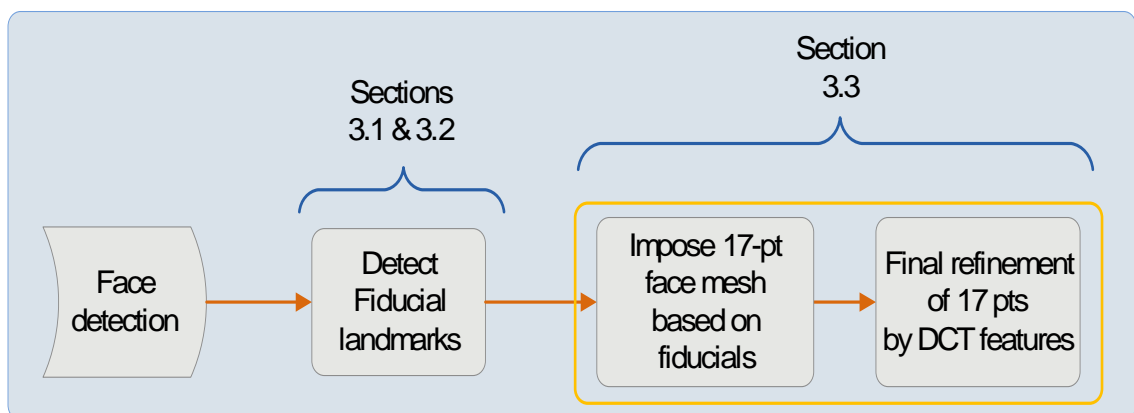


Figure 3.21. Main steps of automatic facial landmarking method

for seven fiducial facial landmarks (four eye corners, two mouth corners and the nose tip) are detected via SVM trained separately on DCT features of each landmark. The estimated positions of landmarks are improved via probabilistic graph models, which on one hand eliminates false alarms (outlier estimates) and on the other hand increases the precision of landmark locations (Section 3.2). The analysis of facial expressions and head gestures would need more than seven landmarks. Therefore, the next tier estimates ten more landmarks. These are inner corner, outer corner and middle point of the eyebrows, left and right nose corners and middle points of the upper and lower lips. We denote them as ancillary simply because their localization is dependent upon the first seven fiducial ones. Instead of brute-force searching for all ten ancillary facial points on the entire image, we place a 17-point face mesh by anchoring the seven corresponding nodes of the mesh on the actual fiducials which were already detected.

Training steps of the 17-point face mesh obtained by using manually landmarked face dataset are summarized as follows:

- (i) 7 fiducial facial feature points are translated to origin.
- (ii) Scaled such that the sum of the squared distances to origin becomes 1.
- (iii) Rotated until the first landmark aligns to the y-axis.
- (iv) The obtained rotation, scale and translation parameters are applied to the rest of the 10 landmarks, means of these landmarks are saved.

The distribution of the ten ancillary points anchored on the seven fiducial landmarks are illustrated in Figure 3.22. When the coordinates of the seven fiducial facial landmarks are detected then the following steps are processed to estimate the remaining ten ancillary landmark coordinates roughly:

- (i) 7 facial feature points are translated to origin.
- (ii) Scaled such that the sum of the squared distances to origin becomes 1.
- (iii) Rotated until the first landmark aligns to the y-axis.
- (iv) The obtained rotation, scale and translation parameters are applied inversely to the normalized means of the remaining 10 facial points.

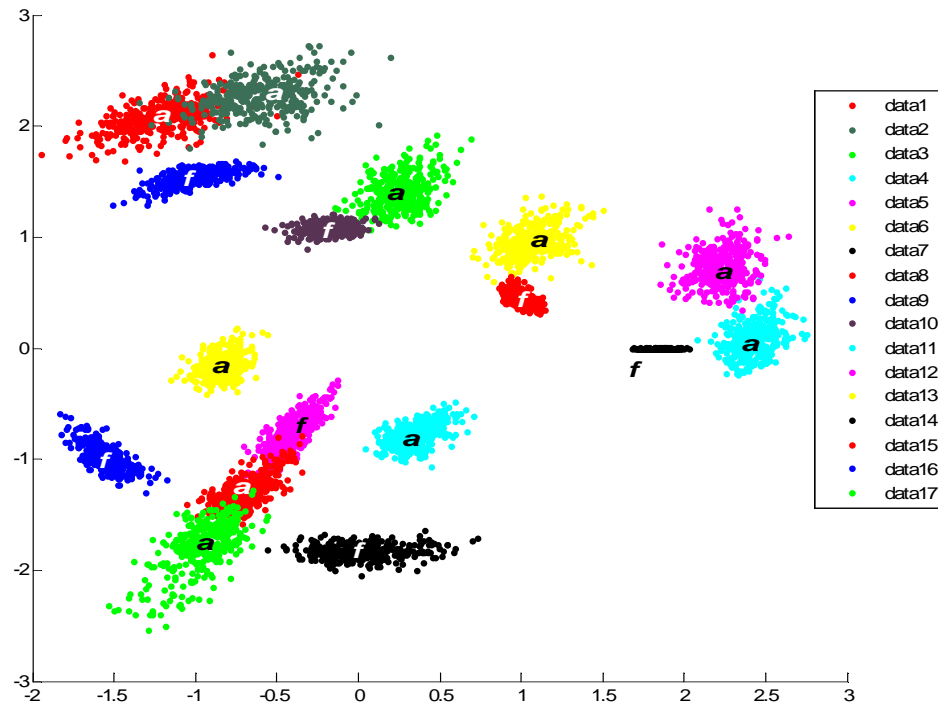


Figure 3.22. Distribution of ancillary landmark points (shown with  $\mathbf{a}$ ) anchored on fiducial landmark points (shown with  $\mathbf{f}$ )

The last step of the facial landmarking algorithm is the final refining. We search with a window in the neighborhood of the mesh-initialized ancillary landmark positions for the best match using SVM-classifiers. The SVM classifiers are trained with DCT features, one SVM for each landmark. For final refining, a subset of Bosphorus face database [8] is utilized for the training of the DCT-based SVM classifiers. Unlike UND face database [114], Bosphorus database includes a rich set of expressions and systematic variation of head poses with 3D depth information. Sample images under head pose and facial expressions are illustrated in Figure 3.23. Note that images with head poses up to  $45^\circ$  yaw angle are included to the training process. Including head poses and facial expression to the training stage makes the overall landmark detection system capable to detect landmarks of the faces under head poses and facial expressions.

With the facial landmarks initialized in the first frame of the sequence as in Figure 3.1(d) we can start their tracking in the subsequent frames of the face videos. In the next chapter of this dissertation, the proposed tracking algorithm based on the automatically initialized facial landmarks will be described in detail.

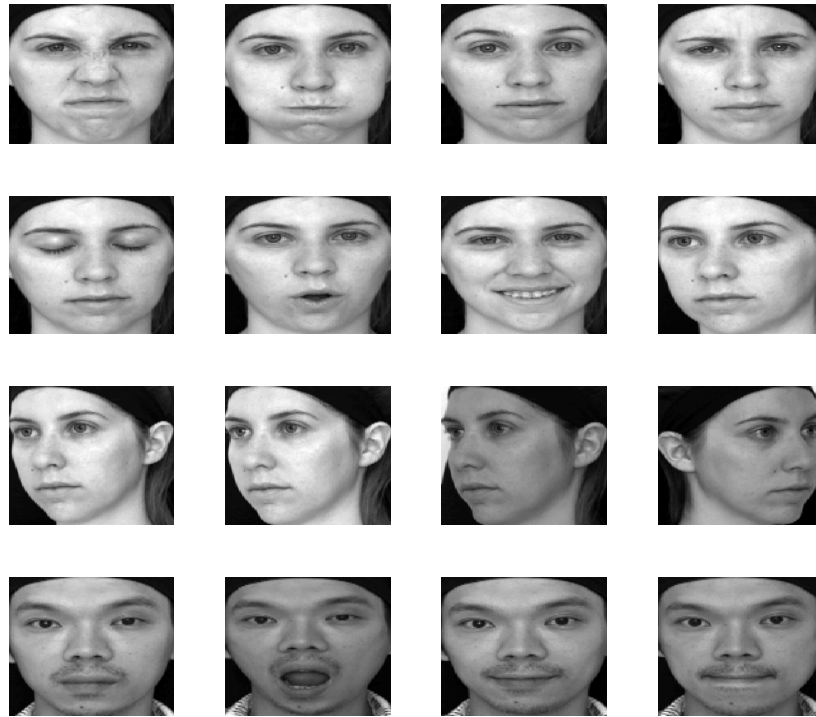


Figure 3.23. Sample 2D face images from Bosphorus Database[8]

With the proposed facial landmark detection algorithm we obtained the following computation times for the automatic detection of seven fiducial landmarks:

- PGM-I with 3 reliable subset requires 7.8 seconds
- PGM-I with 4 reliable subset requires 15 seconds
- PGM-I with 5 reliable subset requires 30 seconds
- PGM-II requires 7 seconds

Final refining of ancillary landmarks requires about 5.3 seconds of computation time. Note that these computation times are obtained by running the non-optimized MATLAB codes on Core 2 Duo at 3 GHz.

### 3.4. Conclusions

Automatic localization of facial landmarks is a critical process for many practical applications of computer vision technology, such as expression recognition, head

pose and gesture analysis, face recognition, applied to the human face. Despite many studies in the literature, facial feature localization is still an open problem because of its challenging nature. Wide variety of illumination conditions, pose, expression, gender, race and occlusions cause trade-off between accuracy and robustness of the facial landmarking systems. It is observed that the detectors are mostly dependent to the training data. Especially, appearance based approaches tend to fail if the imaging conditions of the test images are different than the training dataset, on the other hand, structure based approaches requires more training data to capture the variations of the facial landmark locations under head poses and facial expressions. To address this problem, we have proposed a two stage facial landmarking algorithm. The coarse level facial landmark detectors, which are assisted by probabilistic graph models are more robust to varying conditions. In the sequel, for better precision and accuracy, the output positions of the coarse stage are fed into the fine localization stage which is based on appearance based classifiers only. This stage, using a DCT-based scheme, finds more accurate feature positions.

The main problem addressed was the fact that facial feature localizers while performing very satisfactorily for the dataset for which they are trained, collapse when they operate on alternate datasets that differ substantially in illumination conditions, poses and accessories. To this effect we have proposed two remedial algorithms. *PGM – I* is a heavy-duty algorithm tuned to eliminate the false alarms and recuperates missing features by estimating their location based on a subset of reliable landmarks. *PGM – II* is an algorithm that estimates the location of a feature by fusing information of its companion features. The robustness and satisfactory performance of the coarse landmarking algorithm is improved by increasing the number of candidates from one to four (in order to prevent miss-detection) and then the best combination of the landmarks (eliminated from the outliers) are obtained with probabilistic graph models.

We evaluate the contributions to the accuracy of facial feature localization of graph-based methods and of joint usage of 2D and 3D information for landmark localization. It is observed that 3D-aided 2D or 2D-aided 3D methods are only marginally better than pure 2D coarse-2D fine or 3D coarse-3D fine methods. In the final analysis,

the proposed 7-point DCT-based method outperforms its competitor methods in the literature ([6] and [5]).

Search region of the coarse level localization is narrowed as a final adjustment step by using the prior anthropometric knowledge in order to achieve a more accurate facial landmark detectors. After final adjustment, facial landmarking algorithm becomes ready as an automatic initialization step for facial landmark trackers in various face videos.

## 4. MULTI-STEP FACIAL LANDMARK TRACKING ALGORITHM

Robust and accurate facial landmark detection and tracking is an essential step for an effective automatic FHG analysis system. In this chapter, a multi-stage robust facial landmark tracking framework is described. The tracker is initialized with a robust and accurate facial landmark detector, as discussed in Chapter 3. The tracking algorithm is basically composed of appearance-and model-based four successive steps namely: landmark location prediction using Kalman filter, block matching with template library, regularization with multi-pose shape models and a final refinement. Naturally occurring facial videos may include severe head rotations with either strong or subtle facial appearance changes. To cope with these challenges, a multi-step tracking algorithm is introduced in order to handle changes in the geometry of the tracked landmark points under head rotations (pitch and yaw angles up to  $\pm 45^\circ$ ) and facial expressions. Then the trajectories (tracked positions) of facial landmark positions extracted during the course of the head gesture or facial expression within the video are utilized in order to extract discriminative features.

The general flow chart of the proposed multi-step tracking algorithm is illustrated in Figure 4.1). The steps of the facial landmark tracking algorithm are discussed in the following sections. Figure 4.2 illustrates the 52 manually landmarked points from the BUHMAP Database [9] and the 17 tracked points with our tracking method.

### 4.1. Kalman Prediction

The Kalman filter has been used extensively for motion prediction in computer vision applications [115, 116, 72, 117, 118, 119]. In the proposed tracking system, we predict the locations of each landmark point using Kalman filtering. In addition to Kalman filtering, particle filtering [120] or extended Kalman filtering [121] methods can also be used. We choose Kalman filtering method because of its simplicity and its low computational cost. The predicted landmark locations reduces the computational

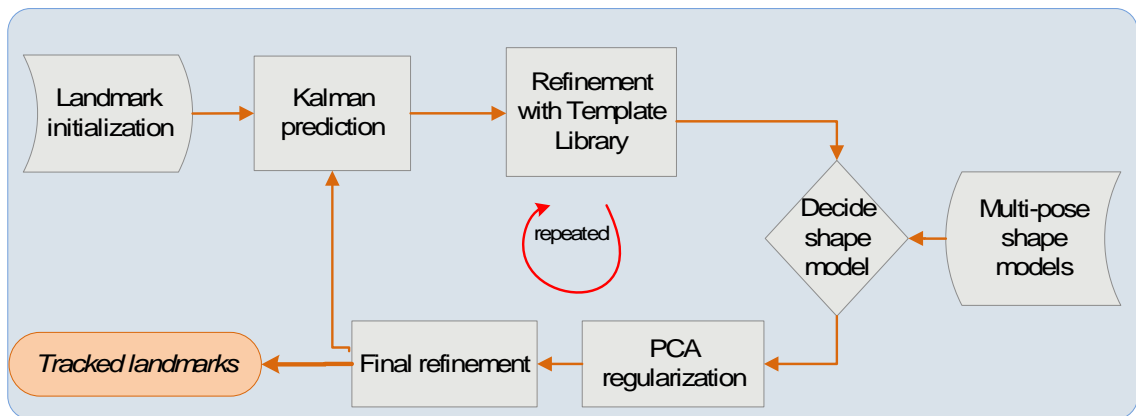


Figure 4.1. The general flow diagram of the proposed landmark tracking algorithm

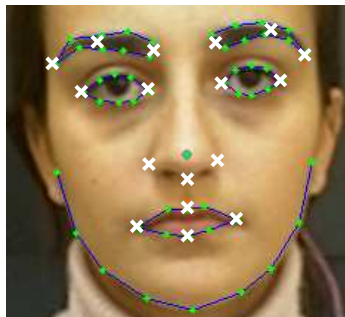


Figure 4.2. 52 manually annotated landmarks (dots), 17 landmarks tracked by our algorithm (crosses)

load and increases the accuracy of the template library based matching by predicting the landmark locations.

In this study, the locations of the identified landmark points are predicted using the Kalman filter toolbox [122] for each subsequent frame. In this algorithm, the estimated instantaneous facial landmark point positions  $(x^j, y^j)$ , for  $j = 1, \dots, 17$  at time  $k$  are fed into a linear Kalman filter and we expect Kalman filter to produce an estimate at time  $(k + 1)$ . We model the temporal changes of facial landmark positions  $(x^j, y^j)$  as 17 independent 2D translations within the image plane. Let  $X_k$  denote the state vector at instant  $k$  for the  $j^{th}$  landmark, which is to be estimated. The system model for tracking a generic landmark point (any of the 17 we have chosen) is a constant

acceleration model:

$$X_{(k+1)} = A_s X_{(k)} + w_{(k)}, \quad (4.1)$$

$$X_{(k)} = \begin{bmatrix} x_{(k)} & y_{(k)} & \dot{x}_{(k)} & \dot{y}_{(k)} & \ddot{x}_{(k)} & \ddot{y}_{(k)} \end{bmatrix},$$

where  $X_{(k)}$  is the state vector of the process at time  $k$ ,  $(x_{(k)}, y_{(k)})$ ,  $(\dot{x}_{(k)}, \dot{y}_{(k)})$  and  $(\ddot{x}_{(k)}, \ddot{y}_{(k)})$  represent the position, speed and acceleration in the  $x$  and  $y$  directions respectively and  $A_s$  is the state transition matrix of the process from the state at time  $k$  to the state at time  $k+1$  and  $w_{(k)}$  is the vector of process noise. Observations on this variable can be modeled in the form;

$$Z_{(k)} = HX_{(k)} + v_{(k)}, \quad (4.2)$$

where;  $Z_{(k)}$  is the actual measurement of  $X$  at time  $k$ ,  $H$  is the noiseless connection between the state vector and the measurement vector and  $v_{(k)}$  is the associated measurement error. These ideas are summarized in matrix state space form as

$$\begin{bmatrix} x_{(k+1)} \\ y_{(k+1)} \\ \dot{x}_{(k+1)} \\ \dot{y}_{(k+1)} \\ \ddot{x}_{(k+1)} \\ \ddot{y}_{(k+1)} \end{bmatrix} = \begin{bmatrix} 1 & 0 & dt & 0 & dt^2/2 & 0 \\ 0 & 1 & 0 & dt & 0 & dt^2/2 \\ 0 & 0 & 1 & 0 & dt & 0 \\ 0 & 0 & 0 & 1 & 0 & dt \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{(k)} \\ y_{(k)} \\ \dot{x}_{(k)} \\ \dot{y}_{(k)} \\ \ddot{x}_{(k)} \\ \ddot{y}_{(k)} \end{bmatrix} + w_{(k)},$$

$$Z_{(k)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_{(k)} \\ y_{(k)} \\ \dot{x}_{(k)} \\ \dot{y}_{(k)} \\ \ddot{x}_{(k)} \\ \ddot{y}_{(k)} \end{bmatrix} + v_{(k)}.$$

The random variables  $w_{(k)}$  and  $v_{(k)}$  are assumed to be mutually independent of each other and process noise  $w_{(k)}$  is also assumed to be white and with normal probability distribution

$$p(w) \sim N(0, \zeta), \quad (4.3)$$

$$(4.4)$$

The process noise covariance  $\zeta$  and measurement noise  $R$  can be initialized as

$$\zeta = E\{w_k w_j^T\} = \zeta \delta_{kj}, \quad \text{with} \quad \delta_{kj} = \begin{cases} 1 & k = j \\ 0 & k \neq j \end{cases},$$

$$E\{w_k v_j^T\} = 0,$$

$$R = E\{v_k v_j^T\} = r \delta_{kj},$$

$$E w_k = E v_k = 0.$$

In this work, process noise variance  $\zeta$  is chosen as 0.1 and it was kept constant during tracking. However in practice, process noise variance  $\zeta$  might change with each time step. This variance is used to model not only the uncertainties in model parameters but also deviation of tracked landmarks from their assumed trajectories. Hence, the magnitude of this variance depends on the dynamics of the tracked landmarks. Covariance of measurement noise  $R$  for the  $j^{th}$  landmark was initialized using the tracked

positions of that landmark in the first two frames within the video using

$$p^j = \begin{bmatrix} (x_{j,1} & y_{j,1}) \\ (x_{j,2} & y_{j,2}) \end{bmatrix},$$

$$R = E\{(p^j - p_{ave}^j)(p^j - p_{ave}^j)^T\}, j = 1, \dots, 17.$$

The filter outputs consist of estimated  $(x_j, y_j)$  coordinates of the landmark points for the next frame of the video. In this work Murphy's Kalman Toolbox [122] was used. The contribution of the Kalman prediction step to the tracking accuracy is presented in the results section of this chapter.

## 4.2. Template Library

### 4.2.1. Refinement with Template

We use more than one template, i.e., a template library per landmark in order to adequately represent varying appearances of each landmark. The dimensions of the templates are chosen as one fourth of the IOD, as determined in the initial frame. Fig. 4.3 demonstrates some of the chosen templates on the face image. As a follow-up to Kalman prediction, the position of each landmark is refined by finding the best match in its template library. This refinement stage is intended to compensate for nonlinear local displacements, which are not corrected by the Kalman filter. The refinement operation is a generalization of the block matching used in motion estimation, which searches for the best match between image block in the current frame and candidate locations in the past frame [78]. In contrast, we allow for more latitude by searching the location of the landmark not only by using the current image block centered on the landmark in the previous frame, but in addition we use alternate templates of the landmark in an ever evolving library. Landmark template libraries are evolutionary in that the number and types of templates can change in time to adapt to novel appearances of landmarks as explained in the sequel.

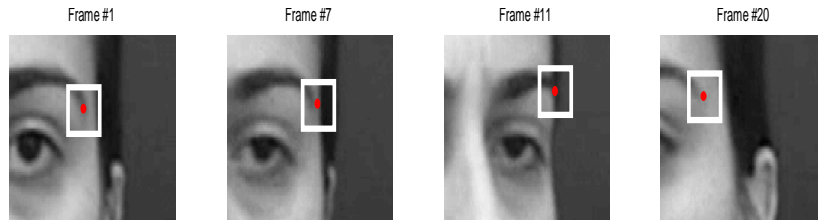


Figure 4.3. White rectangles represent the chosen templates for outer-eye corner in an image sequence

#### 4.2.2. Template Update

The template library of a landmark is updated whenever a landmark appearance is encountered which is significantly different from any in the present library. Fig. 4.3 illustrates the appearance changes of a landmark (outer-eye corner) in a sequence of images, and justifies the landmark template adaptation. We start with a single template in the library, the one obtained during landmark detection in the first frame. In the subsequent frames, new DCT templates can be added to the library based on a (dis)similarity measure. We have adopted the normalized cross correlation ( $NCC$ ) coefficient as in Eq. 4.5 for similarity measure between templates.

$$\begin{aligned}
 Dist(T_{test}, T_{library}^k) &= 1 - \{NCC\{T_{test}, T_{library}^k\}\} \\
 &= 1 - \left\{ \frac{\langle T_{test}, T_{library}^k \rangle}{\|T_{test}\| \|T_{library}^k\|} \right\}, \tag{4.5}
 \end{aligned}$$

$$k = 1, \dots, N, \quad N = |T_{library}|$$

where  $NCC$  is the *Normalized Correlation Coefficient*,  $\langle \cdot, \cdot \rangle$  is the inner product and  $\|\cdot\|$  is the  $L - 2$  norm. There are 17 landmark libraries and each library contains a dynamic number of templates,  $N$ , given by the cardinality of the corresponding template set.

If the appearance of a tracked landmark as observed within a template window (here denoted as  $T_{test}$ ) is sufficiently similar ( $NCC > 0.8$ ), there is no need to update. If the maximum of the correlation between existing templates and the landmark

Figure 4.4. Template Library Update Algorithm

```

for for all 17 landmarks do
  for  $k = 1$  to  $N$  do
     $distance(k) = Dist(T_{test}, T_{library}^k)$ 
  end for
  if  $0.3 \leq \min(distance) \leq 0.8$  then
    add  $T_{test}$  to  $T_{library}$ 
  end if
end for

{here  $T_{test}$  is the template of located landmark for the current frame}

```

appearance is  $0.3 < NCC < 0.8$ , then the DCT template of the found landmark is included in the library to enrich it. In the case the maximum  $NCC$  is less than 0.3, then it is not added to the template library lest we incur an erroneously tracked case. Template library update is summarized in Figure 4.4. The lower threshold is set to limit the maximum allowed distance (minimum similarity) of a candidate template beyond which the appearance is considered to be an outlier. Conversely any appearance which has  $NCC$  higher than 0.8 is not considered novel enough to be included in the library.

In summary, the position indicated by the template with minimum distance score to all patches tested within a search window is declared as the new landmark location. The test patch and the search window concepts are illustrated in Fig. 4.5. The search window is centered on the spot predicted by the Kalman filter, or if Kalman predictor were not to be used, simply on the landmark spot in the previous frame. The test patch visits every position within the search window. Figure 4.6 gives the search steps. In Figure 4.5, the  $NCC$  values for the right eye and the right eyebrow landmarks are illustrated in gray tones within search boxes centered on the respective landmark points. Degree of match in terms of  $NCC$  is given by gray tones where bright pixels mean better matches. For clarity the patch (with red dash-dot line) with highest  $NCC$  score is shown for each landmark. Notice that we search by moving the test patch to all locations where its center remains inside the search window. In our study, the size of

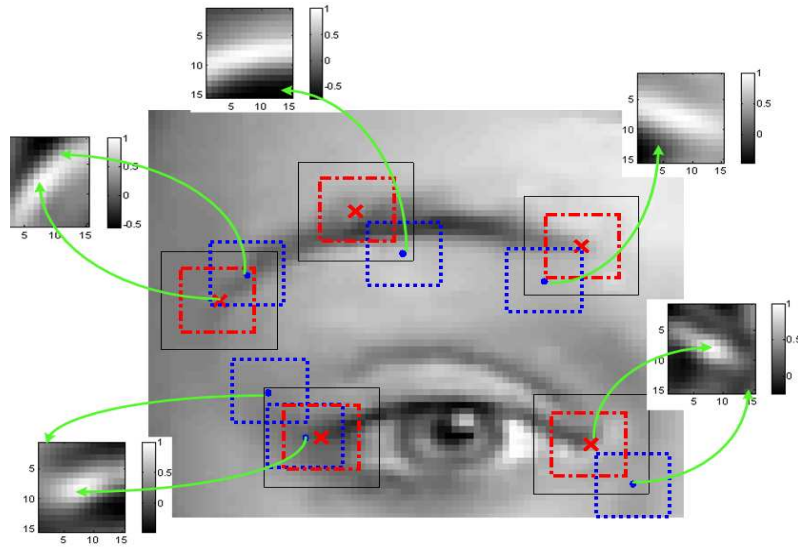


Figure 4.5. Illustration of similarity scores in search windows on a sample face image.

Solid box: search window; dotted window: a template in a test position; dash-dot window: template in optimal position.

the search window is based on the total displacements (displacement of the landmarks between last two consecutive frames) of the landmarks in x and y directions ( $\delta_x, \delta_y$ ) and the search window size is selected as two times of the total displacements ( $2\delta_x \times 2\delta_y$ ).

In Figures 4.7-4.8, we plot the index of the chosen template on the y-axis as one proceeds along the frames. The maximum template index on the y-axis shows the population size of the template library of the facial landmark for that gesture video. It can be observed that the population of template library is proportional to the variation in the appearance of the facial landmark in a given image sequence. For example, in head shaking video, the number of templates for the right and left outer eyebrow corners is four and five, respectively, while the number of templates for inner eye and inner eyebrow corners is only 1. Besides this, in smiling video, lip corners have the most populated template libraries. It is obvious that pulling lip corners changes the appearance of the lip corner area which is inherent to smiling action.

It is seen that the first template, which is extracted in the initial frame, is the most dominant template for all landmarks and its percentage of usage changes between

Figure 4.6. Template Matching Algorithm

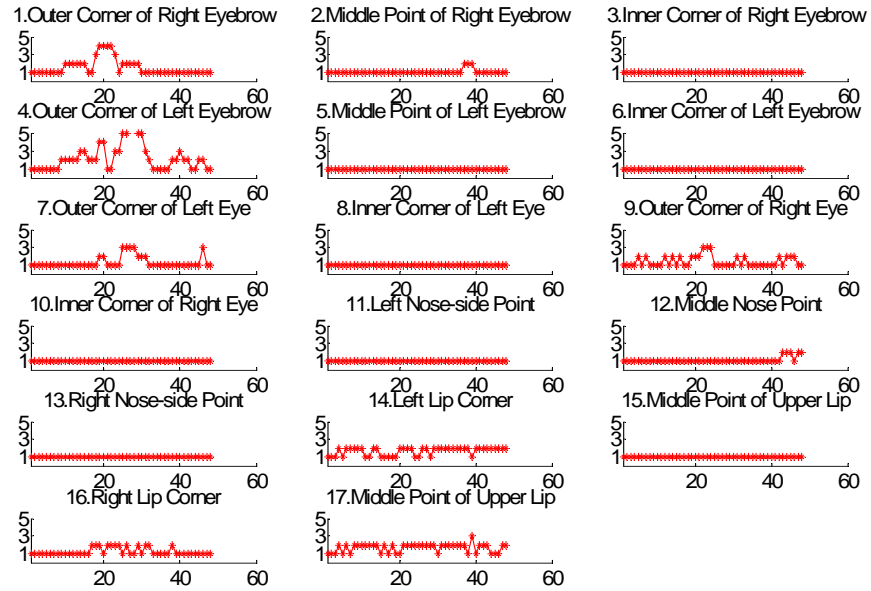
```

for for all 17 landmarks do
  for  $k = 1$  to  $N$  do
     $distance(k) = Dist(T_{test}, T_{library}^k)$ 
  end for
  {find most similar template index}
   $ind = \underset{k}{\operatorname{argmin}}(Dist(T_{test}, T_{library}^k))$ 
  {search  $T_{library}^{ind}$  within a window centered at the predicted position}
  for  $i = 1$  to  $WindowSize$  do
     $WindowScore(i) = Dist(T_{window^i}, T_{library}^{ind})$ 
  end for
   $New\ landmark\ position \leftarrow \underset{i}{\operatorname{argmin}}(WindowScore(i))$ 
end for
  {here  $T_{test}$  is the template of located landmark in the previous frame}

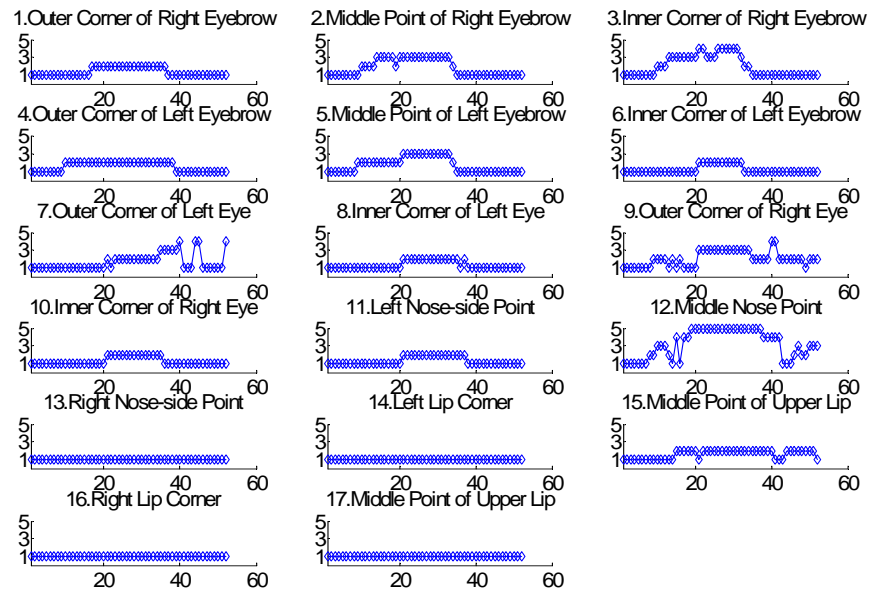
```

53.7% and 91.4%. The most dominant template usage belongs to the inner eye corner pattern, since the variation in the appearance of inner eye corner pattern is less than the other facial features for head and facial gestures.

Table 4.1 shows the percentages of template usage for the 17 landmarks estimated using seven head and facial gesture videos of a subject. As expected the template library sizes reflect the variability of a landmark. For example, inner eye corners vary less and the first two templates explain more than 90% of the cases, while the more variable lip corners need three to four templates to reach that percentage. The first template introduced in the initial frame is the dominant template for all landmarks and its percentage of usage varies between a low of 56% for lip corners to a high of 88% for eye inner corners. In conclusion template libraries have never grown beyond 5 at least for the BUHMAP database [9]. Further growth can always be trimmed on the basis of template usage statistics.

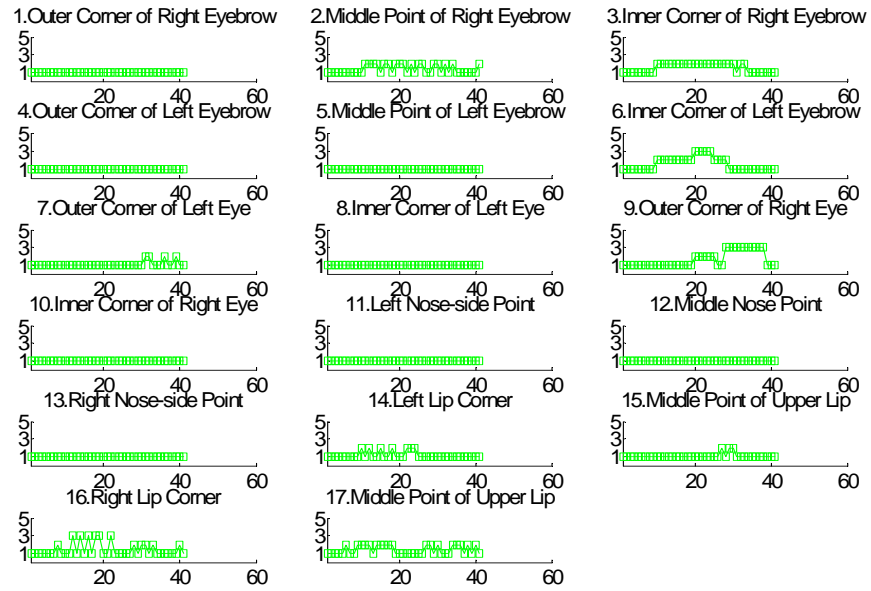


(a)

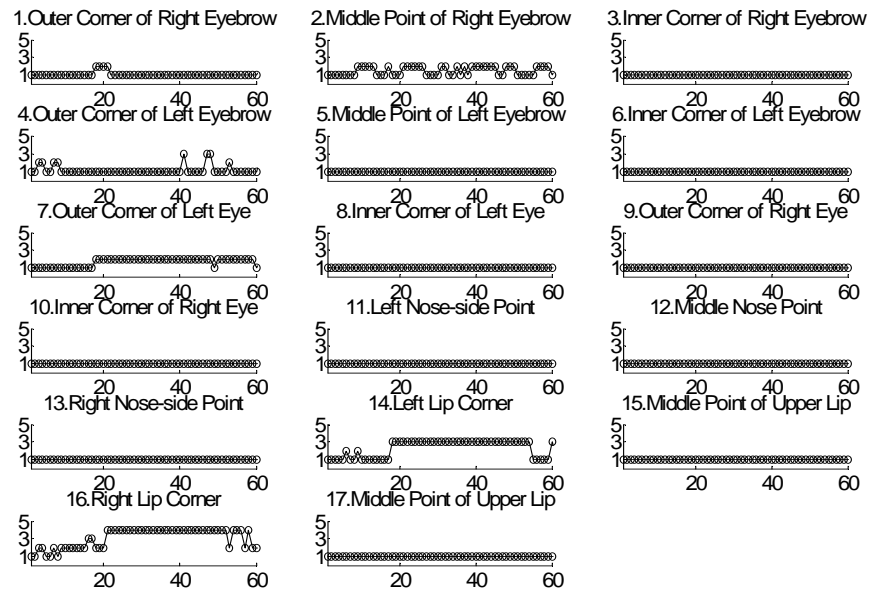


(b)

Figure 4.7. Template library information for sample videos:(a) Head-shaking video, (b) Head up and eyebrow raise video, (x-axis : frame index, y-axis : index of the chosen template)



(a)



(b)

Figure 4.8. Template library information for sample videos:(a) Head forward and eyebrow raise video, (b) Smiling video (x-axis : frame index, y-axis : index of the chosen template)

Table 4.1. Percentage of templates usage (%)

Landmark index	Template index and percentages				
	1 <sup>st</sup>	2 <sup>nd</sup>	3 <sup>rd</sup>	4 <sup>th</sup>	5 <sup>th</sup>
1 - outer corner of right eyebrow	65	26.8	6.9	1.1	0
2- midpoint of right eyebrow	77	17.7	5.3	0	0
3- inner corner of right eyebrow	79.5	11.1	7.2	2.2	0
4- outer corner of left eyebrow	54.5	33.8	9.4	0.6	1.4
5- midpoint of left eyebrow	83.9	12.5	3.6	0	0
6- inner corner of left eyebrow	85	13.6	1.4	0	0
7- outer corner of left eye	75.9	20.2	2.8	1.1	0
8- inner corner of left eye	85	6.4	8.6	0	0
9- outer corner of right eye	62	25.7	11.3	0.8	0
10- inner corner of right eye	91.4	8.6	0	0	0
11- right nose wing	85	14.9	0	0	0
12- middle nose point	78.9	11.6	1.9	2.2	5.3
13- left nose wing	89.7	10.2	0	0	0
14- right lip corner	59.8	27.9	11.6	0.5	0
15- midpoint of upper lip	70.3	15.5	5.5	6.9	0.6
16- left lip corner	53.7	23.2	8.3	14.1	0.6
17- midpoint of lower lip	72.8	20.2	5	1.4	0.6

### 4.3. PCA Regularization with Multi-Pose Shape Model

Landmarks can occasionally go astray due to occlusion, interfering patterns or inadequacy of templates, which fail to represent all variations. To minimize this risk, we project the ensemble of all 17 ( $n$ ) landmark points, that is their configural information on the shape subspace following the Point Distribution Models (PDM) paradigm [81, 63]. PDMs provide a convenient method for representing flexible objects by means of a set of feature points. Once landmark locations have been updated, their  $2 \times 17$  dimensional coordinate vector is projected onto the PCA subspace so that excessive shape deformations can be trimmed back to the regularized shape space. In other words, regularization is used to prevent drifting of one or more feature points to an irrational shape, that is, a non-face shape, and hence pull them in the range of learned face shape space.

However landmark configuration variations induced by pose changes are too severe to be handled by a single PDM. We found it necessary to build multi-pose shape model set. Therefore we used three separate shape models for the yaw angle ranges of  $(-20^\circ$  to  $20^\circ)$ ,  $(-45^\circ$  to  $-20^\circ)$  and  $(20^\circ$  to  $45^\circ)$ , respectively. Since yaw is the source of major variation in the templates, splitting the range into three enables better control of shape variations.

The three shape models (landmark configurations) have been learned by using 1208 posed face images with manually annotated landmark locations (subset of Bosphorus Face Database [8]). Sample images under head pose and facial expressions can be seen in Fig. 4.9. We have observed that video frames where head shaking occurs with large yaw sweeps,  $\pm 30^\circ$  or greater are more accurately handled with three-posed model. Instances where the second, first and third shape poses occur are shown in Fig. 4.10.

The construction of the new shape spaces consist of the following steps:

- (i) Concatenate the x and y coordinates of annotated landmarks of the training dataset [8] to form a 34-dimensional shape vectors.



Figure 4.9. Sample face images from Bosphorus Database [8] illustrating various head poses and facial actions



Figure 4.10. Tracked frames with different pose models: right yaw, frontal or near frontal, left yaw

(ii) Organize the shape vectors into a shape data matrix  $\mathbf{S}$ .

$$\mathbf{S} = \begin{bmatrix} x_1^1 & x_1^2 & \dots & x_1^n \\ \vdots & \vdots & & \vdots \\ x_{17}^1 & x_{17}^2 & \dots & x_{17}^n \\ y_1^1 & y_1^2 & \dots & y_1^n \\ \vdots & \vdots & & \vdots \\ y_{17}^1 & y_{17}^2 & \dots & y_{17}^n \end{bmatrix}$$

(iii) Use the Procrustes transform [65] to align the shape vectors of an ensemble of  $2n$  shape instances to minimize translation, rotation and scale differences.

- Each training shape vector is translated so that its center is at the origin.
- Choose one of them as a mean shape and scale so that  $\|\bar{\mathbf{S}}\| = 1$  and record it as reference frame.
- Align all shapes with the current estimate of the mean shape. Alignment is accomplished by solving the least squares fit problem.
- Re-estimate the mean from the aligned shapes, and align it with reference frame and scale so that  $\|\bar{\mathbf{S}}\| = 1$ .
- If the mean  $\|\bar{\mathbf{S}}\|$  has not changed significantly then stop, else go to step (iii).

(iv) Using the PCA method obtain shape eigenvectors that incorporate 98% of the energy. In our work we found the subspace dimension to be 20.

The shape eigenvector matrix  $\Phi$  and eigenvalue matrix  $\rho$  are obtained from the eigen-decomposition of the shape covariance matrix,  $\mathbf{C} = \mathbf{E}\rho\mathbf{E}^T$  where  $\mathbf{E} = (\mathbf{S} - \bar{\mathbf{S}})(\mathbf{S} - \bar{\mathbf{S}})^T$ .

Aligned points represent an exemplar shape, free of translation, rotation and scale variation. The dimensionality (2x17) of shape space is reduced by applying PCA. The shape model can be fitted to the new shape by finding a suitable transformation and shape model parameters. Then the likelihood of the new shape can be estimated. Any given shape  $X$  is regularized by projection on this subspace, that is:

(i) Initialize the shape parameters,  $b$ , to zero.

- (ii) Generate the model  $\hat{X} = \bar{S} + \mathbf{E}b$ ,
- (iii) Align the new shape to the mean shape  $\bar{S}$ ,
- (iv) Find the model parameters  $b$  to match the new aligned shape  $\hat{X}$ , by
 
$$b = \mathbf{E}^T(\hat{X} - \bar{S}),$$
- (v) Apply constraints on  $b$ , if not converged, return to step (2).
- (vi) Finally,  $\hat{X} \approx \bar{S} + \mathbf{E}b$ ,

Dryden and Mardia [65] stated that the individual parameters  $b_i$  can be assumed to be independent and Gaussian. The variance across the training set of an individual parameter  $b_i$  is given by  $\rho_i$ . Therefore face shapes similar to the training shapes can be generated by varying the parameters  $b_i$ . We augmented the range of the  $b_i$  parameters empirically such that  $-\frac{2}{1+0.2i}\sqrt{\rho_i} < b_i < \frac{2}{1+0.2i}\sqrt{\rho_i}$ ,  $i = 1, \dots, 20$ . Thus the perturbation limits of the first parameter  $b_1$  becomes  $\pm 1.67\sqrt{\rho_1}$  while it is  $\pm 0.4\sqrt{\rho_{20}}$  for the last parameter  $b_{20}$ . In Cootes et al. [63] the variation limit is set at  $\pm 3(\rho_i)^{1/2}$  for the  $b_i$  parameters to ensure that the generated shape is similar to the training shapes. For our training shape data, we observed that if the  $b_i$  parameters are selected within the range suggested as in ASM's [63], e.g  $\pm 3(\rho_i)^{1/2}$ , unplausible face shapes can be generated as shown in Figure 4.11. It is seen that these generated face shapes are not plausible. For example, the middle nose point is represented very close to the lower lip middle point on Figure 4.11-a and the inner eyebrow points are very close to inner eye corner points in the right shape of the upper row. Obviously these are not valid and realistic face shapes.

In Figure 4.12, the first six modes of the frontal shape model are illustrated in order to correlate the perturbation on a specific mode with the changes in face shape. Details are given in the figure caption.

By narrowing the interval of the  $b_i$  parameters, we may obtain more plausible face shapes, on the other hand we may lose the tracking potential especially for lip and eyebrow landmarks. We have found it useful to do a final refinement in a narrower search area (5x5 search windows) via SVM-classifiers trained with DCT-based features. We use the same landmark specific SVMs already trained in the landmark initialization

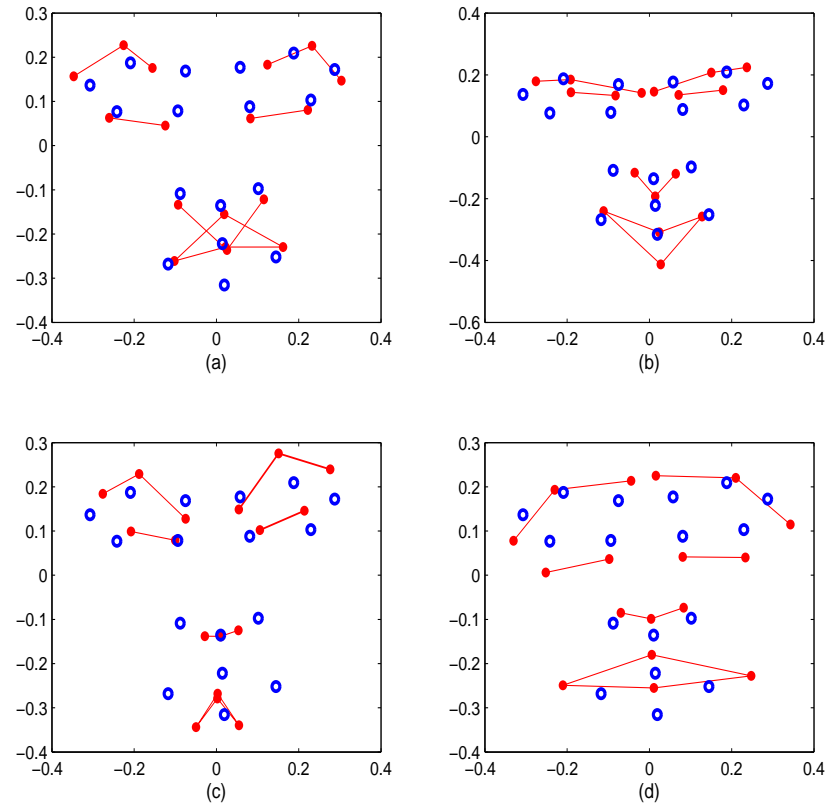


Figure 4.11. Lines: generated face shapes ( $b_i$  parameters are limit points), Circles: mean face shape for the frontal faces, (a) nose middle point is below the midpoint of upper lip, (b) inner eye and eyebrow points are very close to each other, (c) midpoint of upper and lower lip are almost overlapping and above the lip corners (d) mouth corners are almost on the same x-coordinate with outer eye corners and eyes and eyebrows are far away from each other.

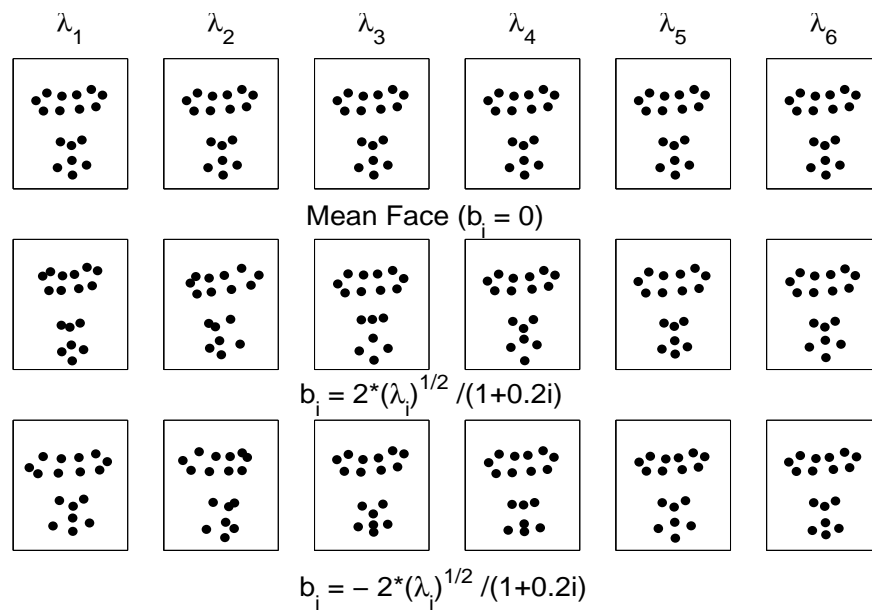


Figure 4.12. Generated face shapes under perturbations of single modes for near frontal faces. The first mode affects the width of the face, the second mode affects the yaw angle of the head, the third and fourth modes affect the pitch angles and mouth states (open or close) and the fifth and sixth modes affect the eyebrow movements of the frontal face shape model.

stage as in Section 3.3.

In Figure 4.13 you can see the contribution of each tracking step to the facial landmark tracking in a sample frame of a gesture sequence.

PGM-I or PGM-II algorithms that are tuned to eliminate false alarms and recuperate missing landmarks by using statistical structural information, can be alternative to PCA based shape regularization. But we choose the PCA-based shape regularization method for the tracking algorithm because PGM-I and PGM-II algorithms are designed to solve a combinatorial problem which has a high computational load. Note that the computation time of unoptimized MATLAB code written for multi-step tracking algorithm is about 3.5 seconds per frame whereas the lowest computation time for the PGM-based detection algorithm is about 7 seconds.

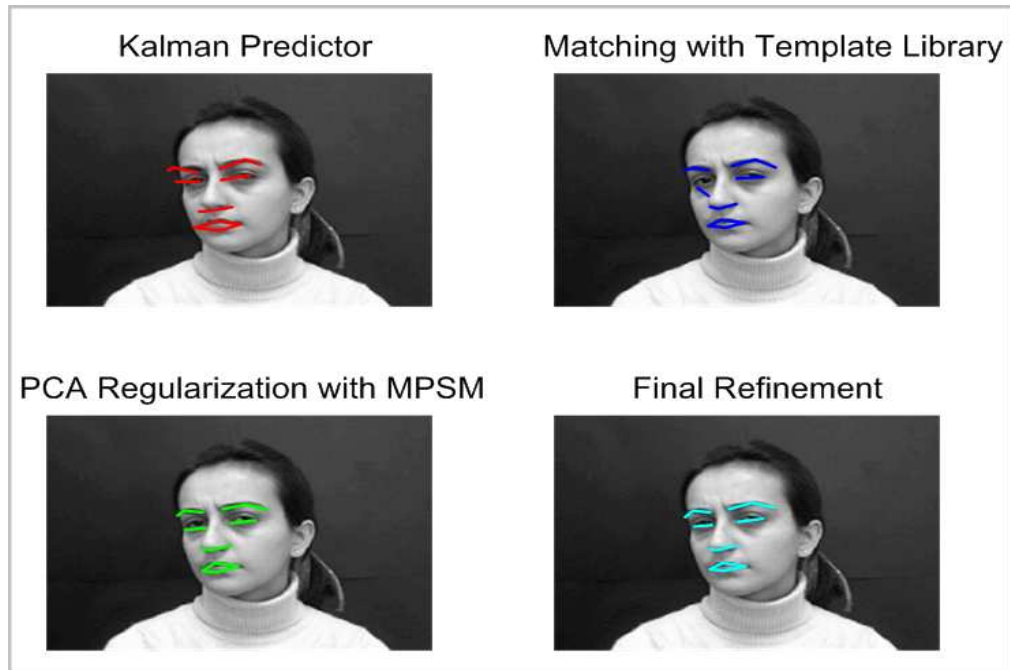


Figure 4.13. Contribution of tracking steps to the landmark tracking accuracy illustrated on a sample frame of a gesture video

#### 4.4. Facial Landmark Tracking Results

##### 4.4.1. Video database (BUHMAP)

We evaluated our automatic landmark tracking algorithm on the BUHMAP video database [9]. BUHMAP includes seven non-manual gesture classes (except neutral states) selected from Turkish Sign Language (TSL). The details of the gesture classes are given in Table 4.2. Our test set includes seven gesture types acted by eleven subjects, with five repetitions each, hence overall 385 video shots.

The videos are recorded at 30 fps at the resolution of 640x480. Each video starts and ends in the neutral state of the face.

Table 4.2. Head and facial gesture classes in BUHMAP DB [9]

Head shaking ( <b>G1</b> ):	Rotating head left and right sides repetitively
Head up ( <b>G2</b> ):	Tilting the head back while at the same time raising the eyebrows
Head forward ( <b>G3</b> ):	Moving head forward and raising eyebrows
Sadness ( <b>G4</b> ):	Lips turned down, eyebrows down
Head up-down ( <b>G5</b> ):	Nodding head repetitively
Happiness ( <b>G6</b> ):	Smile and expression of joy
Happy up-down ( <b>G7</b> ):	Nodding with smile

#### 4.4.2. Landmark tracking results

Figure 4.14 displays results of the proposed tracking and localization algorithm, when all the point-to-point distances between tracked points and their groundtruth are normalized by IOD for each frame. In these figures, the x-axis represents the normalized distance (error tolerance), while the y-axis represents the cumulative probability of success according to the groundtruth values. In each graph, the curve labeled as eyes represents the tracking and localization performance averaged over the four eye corners (inner and outer); the nose curve represents the performance averaged over three nose landmarks. The mouth curve represents the performance averaged over the four mouth landmarks; finally the eyebrows curve represents the tracking and localization performance averaged over the inner and outer eyebrow end points.

In the landmarking performance evaluations we have made use of a ground-truthed subset of the BUHMAP database [9]. This subset consisted of 3 repetitions of 4 gestures (head shaking, head up, head forward, happiness, and performed by 2 male and 2 female subjects, totally 48 video shots), which have been manually landmarked over 52 points. These 2880 frames (about 60 frames per video shot) were sufficient to give an idea about the performance of our landmark detector and tracker.

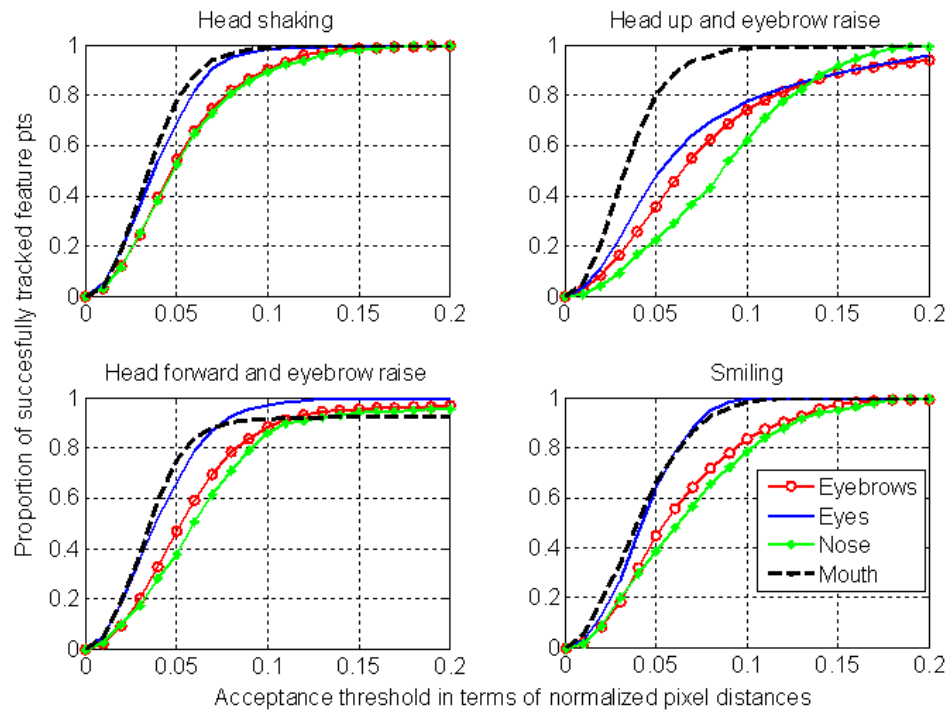


Figure 4.14. Cumulative performance of landmark errors vis-à-vis their ground-truth data. (Eyes: mean of the 4 eye points, Nose : approximated nose tip using tracked nose points, Mouth: mean of 4 lip points, Eyebrows: mean of the pair of outer and inner eyebrow points)

We have two observations. First, among the gestures in our experiment head shaking is the best tracked gesture class. Notice that almost all facial features are accurately tracked in the sense that digressions are smaller than 10 percent of the IOD. In contrast, tilting head back accompanied with eyebrow raise is the most difficult case especially for tracking eyebrows. This is largely due to the fact that in this gesture, eye and especially eyebrows suffer from perspective distortion.

The average tracking accuracy of four of the gesture classes is shown in Figure 4.15. These curves represent the mean absolute deviation in pixels averaged over all 17 landmarks. In addition, the 'grand' average over all gestures and landmarks is plotted. Recall that since the durations of gestures are not constant, landmark trajectories are time re-sampled to 60 frames. As expected H-U with eyebrow raise is the most difficult

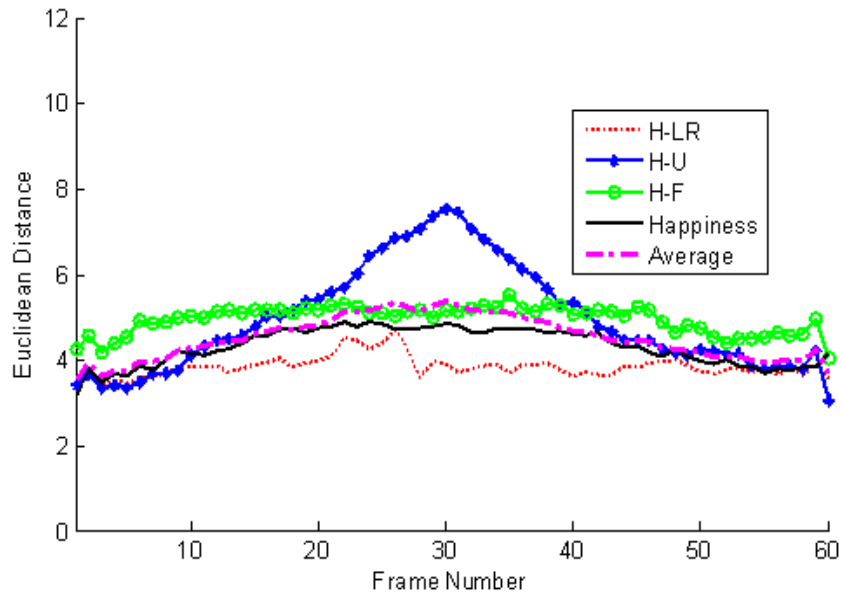


Figure 4.15. Mean Euclidean distance for each gesture class (averaged overall 17 landmarks) and the grand mean (averaged over gestures and landmarks)

case. The position discrepancy, averaged over all gesture classes, remains less than 4 pixels distance. This is a satisfactory performance. In fact, to give an idea, the average IOD is about 80.5 pixels with a standard deviation of 8.8 pixels, the average size of the eye is 37.3 pixels, and the  $0.1 \times \text{IOD}$  acceptability threshold is 8 pixels. Finally, the average detection performance of landmarks for all gestures at 0.1 IOD is about 88.4%.

Figure 4.16 demonstrates visual instances of the proposed tracking algorithm. This algorithm tracked the facial landmarks under large head rotations and various facial expressions robustly and accurately, even for test sequences that have not been used either in training part of the feature detection algorithm or in training part of the tracking algorithm.

#### 4.4.3. Contribution of the Tracking Steps

We also check the contribution of each step of the tracking algorithm. To reveal the contribution of each processing, we remove one of the steps and run the tracking



Figure 4.16. Tracked landmarks on sample image sequences; 1<sup>st</sup> row - G1 ; 2<sup>nd</sup> row - G2; 3<sup>rd</sup> row - G3; 4<sup>th</sup> row - G4; 5<sup>th</sup> row - G5; 6<sup>th</sup> row - G6; 7<sup>th</sup> row - G7;

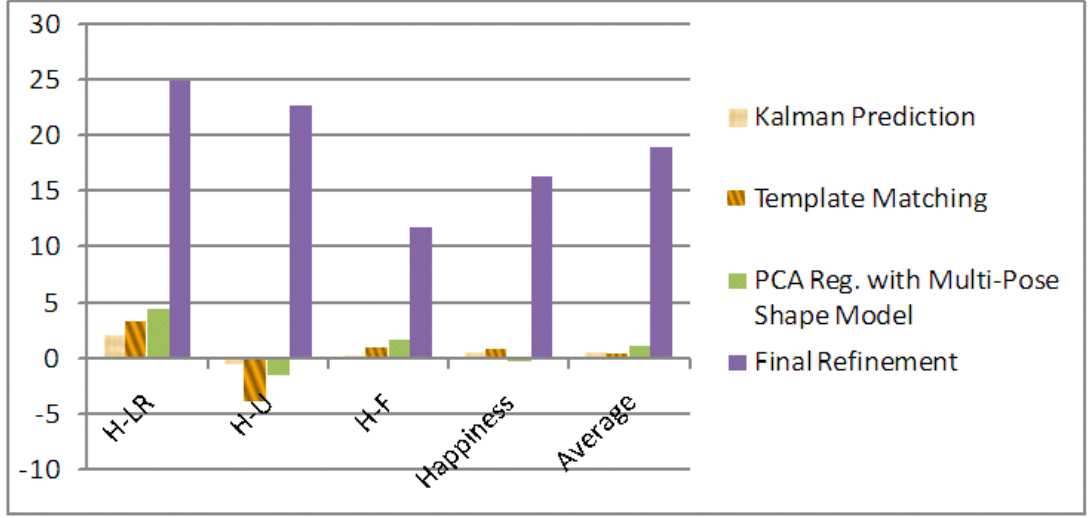


Figure 4.17. Contribution of tracking steps to the tracking performance in terms of AUCDC (H-LR head shaking; H-U head up and eyebrow raise; H-F head forward and eyebrow raise)

algorithm with the remaining steps. Then cumulative distribution curves are calculated respectively (e.g. Figure 4.14). In order to evaluate the contributions of each step to the tracking performance, we calculated the Areas Under Cumulative Performance Curves (AUCDC) using the formula

$$AUCDC = \frac{1}{T} \int_0^T Cumulative\_Performance\_Curve(x) dx. \quad (4.6)$$

where  $T = 0.1$ . With the removal of a step, AUCDC suffers and that the contribution of that processing step is calculated as

$$\Delta_{AUCDC} = AUCDC_{for-all-steps} - AUCDC_{one-step-removed} \quad (4.7)$$

where  $\Delta_{AUCDC}$  represents the contribution of the removed step. Figure 4.17 illustrates the contribution results in terms of  $\Delta_{AUCDC}$ . It is observed that *Kalman Prediction* step improves the tracking performance about 1%, *Template Matching* step improves the tracking performance about 0.3%, *Multi-Pose Shape Model* step improves the tracking performance about 1.3% and *Final Refinement* step improves the tracking performance about 18.9%. This test verifies that each tracking step has a contribution to

the tracking accuracy on almost all tested head and facial gesture videos. The major improvement in the tracking algorithm is achieved by final refinement step. A surprising result is that H-U gesture class is tracked %4 and %1.5 more accurately without *Template Matching* and *Multi-Pose Shape Model* steps, respectively. It is assumed that insufficient number of training data for head up action in shape models may induce this performance drop.

#### 4.5. Conclusions

Tracking facial landmarks under head rotations and local facial appearance deformations requires both low level modeling for facial deformations and high level adaptation to global shapes due to head rotations. Therefore, in this chapter, we have developed a multi-step facial landmark tracking algorithm successfully in order to handle simultaneous head rotations and facial expressions. The algorithm detects facial landmarks in the initial frame using DCT-features trained with SVM classifiers as described in Chapter 3, and then applies a multi-step tracking method based on Kalman predictor, adaptive templates, subspace regularization with multi-pose shape models and final refinement for the subsequent frames.

During tracking, Kalman filter predicts the landmark location for the subsequent frame by imposing a smooth constraint on the motion of each facial landmark separately. Template library is updated adaptively with facial appearance changes. Hence a subject-dependent template library is used for searching the best match of the previous template in the new frame. However due to large head rotations, a multi-pose shape model is learnt in order to prevent arbitrary deviations in spatial locations of facial landmarks. By applying multi-pose shape regularization, we may obtain more plausible face shapes, on the other hand, we may loose the tracking potential, especially for lip and eyebrow landmarks. Accuracy of the landmark locations is improved by using a final refinement in a narrower search area via SVM-classifiers trained with DCT-based features.

Sample frames are illustrated in Figure 4.16 to demonstrate results on head ges-

ture and facial expression sequences under out of plane head rotations and facial expressions. As a result, our developed tracking framework is robust to head rotations and facial appearance deformations. It can be effectively applied to face videos for head gesture and facial expression analysis.

## 5. CHARACTERISTICS AND FEATURES OF FACIAL EXPRESSIONS AND HEAD GESTURES

As we remarked in the introduction, the main challenging issue in affective state analysis is the lack of a defined grammar which will map a certain nonverbal message into a corresponding affective state. Hence, the same facial expression may mean different emotions and the same emotion may be expressed through different expressions. Furthermore, elicited non-verbal messages and the observed emotions have dependency on personality, society, state of the mood and also the context in which they are displayed or observed.

Acted or posed facial expressions represent facial behaviors that are recorded under controlled conditions and they mostly start with a neutral state. In naturally-occurring or spontaneous facial behaviors, more than one facial expression may be observed simultaneously. They may overlap with the other expressions and they mostly do not start with a neutral state. Psychological studies reveal that the acted emotions may differ in appearance and timing from corresponding spontaneous emotions [123, 124]. It is also observed that posed smiles lasts longer with higher amplitude than spontaneous smiles [124]. Ekman and et. al revealed that acted and spontaneous facial expressions are mediated from different neural pathways of the brain such that spontaneous facial movements originate in the cortical motor strip, whereas the more involuntary (acted) facial actions originate in the subcortical areas of the brain [94, 97]. Therefore, different neural pathways are effecting the intensity and the dynamics of the facial muscle movements and hence the facial expressions.

By considering the above properties, we should extract effective and reliable features from the given non-verbal message videos independent from their context whether they are posed or spontaneous.

### 5.1. Dynamics of the Facial expressions and Head Gestures

Even though FHGs may differ in total duration, they mostly follow a fixed pattern of temporal order. Some of the head gestures are periodic such as head nodding and head shaking. In such periodic gestures, the number of head-up, head-down or head-left, head-right cycles can vary slightly from acting to acting. Other gestures such as head-forward and those corresponding to basic expressions are episodic gestures. Episodic gestures have characteristic phases called onset, apex and offset. Onset defines the time from a neutral state to the peak of the action; apex phase defines the interval when the action is held at its peak; offset is the time to return from peak to a neutral state. The onset, apex and offset phases may have temporal variations depending upon the specific gestures and the intensity of the action.

Finally, there are gestures that are both episodic and periodic such as head nodding with smiling. Any FHG detector must take into account the variability in gesture duration, the intensity of the underlying actions, the variations due to individual actors and the context. In Figure 5.1, a facial expression sequence is illustrated with its tracked landmarks as a function of time index. We can roughly say that, time index  $k = 3$  to  $k = 5$  corresponds to the onset phase, time index  $k = 5$  to  $k = 10$  corresponds to the apex phase and time index  $k = 10$  to  $k = 12$  corresponds to the offset phase of the illustrated facial expression sequence.

### 5.2. Data Representation and Feature Types

We have used two types of data representations which are facial landmark trajectories and gray-level intensity image patches. Then, three types of features are extracted using the defined data representations. These extracted features can be briefly described as follows: (i) Landmark trajectory matrix which represents the automatically tracked landmark coordinates obtained during the course of facial gesture. (ii) Seventeen geometric features are composed by using distances, angles and their ratios obtained from the tracked landmark coordinates. (iii) Appearance features are extracted from the partitioned face intensity image. All these features are selected to



Figure 5.1. A facial expression sequence belonged to surprised emotion with time index  $k$ .  $k = 1$  to  $k = 2$  represents the neutral state,  $k = 3$  to  $k = 5$  represents the onset phase,  $k = 5$  to  $k = 10$  represents the apex phase and  $k = 10$  to  $k = 12$  represents the offset phase)

encode discriminative information about the underlying facial expression and gesture.

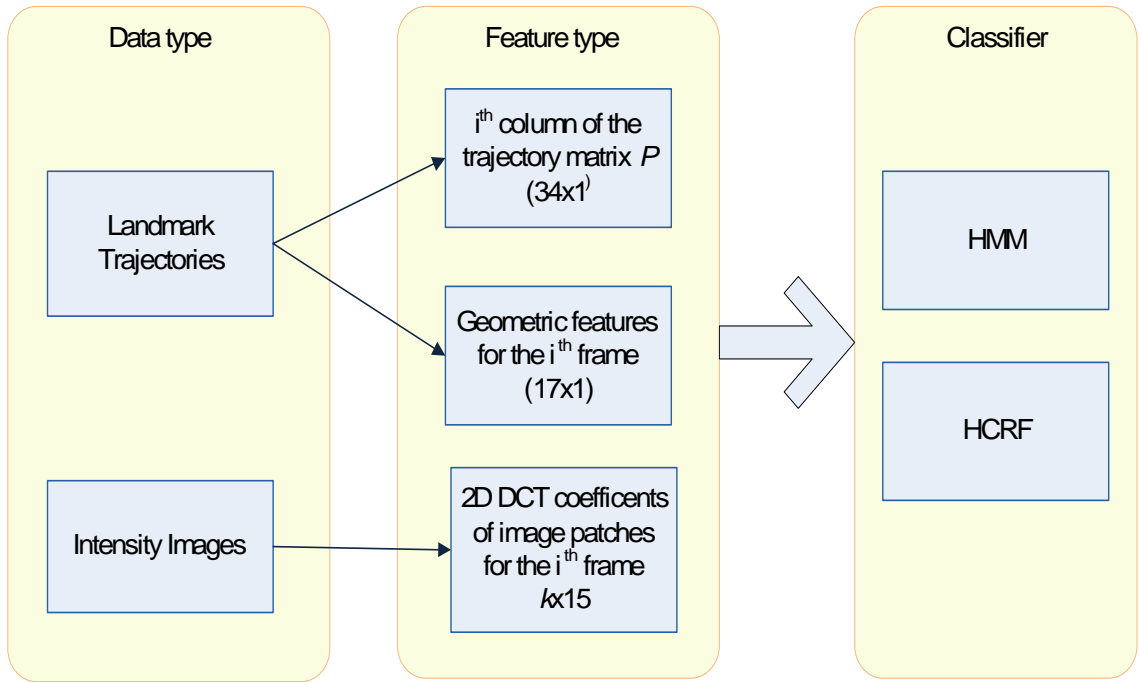
We interpret the extracted features in two different forms and hence two classification paradigms are employed to the extracted features. One of them is the sequential form, in which the features are extracted frame by frame and given into a sequential classifier like HMM or HCRF. In HMM or HCRF classification paradigm, for each time sample (frame of the video), an observation vector is composed and fed into the feature-sequence classifier. In the second form, extracted features are not processed frame by frame but a spatiotemporal feature matrix is obtained by observing the whole sequence and this spatiotemporal matrix is processed as a static image by employing subspace projection methods like ICA, NMF and DCT. Then these subspace features are classified by using a MNN classifier. The details of these extracted features will be explained in the following sections. Figure 5.2 gives an overview of the extracted features and their companion classification methods.

### 5.2.1. Facial Landmark Coordinates

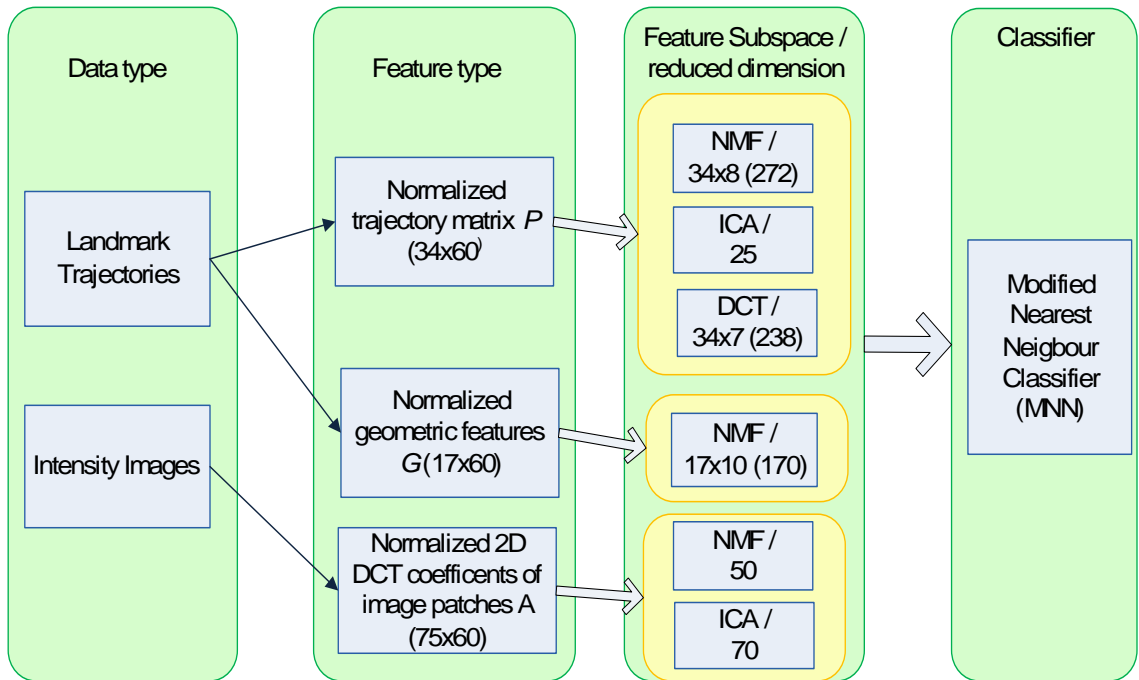
The landmarks tracked over a  $T$ -frame long gesture video is collected in a  $34 \times T$  dimensioned trajectory matrix  $P$ . Here, each row of the  $P$  matrix represents the time sequence of the x or y coordinates of one of the 17 landmarks. In order to obtain landmarks independent from the initial position of the head, the first column is subtracted from all columns of  $P$ , so that we only consider the relative landmark displacements with respect to the first frame. This presupposes that the landmark estimates in the first frame of the sequence are reliable. In addition, the minimum value of the manipulated  $P$  matrix, which will be a negative value after the subtraction from the first frame, is subtracted from the  $P$  matrix in order to obtain a non-negative  $P$  matrix for the NMF method.

$$P_{i,j} \leftarrow P_{i,j} - P_{i,1}, \quad \text{for } i = 1, \dots, 34, \quad \text{and } j = 1, \dots, T$$

where  $P_{i,j}$  is the  $i^{\text{th}}$  feature (landmark coordinate) at instant  $j$ .



(a)



(b)

Figure 5.2. (a) Sequential features and feature-sequence classifiers, (b) Subspace features and feature-subspace classifier

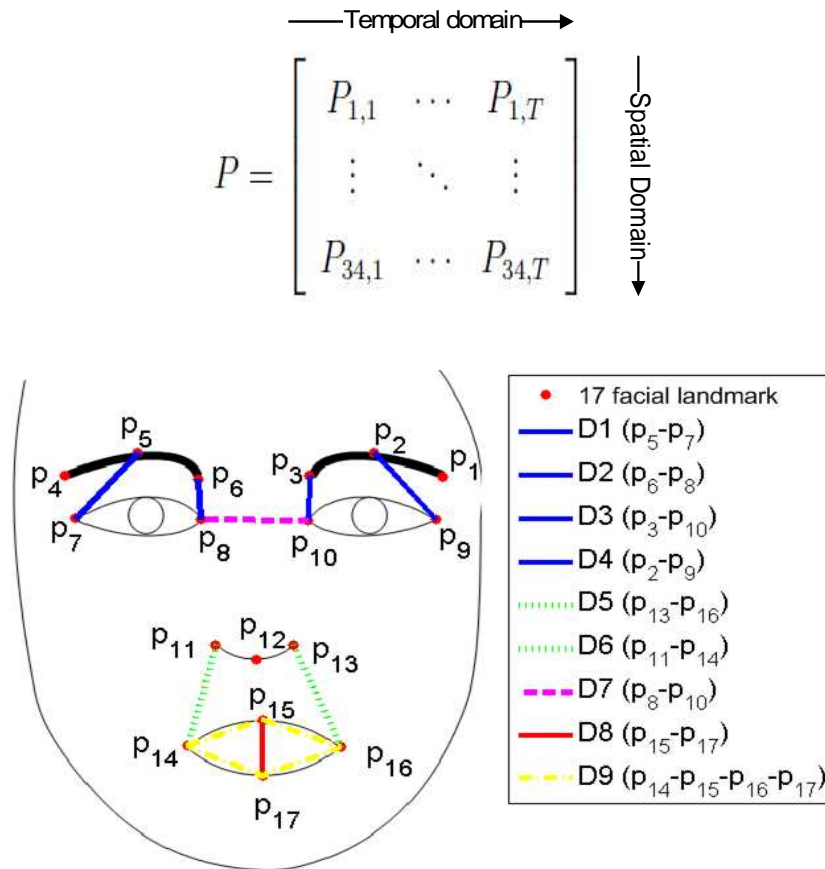


Figure 5.3. Some of the distances that are used for geometric feature extraction

### 5.2.2. Face Geometric Features

Several heuristic but effective features can be obtained using distances, angles and length ratios between chosen landmark pairs as illustrated in Figure 5.3. The reason of generating such features in lieu of using simply raw coordinate features is the conjecture that they are more gesture oriented and the assumption that they can encode the information of the underlying emotional state. It is conjectured that by tracking geometric features over an image sequence and analyzing their displacements over multiple frames, a characteristic motion pattern for given FHG can be established. In fact, Kaliouby [27], Sheerman-Chase et al. [11] and Hupont and et al. [89] also defined geometric measurements to capture spatial and motion characteristics of the emotional states. In our study, it is useful to note that geometric features defined in length, except the IOD itself (13<sup>th</sup> geometric feature in Tables 5.1-5.2), are normalized

with respect to the IOD to make them invariant to head size and zooming effect.

The geometric features extracted over a  $T$ -frame long gesture video is collected in a  $17xT$  dimensioned geometric feature matrix  $G$ .

### 5.2.3. Appearance based Features

Deformations occurring on the face during an expression involves holistic or local changes as well, such as in the mouth and eye regions. The estimated landmarks enable us to parse the face into regions of interest. We have heuristically selected four patches covering the most expressive facial parts, as shown in Figure 5.4. Patch sizes are chosen large enough to cover a whole expressive face region. Furthermore, patches are positioned using the tracked facial landmark locations. In that respect, sizes and semantic positions of the patches do not vary with changes in head orientation.

The two important regions are the eye and eyebrow patches and the mouth patch; the other two, the nose and the front patches, are somewhat less effective but still useful. The first patch is sensitive to expressions involving eyes and eyebrows; the nose patch does not get deformed but it will aid in tracking head gestures, like nodding and sideways shaking. The mouth patch is thought to be useful in differentiating the lip movements among happy, sad and neutral states. The fourth patch will be used for detecting expressive changes, that is, creases on the front occurring during eyebrow raise.

Extracted patches are also partitioned into blocks and each block is scaled into fixed block size (16x16) as in Table 5.3. The discriminative features from patches consist of DCT coefficients, not from the whole patch but from the 16x16 non-overlapping blocks tessellating the patch. Since the expressive eye region is critical, it is doubly covered. Beside the eye and eyebrow patches, one extra block (the region between the dotted lines, see Figure 5.4) that jointly covers them and that overlaps with the other two (16x16 DCT block) is used in order to encode the appearance changes between the eyebrows. We selected the first  $k$  DCT coefficients (after skipping the DC value)

Table 5.1. Face geometric features derived from tracked landmarks (Euclidean distances)

Feature Label	Geometric Features	Feature sensitive to
1,2	Mean x and y coordinates of the landmarks ( $x_{mean}, y_{mean}$ )	Head movements
3,4	Number of peaks (local maxima and minima) of $y_{mean}, x_{mean}$ trajectories over the observation epoch	Head movements
5	Distance between right and left lip corners	Lip states
6	Difference of y coordinates of lip corners between successive frames	Lip states
7	Sum of eyebrow-eye inner corner distance and eyebrow midpoint-eye outer corner distance, averaged over right and left parts ( $D1+D2+D3+D4$ )	Eye-eyebrow separation
8	Horizontal distance between eyebrow inner corners	Separation of eyebrows
9	Change in distance between eyebrow inner corners from initial frame (neutral and frontal) until current frame	Eyebrow motion
10,11	Difference of the coordinates of the nose middle point from initial to current frame	Displacement of the nose
12	The ratio of right and left eye widths, the width being defined as the distance between inner and outer eye corners	Head yaw
13	Distance between eye inner corners (D7)	Face zoom
14	Maximum of the right and left between eyebrow inner point-eye inner corner distances	Eyebrow raise

Table 5.2. Face geometric features derived from tracked landmarks (Euclidean distances) (Table 5.1 continued)

Feature Label	Geometric Features	Feature sensitive to
15	Vertical distance between middle points of lower and upper lips (D8)	Mouth opening or tightening
16	The perimeter of quadrilateral joining the four lip points (D9)	Lip configuration
17	The sum of distances between nose middle point and lip corners (D5+D6)	Lip shape

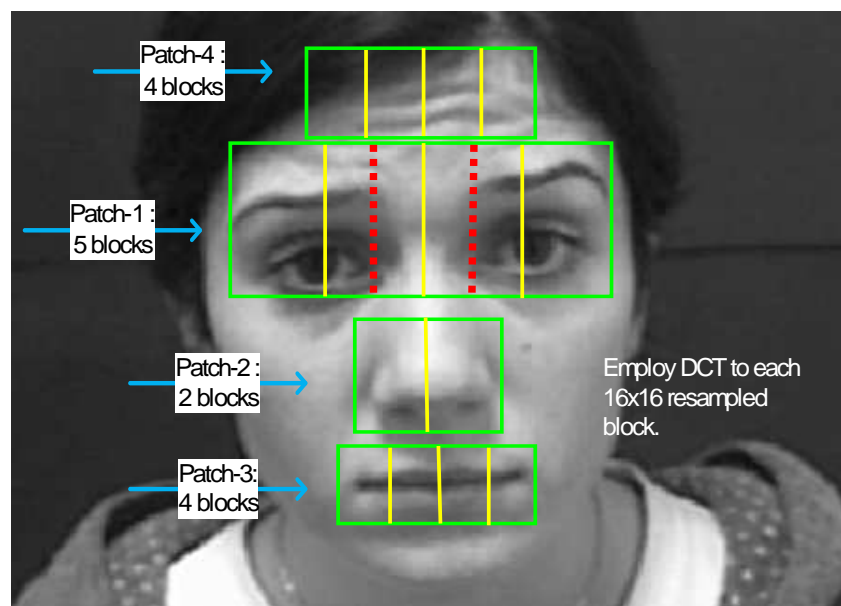


Figure 5.4. Facial patches and the corresponding blocks defined on a sample image

Table 5.3. Patches and the corresponding block sizes

Label of Patch	Region on the Face	Scaled Patch Size	number of 16x16 blocks
1	eyes and eyebrows	16x64	5
2	nose	16x32	2
3	mouth	16x64	4
4	forehead	16x64	4

$$\begin{array}{c}
 \xrightarrow{\text{Temporal Domain}} \\
 A = \begin{bmatrix}
 DCT_{1,1} & \cdots & DCT_{1,T} \\
 \vdots & \ddots & \vdots \\
 DCT_{(kx15),1} & \cdots & DCT_{(kx15),T}
 \end{bmatrix} \\
 \downarrow \text{Spatial Domain}
 \end{array}$$

Figure 5.5. Representation of  $A$  matrix which is composed of DCT features of image patches

from the zigzag order. All DCT block patterns are then concatenated into a single vector to form the feature vector  $k \times 15$  (total number of blocks). Since the patch-based,  $k \times 15$ -coefficient long appearance feature is extracted from each of the  $T$  frames, the gesture video thus generates  $(k \times 15) \times T$  dimensioned feature matrix  $A$ . As can be seen from Figure 5.5, the rows of the  $A$  matrix represents the temporal changes of the DCT coefficients and the columns represents the selected  $k \times 15$  DCT coefficients at time instant  $t$  (spatial features extracted at time  $t$ ).

### 5.3. Alternative Data Representation and Feature Types

Alternative to landmark trajectory matrices and 2D intensity images, we can derive and utilize many forms which are incorporating spatiotemporal discriminative information inherent to nonverbal-messages. One of the data representation form is spatiotemporal 3D face prism which is a novel data representation type for facial expression analysis [38]. The aim of this data representation form is to model motion and appearance jointly. This alternative data representation form and feature extraction methods, which will be described in this section, is applied only on facial expression recognition task. Cohn-Kanade facial expression database [71] is utilized in order to evaluate the performance of this data representation and feature extraction methods.

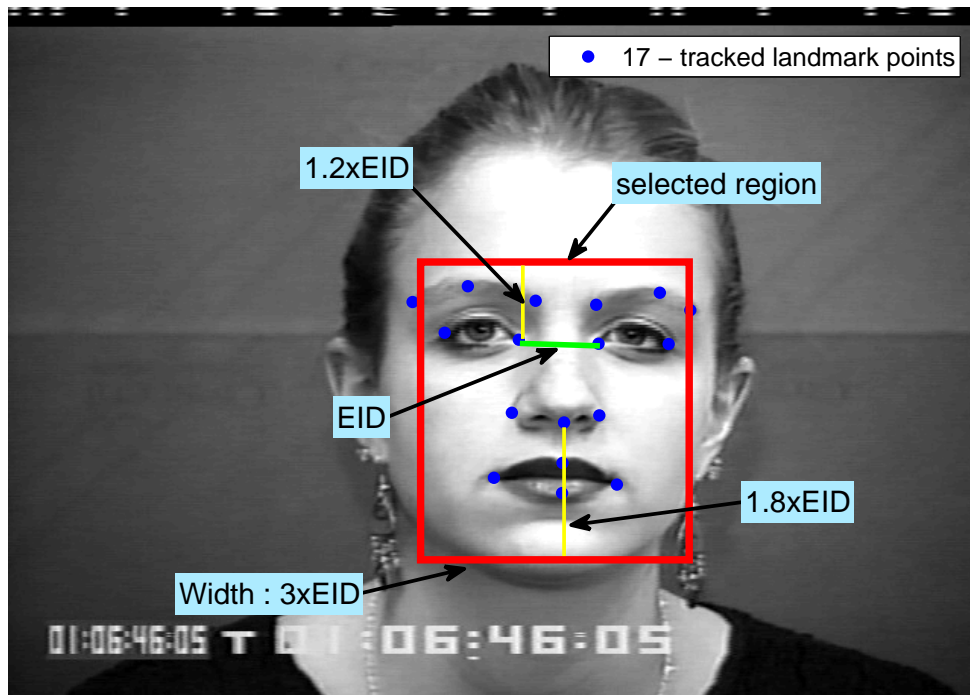


Figure 5.6. Facial landmarks to be detected and the cropped face region. The cropped area is dimensioned according to the EID

### 5.3.1. Spatiotemporal prism of face sequences

Spatiotemporal face prism can be composed by using automatically tracked or manually annotated facial landmark coordinates. Faces extracted from each frame are aligned and size normalized using the information of tracked landmarks. Alignment and size normalization is done for each video frame as follows : (i) Calculate the distance between inner eye corners (EID), (ii) Select the most informative region of the face using the landmark coordinates and the EID. In Figure 5.6, the selected region for one frame is shown on a sample image with its tracked landmarks. (iii) Crop the selected region of the face and scale this region to  $m \times n$  resolution facial patches (typically  $64 \times 48$  and  $64 \times 32$ ). These alignment and normalization steps are applied to the each frame.

These extracted patches are stacked to form a 3D face prism. This face prism is interpreted as volumetric data as illustrated in Figure 5.7 and consists of  $V_{m \times n \times T}$  voxels. We can apply the resample function to the pixels of aligned face pixels to obtain a  $T$ -long pixel trajectory, independent of the actual duration of the emotion

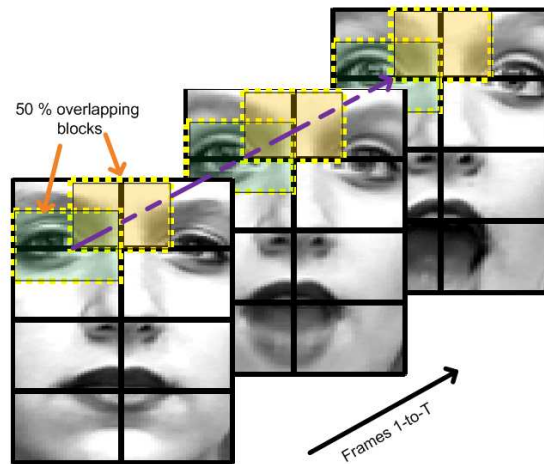


Figure 5.7. Illustration of a spatiotemporal prism over a three-frame instance. There is a total of 21 blocks of size  $16 \times 16$  extracted from the video shot. The blocks overlap by 50%.

sequence. In this study we chose  $T$  as 16. Notice that the frame lengths of the image sequences in the Cohn – Kanade database vary from 9 to 47 frames.

$V_{m \times n \times T}$  representation can also enable the use of subspace projections for feature extraction and classification. In this work we experimented with DCT due to its good energy compaction property and which can serve the purpose of summarizing and capturing the video content. Other model-based transforms such as Gabor and data-driven transforms PCA, ICA and NMF will be explored in a future work.

### 5.3.2. Features from spatiotemporal face cube

We can consider both global and local 3D DCT transform of the  $V$  matrix at two different resolutions. The details of global and local feature extraction and classification methods are as follows:

*Global 3D DCT transform:* In this case, 3D DCT is applied to the whole 3D

face prism  $V$  of sizes  $64 \times 48 \times 16$  and  $64 \times 32 \times 16$ . This results in 3-D DCT arrays containing, respectively 49152 and 32768 coefficients. We have selected low-frequency DCT coefficients using 3D zigzag order (excluding the DC term) and ordered them as a feature vector. Using 3D zigzag scan order we selected low order DCT coefficients with  $i, j, k$  indices such that  $1 \leq i + j + k \leq u$ . We determined  $u$  as 13 which roughly corresponds to 280 DCT coefficients of the face prism data. Notice that the selected DCT coefficients include only 0.5% and 0.8% of all coefficients.

*Block-based (local) 3D DCT transform:* Here we consider sub-prisms of the spatio-temporal  $V$  matrix. A sub-prism consists of a  $16 \times 16$  block on the face plane as was illustrated in Figure 5.7, and of the total temporal length  $T$ . Thus in fact sub-prisms become with this choice of dimensions  $16 \times 16 \times 16$  cubes, and each such cube is subjected to DCT. Notice that since the face blocks overlap by 50% the face is covered by  $B = 7 \times 5 = 35$  blocks for the  $64 \times 48$  sized crop and by  $B = 7 \times 3 = 21$  blocks for the  $64 \times 32$  sized crop. The  $16^3$  DCT coefficients are zigzag scanned and the first 280 DCT coefficients are selected from each transform cube. Finally the selected DCT coefficients of all cubes are concatenated into a single vector to serve as a feature vector  $q$  with dimension  $280 \times B$ .

The outcome of the DCT transform (global or local) is a set of transform vectors, one from each emotion sequence. In the case of global transform, the vectors of selected DCT coefficients form the feature vector themselves. Thus the feature vectors for the global DCT are 280-dimensional low-frequency 3D DCT coefficients. In the case of local transforms, the DCT coefficient dimensionality is excessive ( $B \times 280$ ), hence must be reduced. We organized these vectors into a data matrix,  $Q$  where the number of rows is  $B \times 280$  and where the number of columns is equal to the set of training videos. We obtain feature vectors for each emotion video by using ICA [109] algorithm.

## 6. CLASSIFIERS FOR FACIAL EXPRESSION AND HEAD GESTURE RECOGNITION

A set of discriminative and effective features should be selected from the extracted features to construct the FHG classifier. It is known that the motion of certain landmarks are more expressive and hence contain more discriminative information, and this selection depends on the face and head gesture types. Therefore it would pay to pinpoint these more discriminative and effective features per gesture.

### 6.1. Feature - Sequence Classifiers

We denote classifiers that explicitly take into consideration the state sequence information in the time series as sequence classifiers. HMM is the prototypical sequence classifier. Recently HCRF, introduced by Quattoni et al. [125], has been proposed as an alternative in labeling sequential data such as in speech processing, gesture recognition and in bioinformatics [101, 125, 126, 127, 128].

For gesture classification, we have the HMM or HCRF models learn their parameters separately during training for each gesture class; and during testing the highest scoring gesture class is declared as the recognized gesture. The numbers of states are selected heuristically by considering the complexity of gesture classes. The inputs to HMM or HCRF model are same and consist of feature sequences, as illustrated in Figure ??.

#### 6.1.1. HMM

HMMs are generative models, based on a directed graph, which defines a joint probability distribution  $p(x, s)$ , where  $X$  and  $S$  random variables respectively range over observation sequences and their corresponding states. HMMs can model spatiotemporal information in a natural way. Graphical structure for a simple HMM is given in Figure 6.1.

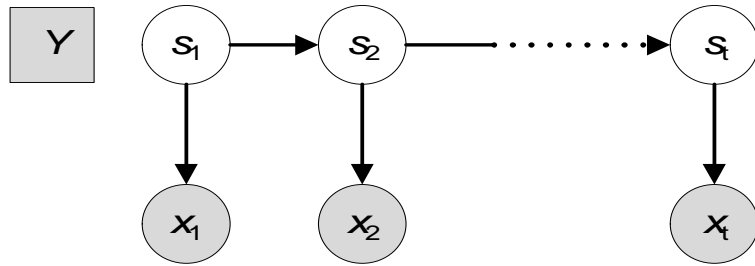


Figure 6.1. Graphical structure of a simple HMM ( $Y$  is the output label,  $s$  is the state label and  $X$  is the observation sequence )

In this thesis, the observed quantities and therefore the feature vectors are continuous, so we need to run FHG recognition algorithm using continuous HMMs with mixture of Gaussian outputs. The number of hidden states and of the mixture of Gaussians are chosen in accordance with the complexity of the action classes. For example for the head nodding with smiling class five hidden states and six mixtures of Gaussians were used while for sadness and for smiling classes only three states and five mixture of Gaussians were used. We have used Murphy's HMM toolbox [129] to evaluate the classification performance of multivariate continuous density HMMs for head and face gesture videos.

### 6.1.2. HCRF

HCRF is an undirected graph model for classification based on Conditional Random Field (CRF)[127, 128, 125, 130] augmented with latent states. CRF is a probabilistic framework for labeling and segmenting sequential data, based on the conditional approach. It models a conditional probability  $p(y|x)$  over label sequences given a particular observation sequence  $X$ , rather than a joint distribution over both label and observation sequences as in HMM.

HCRF models the distribution  $p(y, s|x)$  directly, where  $Y$  is class label and  $S$  is an intermediate hidden state modeled as a Markov random field globally conditioned on observations  $X$ . These hidden states model the latent structure of the input domain, thus it naturally suits the task of gesture categorization. Fig. 6.2 illustrates the graphical structure of an HCRF model.

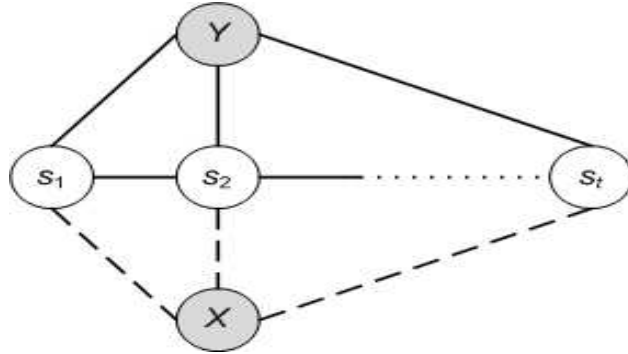


Figure 6.2. Graphical structure for HCRF ( $Y$  is output label,  $s$  is the hidden state labels and  $X$  is the observation sequence)

Different from HMM, HCRF can incorporate long range dependencies of observations by modifying the potential function via a window parameter that defines the amount of past and future history to be used when predicting the state at time  $t$ . In addition, discriminative learning of the latent structure using HCRF is an advantage against the generative learning approach using HMM [131, 130].

We have used HCRF library [132] for HCRF-based gesture classifications. For HCRF based gesture sequence classification the following steps are carried out for the training and test parts:

- (i) First, for each gesture class we trained a separate HCRF classifier to discriminate each against all other gesture classes.
- (ii) The numbers of hidden states are selected in accordance with the complexity of gesture classes. Notice that the same number of hidden states are used for both HCRF models and HMM models.
- (iii) For a given test sequence, we run it through each HCRF gesture classifier, and the highest scoring model is declared as the recognized gesture.

HMM and HCRF methods do not require a-priori segmentation information of underlying substructures, e.g. label of each frame, because of its hidden state structure.

## 6.2. Feature - Subspace Classifiers

While HMM/HRCF variety processes information sequentially, subspace methods can look at the panorama of the sequence of events in one go. Thus we consider the video shot of a whole gesture in its entirety and express it in terms of the spatiotemporal data matrix  $D$ . The  $m \times T$  matrix  $D$  incorporates the spatiotemporal data of the gesture, where  $m$  is the length of feature vector (e.g., 34 for trajectory data and 17 for geometric features) and  $T$ , the number of frames in the gesture. Since the duration  $T$  of gestures are variable depending upon the gesture type and, for a specific gesture, upon the actor, we normalized the number of frames ( $T$ ) for an FHG shot, by using the “resample” function of the Matlab so that all gesture spatiotemporal shots had length  $n$  (number of frames). Note that “resample” function basically changes the sampling rate of a given sequence to a desired one using a polyphase implementation. The resulting spatiotemporal data matrix  $D$  has columns corresponding to spatial features and rows corresponding to normalized time index. Figure 6.3 illustrates comparatively the original feature sequence (green line), the linearly stretched, that is, normalized sequence (dashed blue), and the hidden state sequence in the HMM (turquoise line with dots). Hidden states can be regarded as feature sequence after HMM based normalization. HMM model assigns observed features into a hidden state. Note that, in this example 3 hidden states represent the original feature sequence.

The temporal domain normalization process, explained in the previous paragraph, is applied to the trajectory matrix  $P$ , geometric feature matrix  $G$  and DCT-based appearance matrix  $A$ , respectively. The detailed descriptions of these spatiotemporal matrices can be found in Chapter 5.

The resulting spatiotemporal matrices  $P$ ,  $A$  and  $G$  can be regarded as a 2D intensity image, where columns correspond to spatial features and the rows correspond to the temporal changes of spatial features during a gesture. In our study we choose  $n$  as 60, which is also the average length of the gesture sequences. In Fig. 6.4, the histogram of the gesture video durations is plotted. Notice that at 30 frames per second, the average gesture duration is 2 seconds (60 frames) and their standard deviation was

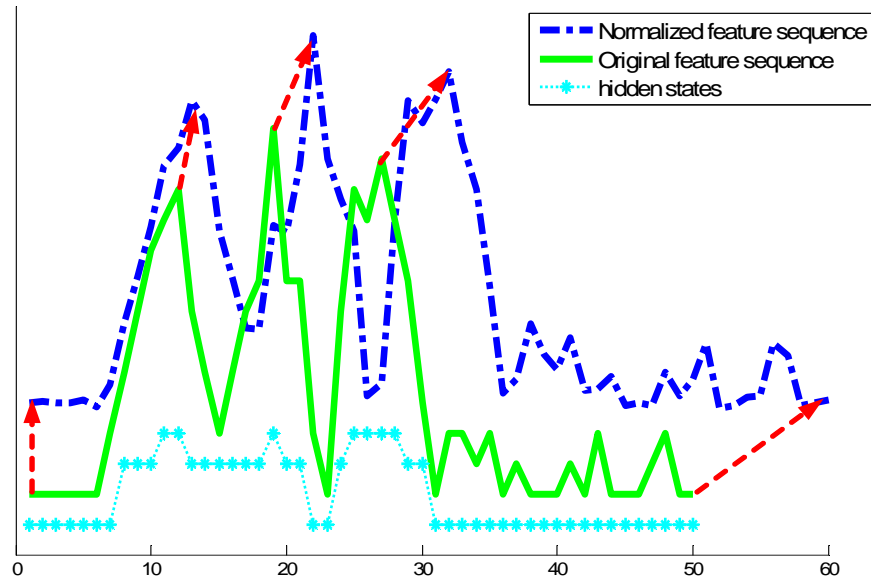


Figure 6.3. Comparison of time normalized feature sequence and hidden state sequence after HMM

estimated to be 0.35 seconds or 11 frames.

Once gesture videos are reduced to a fixed size ( $m \times n$ ) matrix, called the gesture matrix, we treated these spatiotemporal matrices as intensity images. This allows for various subspace projection methods to be utilized for feature extraction and classification. In Figure 6.5, spatiotemporal trajectory matrices ( $P$ ) for different face and head gesture categories are illustrated. One can notice that spatiotemporal trajectory matrices corresponding to each gesture category has a different pattern. For example, Head L-R gesture has a smooth waviness in the upper half of the image which corresponds to the temporal changes in the x-coordinates of the landmarks. We can also say that, it may be possible to discriminate the gesture classes especially accompanying with head rotations by looking over the visualized spatiotemporal trajectory matrices.

The subspace feature alternatives are:

- (i) DCT features: The  $m \times n$  gesture matrix (image) is subjected to DCT transform as if it were an image. The first few coefficients from the zigzag order are selected

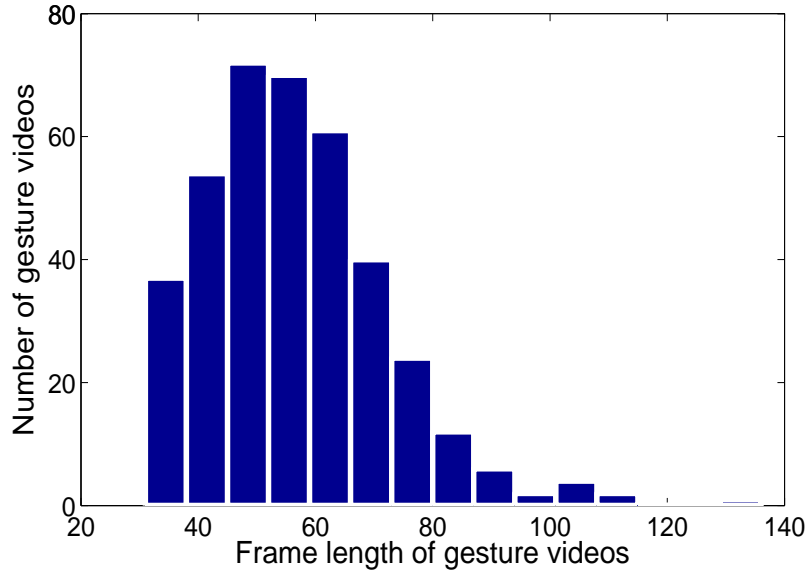


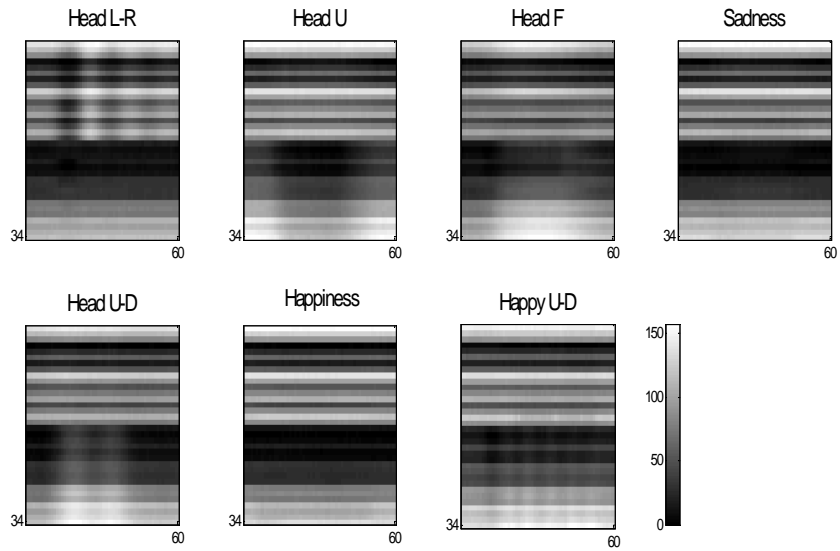
Figure 6.4. Histogram of the length of the gesture videos in the database

and ordered as a gesture feature vector.

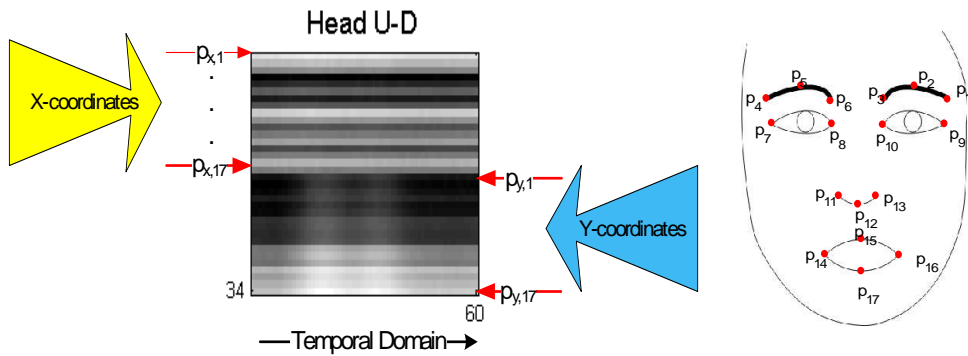
- (ii) ICA features: We perform the ICA [109] of each gesture class, and extract the respective ICA basis vectors and their mixing coefficients. We vectorize each gesture matrix, (e.g,  $P$ ,  $G$  or  $A$ ), to form a vector  $q$  of length  $m \times n$  (e.g.,  $34 \times 60 = 2040$  for  $P$ ) by lexicographic ordering of the matrix elements. The resulting  $Q = [q_1 q_2 \dots q_r]$  matrix, representing the training data, is then of size  $(m \times n) \times r$ , where  $r$  is the number of training gestures. Thus we obtain:

$$Q = M.F \quad (6.1)$$

where the columns of  $M$  (size  $(m \times n) \times u$  where  $u < (m \times n)$ ) are the mixing coefficients for a gesture type and the columns of  $F$ , size  $u \times r$ , are independent source signals. Notice that, there are separate  $Q$ ,  $M$  and  $F$  matrices for each  $L$  gestures, i.e.  $Q_{(i)}$ ,  $M_{(i)}$  and  $F_{(i)}$  for  $i = 1, \dots, L$ . The test vector  $q$  is subjected ICA analysis via  $f_{(i)} = M_{(i)}^+ q$  where  $f_{(i)}$  is the projection of the test vector upon the  $i^{\text{th}}$  gesture subspace  $M_{(i)}$ . Finally decision is made based on MNN classifier which will be described in Section 6.2.1. When the observed signals are trajectory matrix  $P$  ( $Q$ ) then the dimension of the independent source signals is reduced from  $m \times n$  to  $u$ . When  $u$  is selected as 25 then the best classification performance is



(a)



(b)

Figure 6.5. (a) Spatiotemporal trajectory matrices  $P$  as gray level intensity images.  
 (b) The upper half corresponds to x-coordinates of the landmarks and lower half corresponds to y-coordinates of the landmarks.

Table 6.1. Dimensions of the subspace projection matrices for BUHMAP database ( $r$  is the number of training samples)

Data matrix $Q$	ICA( $M$ & $F$ )	NMF ( $W$ & $H$ )	DCT coefficients
Trajectory matrix $P$ (34x60) $\times r$	M: 2040x25 F: 25 $\times r$	W: 2040x(34x8) H: (34x8) $\times r$	(34x7) $\times r$
Geometric features $G$ (17x60) $\times r$	— —	W: 1020x(17x10) H: (17x10) $\times r$	— —
Appearance matrix $A$ (75x60) $\times r$	M: 4500x70 F: 70 $\times r$	W: 4500x50 H: 50 $\times r$	— —

achieved for the trajectory matrix  $P$  extracted from the BUHMAP [9] database.

- (iii) NMF features: Similar to ICA, NMF [133] aims to factorize a matrix into basis vectors and their combiner coefficients. Using a training data set,  $Q$  with size  $(m \times n) \times r$ , the  $k$  basis vectors, columns of  $W$ , are obtained as:

$$\mathbf{Q} \approx \mathbf{W} \cdot \mathbf{H}, \quad (6.2)$$

In the factorization in Eq. 6.2, the columns of the  $(m \times n) \times k$  matrix  $W$  stand for the basis vectors and the columns of the  $k \times r$  matrix  $H$  determine how the basis vectors are activated to reconstruct the image  $Q$ . The columns of  $H$  represent the NMF-based feature vectors of the corresponding gesture data. The classification of a test gesture is based on its NMF feature given by  $h = W^+q$ . The number of columns  $k$  in the (basis) matrix  $W$  was heuristically determined to be chosen to be 272 (34x8) if matrix  $P$  extracted from BUHMAP database is decomposed. This represented a substantial simplification in the representation of the data via basis vectors ( $272 \ll 2040$ ). The heuristic search for subspace dimension  $k$  used multiples of the feature vector size 34. In this study, MATLAB code of NMF based on the projected gradient method is utilized [134].

Table 6.1 summarizes the dimensions of the subspace matrices for the corresponding feature types. Note that these dimensions are computed for the BUHMAP [9] database.

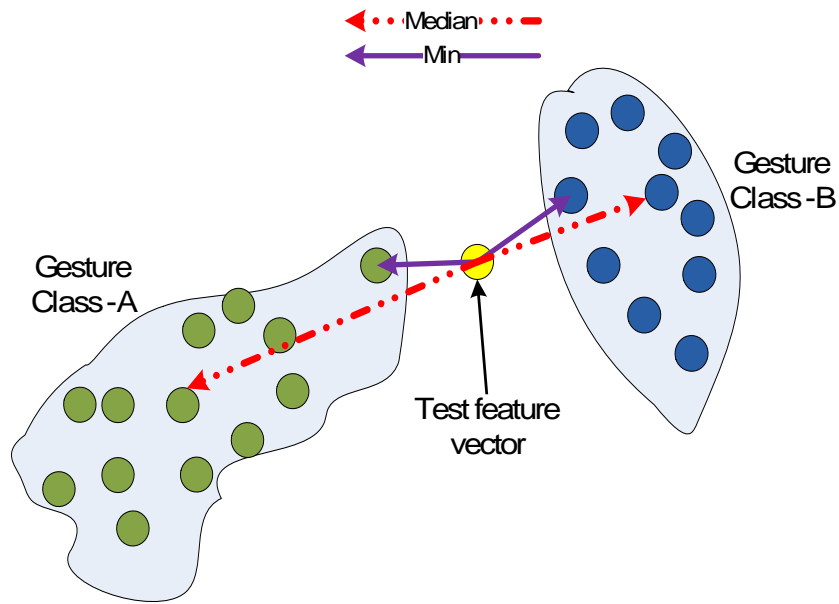


Figure 6.6. Min and median distances of a test feature vector from 2 different gesture classes.

Once the feature vectors (DCT, ICA, NMF or else) are obtained, various distance metrics can be used to find the similarity of the test input and the trained gesture classes, such as L1, L2 norms, normalized correlation or cosine and Mahalanobis distances and so on. The choice of the distance metric is important as it may sometimes affect the classification performance significantly. We found that cosine distance was better to measure the similarity of the ICA and NMF subspace features, and that Spearman correlation [135] worked better for DCT features of BUHMAP database [9].

### 6.2.1. Modified Nearest - Neighbor Classification

The gestures in the feature subspace method are classified by a MNN classifier. When a test gesture arrives, its feature vector is calculated by projecting the gesture data onto the sub-space of each gesture class. Then its distance from projected training feature vectors for each gesture classes is computed and stored in a distance vector. We observed that summing the minimum distance and median distance between the test vector and training vectors of a given gesture class performs better than NN classifier (see Figure 6.6), hence we call this classifier as MNN. Finally, the test gesture is assigned the label of the minimum distance class. In Figure 6.7, we denote the projection

Figure 6.7. Assigning a class label to a test sample

```

for  $j = 1$  to  $L$  (# of gestures) do
   $f_{test} = Pro_j * q$ 
  {Test measurement  $q$  projected onto the  $j^{th}$  gesture class subspace to yield feature
  vector  $f_{test}$ }
  for  $k = 1$  to  $r$  (# of training gesture in class  $j$ ) do
     $distance(k) = Dist(f_{test}, f_{train}^{k,j})$ 
    {the distance between test sample and all subspace training samples in class  $j$ }
  end for
   $Dist_{tot}(j) = min(distance) + median(distance)$  (MNN)
end for
Class Label =  $\underset{j}{\operatorname{argmin}}(Dist_{tot}(j))$ 

```

Table 6.2. Comparison of NN, LDA and MNN methods on NMF features of trajectory matrix  $P$ .

LDA	NN	MNN
76.1	81	86

operation onto the class subspace as  $Pro_j$ , where  $Pro$  can represent DCT, ICA, NMF or other subspace operation.

In order to give an idea about the classification performances, NN, LDA and MNN classifiers are compared with NMF features of trajectory matrix  $P$ . The average classification accuracies over 7 gesture classes are summarized in Table 6.2. It is observed that MNN classifier outperforms the average classification accuracies of the alternative classification methods. Note that in order to make a fair comparison among LDA, NN and MNN classifiers, Euclidean distance metric is used for the computations.

### 6.3. Decision level fusion of classifiers

Classifier combination is a critical strategy to empower the recognition accuracy of many application areas from character recognition to face recognition [136, 137, 138].

There are three levels of fusion schemes [137] based on the individual classifiers output information as follows:

- (i) Abstract-level: This method is applicable for fusion of classifiers that produce only class label as the output.
  - Plurality voting (PV): Plurality voting is the most commonly used one, which just outputs the class label having the highest vote among the classifiers.
- (ii) Rank-level: The combination is made based on the output information of the rank level. All labels are ranked in a list with the label at the top being the first choice.
  - Plurality voting (PV)
  - Borda Count (BC): The Borda scheme calculates the combined ranking by summing the class ranks as assigned by the individual classifiers. The class having the highest vote among top  $n$  rank is assigned as the winner.
  - Weighted Borda Count (WBC): It is obtained by assigning weights to the ranks produced by individual classifiers. Again, the class with highest vote is assigned as the winner. For example, class label with top rank will be assigned by higher weights.
- (iii) Similarity score level: Each classifier produces a similarity score for each class proportional with its likelihood score. The class label with highest similarity score is assigned as the winner. These scores can be combined by using simple arithmetic rules such as follows:
  - MIN: For each class label, minimum of the scores are selected over the classifiers.
  - MAX: For each class label, maximum of the scores are selected over the classifiers.
  - MEDIAN: For each class label, median of the scores are selected over the classifiers.
  - SUM: For each class label, sum of the scores are selected over the classifiers.
  - PROD: For each class label, product of the scores are selected over the classifiers.

- Weighted sum (WSUM): For each class label, weighted sum of the scores are selected over the classifiers. Weights are assigned to the ranks produced by individual classifiers. Class labels with higher ranks will be assigned by higher weights.

Note that, similarity score level contains the highest amount of information and the abstract level contains the lowest. From the similarity score attributed to each label, we could rank all the labels according to a rank rule. By choosing the label at the top rank, or directly by choosing the label with the maximum value at the similarity score level, then we can assign a label to a test data.

Since we considered more than one feature set and more than one classifier, we experimented with fusion schemes given in the above list. We found them quite advantageous. We observed that decision level fusion improves the classification performance significantly. Similarity scores of the classifiers are normalized before fusion. Since the similarity score ranges of the classifiers can change depending on the classifier type and distance measure used. Figure 6.8, illustrates the normalization of the scores for two sample classifiers. Note that each classifier gives a similarity score for each gesture category. We applied unit-norm normalization to the similarity scores of the classifiers. In unit norm normalization method, a given similarity score vector is divided into the length of the score vector. Then the length of the similarity score vector becomes equal to 1. The length of the vector is the square root of the sum of squares of all values, which is also called as L-2 norm of a vector.

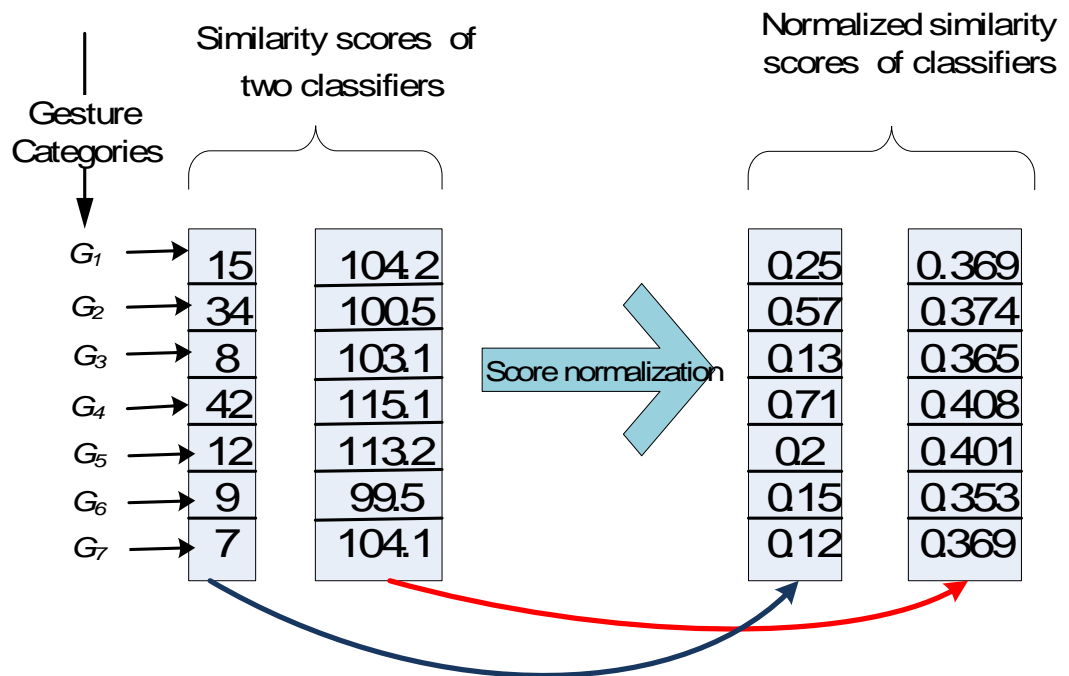


Figure 6.8. Score normalization process for two sample classifier scores

## 7. RECOGNITION RESULTS FOR HEAD GESTURE AND FACIAL EXPRESSION VIDEOS

We evaluated our FHG recognition algorithm on the BUHMAP video database [9] (<http://www.cmpe.boun.edu.tr/pilab/pilabfiles/databases/buhmap/>), LILiR TwoTalk corpus [10, 11] and Cohn-Kanade facial expression database [71].

BUHMAP includes seven non-manual gesture classes (except neutral states) selected from TSL. The videos are recorded at 30 fps at the resolution of 640x480. Each video starts and ends in the neutral state of the face. This data set includes seven gesture types acted by eleven subjects, with five repetitions each, hence overall 385 video shots. Each video lasts about 1-2 seconds. Notice that, each video starts in neutral state, the sign is performed and again ends in neutral state. No subjects have beard, moustache or eyeglasses. There is no occlusion or motion blur. 48 of the videos (3 repetitions of 4 classes such as Head L-R, Head Up, Head F, Happiness, performed by 4 subjects ) are annotated as ground truth data. The videos represents some of the non-verbal messages which are frequently used in TSL (Table 7.1). The details of the gesture classes were also given in Table 4.2.

The LILiR TwoTalk corpus [10] is a collection of four conversations (dyadic) involving 2 persons engaged in casual conversation for 12 minutes each. 527 short clips were extracted and annotated by multiple observers to investigate non-verbal communication in conversations. The data set and annotations are publicly available ([http://www.ee.surrey.ac.uk/Projects/LILiR/twotalk\\_corpus](http://www.ee.surrey.ac.uk/Projects/LILiR/twotalk_corpus)) [10]. Annotation of non-verbal messages (or mental states) was performed by multiple annotators coming from various cultures on 527 clips, extracted from the longer videos. The conversation participants were only instructed to be seated and to talk. The topic of conversation was not constrained. All videos have tracking data supplied (Ong and Bowden [39, 139]). The annotation focused on the non-verbal communication categories as summarized in

Table 7.1. Meaning of the non-manual signs recorded in BUHMAP videos for TSL

Head shaking ( <b>G1</b> ):	Mostly indicates disagreement.
Head up ( <b>G2</b> ):	Corresponds to disapproving and disinclined.
Head forward ( <b>G3</b> ):	Used to change the sentence into a question form. It also represents being curious, interested and asking mental states in daily life.
Sadness ( <b>G4</b> ):	Indicates sadness, e.g. when apologizing or feeling guilty. Some subjects also move their head downwards.
Head up-down ( <b>G5</b> ):	Frequently represents agreement and approval states.
Happiness ( <b>G6</b> ):	Indicates joy.
Happy up-down ( <b>G7</b> ):	Indicates approval and joy.

Table 7.2. Nonverbal communication categories of LILir TwoTalk Corpus [10]

Question category	Minimum Score	Maximum Score
Does this person disagree or agree with what is being said?	Strong disagreement	Strong agreement
Is this person thinking hard?	No indication	In deep thought
Is this person asking a question?	No indication	Definitely asking question
Is this person indicating they understand what is being said to them?	No indication or N/A	Strongly indicating understanding

Table 7.2.

The main difference between BUHMAP [9] and LILiR TwoTalk corpus [10] is that LILiR TwoTalk corpus is more realistic, since the videos were recorded during a spontaneous conversation of two persons. There is a growing interest on the analysis of facial dynamics and naturally occurring (spontaneously displayed) facial behaviour [140, 94, 141]. Therefore, we performed our FHG classifier on LILiR TwoTalk corpus to evaluate its classification performance on more realistic datasets.

In addition to BUHMAP [9] and LILiR TwoTalk corpus [10] experiments, we evaluated alternative data representation form which is called 3D face cube (Section 5.3) based facial expression recognition method on the well known and most widely used Cohn-Kanade database [71]. The database consists of 100 university students ranging in age from 18 to 30 years who enacted among many others, the six prototypical emotions, i.e., anger, disgust, fear, joy, sadness, and surprise. Image sequences showing the evolution of the emotion from neutral state to target level (apex phase) were digitized into 640 x 480 pixel arrays with 8-bit precision for gray scale values. The video rate is 30 fps.

The relevant part of the database for our work has 322 image sequences involving 92 subjects portraying the six basic emotions. The distribution of videos into the 6 emotion classes is not uniform. To guarantee that the reported results are practically person independent, first, we partitioned the dataset into ten groups of roughly equal number of subjects and sequences. Then, we applied ten fold cross validation testing where each fold included novel subjects.

### 7.1. Performance Results for BUHMAP Database

The organization of the test set and of the experiments is given in Table 7.3. An 11-fold cross-validation scheme is carried out for training and testing any one feature set and classifier combination. For each fold, one subject's gesture samples (7x5=35 gesture samples) are left out as test set and the 350 gesture samples of the remaining

Table 7.3. Test set and performed experiments

Test	Subjects (S)	Class (C)	Repetitions (R)
	11	7 ( <b>G1</b> , <b>G2</b> , <b>G3</b> , <b>G4</b> , <b>G5</b> , <b>G6</b> , <b>G7</b> )	5
Experiment	Training	Testing	Method
11 fold	10 S, 5 R, 350 videos	1 S, 5 R, 35 videos	Leave-one-S-out cross validation

subjects are used for training. Thus for each fold, each gesture class has 5 test samples and 50 positive training samples.

Classification results with different combinations of features and classifiers are given in a series of tables. In all tables (Table 7.4, 7.5, 7.6, 7.7, 7.8, 7.9, 7.10, 7.11, 7.12, 7.13, 7.14, 7.15, 7.17, 7.18), we use the following FHG acronyms: **G1**: head shake, **G2**: head up, **G3**: head forward, **G4**: sadness **G5**: nodding, **G6**: smile, **G7**: nodding +smile. The numbers in parentheses is a reminder of the feature vector size. We have itemized the main observations below:

- (i) HMM versus HCRF: We compared HMM classifier with the HCRF classifier for various choices of feature types. HCRF outperforms HMM classifier, as seen in Tables 7.4, 7.5, 7.6 and 7.7. In fact, for landmark trajectory features HCRF is more than 15.3 percentage points ahead while for geometric features the advantage is only 1.5 points when averaged over all gestures.
- (ii) Choice of feature types for sequential classifiers: We compared the three types of features, namely landmark coordinates, geometric features and DCT-based appearance features. The geometric features yielded the best performance followed by landmark trajectory features. DCT-based appearance, not surprisingly, performed poorly. These results follow from comparisons of table pairs, 7.4 & 7.6, 7.5 & 7.7. The scores of DCT appearance features are given in Tables 7.8 and 7.9.
- (iii) Sequential classifier versus subspace classifier: Recall that two classification paradigms were considered, namely HMM or HCRF variety of sequential classifiers and

Table 7.4. HMM with landmark trajectory features (34, $P$ )

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>100</b>	0	0	0	0	0	0
<b>G2</b>	3.6	<b>81.8</b>	1.8	0	5.5	0	7.3
<b>G3</b>	5.5	10.9	<b>80</b>	0	1.8	0	1.8
<b>G4</b>	29.1	5.5	18.1	<b>21.9</b>	9.1	0	16.3
<b>G5</b>	9.1	9.1	0	0	<b>63.7</b>	0	18.1
<b>G6</b>	20	0	0	0	0	<b>27.3</b>	52.7
<b>G7</b>	9.1	1.8	5.5	0	5.5	0	<b>78.1</b>
Average Recognition Accuracy (ARA): <b>64.7%</b>							

Table 7.5. HCRF with with landmark trajectory features (34, $P$ )

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>100</b>	0	0	0	0	0	0
<b>G2</b>	0	<b>90.1</b>	3.6	1.8	1.8	0	1.8
<b>G3</b>	9.1	1.8	<b>81.8</b>	0	5.5	0	1.8
<b>G4</b>	7.3	1.8	7.3	<b>76.4</b>	1.8	0	5.5
<b>G5</b>	1.8	5.5	3.6	0	<b>81.8</b>	0	7.2
<b>G6</b>	3.6	0	0	5.5	1.8	<b>69.1</b>	20
<b>G7</b>	5.5	1.8	3.6	0	25.5	3.6	<b>60</b>
Average Recognition Accuracy (ARA): <b>80%</b>							

Table 7.6. HMM with geometric features (17, $G$ )

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>100</b>	0	0	0	0	0	0
<b>G2</b>	0	<b>96.4</b>	3.6	0	0	0	0
<b>G3</b>	0	1.8	<b>92.8</b>	1.8	1.8	0	1.8
<b>G4</b>	5.5	5.5	7.2	<b>60</b>	5.5	15	1.8
<b>G5</b>	0	5.5	36.3	0	<b>75</b>	0	16.3
<b>G6</b>	0	1.8	0	3.6	0	<b>92.8</b>	1.8
<b>G7</b>	1.8	1.8	3.6	0	11	3.6	<b>78</b>
Average Recognition Accuracy (ARA): <b>84.9</b>							

Table 7.7. HCRF with geometric features (17, $G$ ),  $w=1$ 

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>96.4</b>	0	0	1.8	0	0	1.8
<b>G2</b>	0	<b>92.7</b>	1.8	0	0	5.5	0
<b>G3</b>	0	0	<b>94.5</b>	5.5	0	0	0
<b>G4</b>	1.8	1.8	5.5	<b>72.7</b>	0	12.7	5.5
<b>G5</b>	0	1.8	1.8	5.5	<b>89.1</b>	0	1.8
<b>G6</b>	1.8	1.8	0	7.2	0	<b>85.5</b>	3.6
<b>G7</b>	0	1.8	3.6	3.6	7.3	9.1	<b>74.6</b>
Average Recognition Accuracy (ARA): <b>86.5</b>							

Table 7.8. HMM with DCT-based appearance features (75 block DCT coefficients,  $A$ )

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>78.2</b>	1.8	5.5	5.5	0	3.5	5.5
<b>G2</b>	1.8	<b>83.7</b>	14.5	0	0	0	0
<b>G3</b>	1.8	30.1	<b>27.7</b>	16.3	10.9	7.2	5.5
<b>G4</b>	0	10.9	7.2	<b>49.2</b>	9.1	20	3.6
<b>G5</b>	9.1	20	9.1	12.7	<b>25.6</b>	7.2	16.3
<b>G6</b>	3.6	7.2	0	20	11	<b>51</b>	7.2
<b>G7</b>	5.5	9.1	3.6	18.1	7.2	38.3	<b>18.2</b>
Average Recognition Accuracy (ARA): <b>47.5</b>							

Table 7.9. HCRF with DCT-based appearance features (75 block DCT coefficients,  $A$ )

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>61.8</b>	0	5.5	30.9	0	0	1.8
<b>G2</b>	0	<b>67.3</b>	9.1	9.1	0	3.6	10.9
<b>G3</b>	0	16.4	<b>52.7</b>	9.1	5.5	3.6	12.7
<b>G4</b>	14.5	0	9.1	<b>54.5</b>	10.9	1.8	9.1
<b>G5</b>	1.8	1.8	12.7	23.6	<b>58.2</b>	0	1.8
<b>G6</b>	0	1.8	1.8	1.8	1.8	<b>27.3</b>	65.5
<b>G7</b>	1.8	0	1.8	1.8	0	18.2	<b>76.4</b>
Average Recognition Accuracy (ARA): <b>56.9</b>							

Table 7.10. MNN results with DCT features from landmark trajectory matrix  $P$ 

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>85.5</b>	10.9	1.8	1.8	0	0	0
<b>G2</b>	0	<b>98.2</b>	1.8	0	0	0	0
<b>G3</b>	0	0	<b>100</b>	0	0	0	0
<b>G4</b>	7.2	3.6	5.5	<b>65.6</b>	1.8	7.2	9.1
<b>G5</b>	3.6	5.5	36.3	9.1	<b>36.3</b>	0	9.1
<b>G6</b>	0	10.9	0	0	0	<b>85.5</b>	3.6
<b>G7</b>	1.8	1.8	5.5	0	1.8	29.1	<b>60</b>
Average Recognition Accuracy (ARA): <b>75.8</b>							

subspace feature classifiers using the matrix organization of the spatiotemporal data. Tables 7.4-7.9 belong to sequential classifiers while Tables 7.10-7.13 give the results of the subspace methods. The first conclusion is that the performance of the best subspace classifiers is even better than that of best sequential classifier. NMF features extracted from landmark coordinate features with nearest class classifier achieves 87.3% (Table 7.12) while HCRF method on the sequence of geometric features achieves 86.5% (Table 7.7).

- (iv) Choice of feature types for subspace classifiers: Comparing the average recognition accuracy (ARA) scores in Tables 7.10, 7.11 and 7.12, one can see that NMF features extracted from landmark coordinate trajectory matrix has the highest performance. This is closely followed by the ICA decomposition. The higher performance of the subspace methods can come as a surprise because HMM or HCRF methods are designed to handle time series events that evolve at varying rates from realization to realization. One possible explanation is that the rates of evolution of the gesture acting under the controlled conditions do not vary much. Fig. 6.4 shows the histogram of durations in frame number from neutral state to peak expression and back to neutral still. Second, the linear time warping has mitigated some of the duration variability.

We utilized eight of the designed classifiers as summarized in Table 7.16 to take roles in various fusion schemes as follows.

Table 7.11. MNN results with ICA features from landmark trajectory matrix  $P$ 

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>98.2</b>	1.8	0	0	0	0	0
<b>G2</b>	0	<b>98.2</b>	1.8	0	0	0	0
<b>G3</b>	3.6	9.1	<b>85.5</b>	1.8	0	0	0
<b>G4</b>	7.2	0	1.8	<b>52.8</b>	16.3	18.2	3.6
<b>G5</b>	0	0	3.6	5.5	<b>76.4</b>	0	14.5
<b>G6</b>	0	0	0	0	1.8	<b>94.6</b>	3.6
<b>G7</b>	0	0	1.8	0	12.7	0	<b>85.5</b>
Average Recognition Accuracy (ARA): <b>84.4</b>							

Table 7.12. MNN results with NMF features from landmark trajectory matrix  $P$ 

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>100</b>	0	0	0	0	0	0
<b>G2</b>	0	<b>96.4</b>	1.8	0	0	0	1.8
<b>G3</b>	0	0	<b>72.8</b>	7.2	7.2	0	12.8
<b>G4</b>	0	0	0	<b>92.8</b>	1.8	1.8	3.6
<b>G5</b>	0	0	0	0	<b>78.2</b>	0	21.8
<b>G6</b>	0	0	0	7.3	1.8	<b>81.8</b>	9.1
<b>G7</b>	0	0	0	0	10.9	0	<b>89.1</b>
Average Recognition Accuracy (ARA): <b>87.3</b>							

Table 7.13. MNN results with NMF features from geometric feature matrix  $G$ 

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>98.2</b>	0	0	1.8	0	0	0
<b>G2</b>	0	<b>76.4</b>	3.6	5.5	7.2	5.5	1.8
<b>G3</b>	0	0	<b>92.7</b>	5.5	1.8	0	0
<b>G4</b>	0	0	5.5	<b>58.2</b>	14.5	14.5	7.3
<b>G5</b>	0	0	0	1.8	<b>92.7</b>	0	5.5
<b>G6</b>	0	0	0	14.5	1.8	<b>78.2</b>	5.5
<b>G7</b>	0	0	0	1.8	14.5	0	<b>83.6</b>
Average Recognition Accuracy (ARA): <b>82.9</b>							

Table 7.14. MNN results with NMF features from DCT-based appearance matrix  $A$ 

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>80</b>	0	5.45	3.64	10.9	0	0
<b>G2</b>	1.82	<b>89.09</b>	3.64	1.82	1.82	1.82	0
<b>G3</b>	0	5.45	<b>72.73</b>	12.73	7.27	1.82	0
<b>G4</b>	0	1.82	1.82	<b>70.9</b>	14.55	10.9	0
<b>G5</b>	1.82	0	3.64	29.09	<b>65.45</b>	0	0
<b>G6</b>	0	0	0	9.09	5.45	<b>69.09</b>	16.36
<b>G7</b>	0	0	1.82	14.55	3.64	30.9	<b>49.09</b>
Average Recognition Accuracy (ARA): <b>70.9</b>							

Table 7.15. MNN results with ICA features from DCT-based appearance matrix  $A$ 

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>74.55</b>	0	3.64	10.9	9.09	1.82	0
<b>G2</b>	0	<b>89.09</b>	0	5.45	3.64	0	1.82
<b>G3</b>	0	3.64	<b>72.73</b>	12.73	3.64	5.45	1.82
<b>G4</b>	5.45	1.82	7.27	<b>63.64</b>	7.27	12.73	1.82
<b>G5</b>	9.09	3.64	7.27	20	<b>54.55</b>	3.64	1.82
<b>G6</b>	0	0	1.82	21.82	0	<b>54.55</b>	21.82
<b>G7</b>	0	0	0	5.45	0	20	<b>74.55</b>
Average Recognition Accuracy (ARA): <b>69.1</b>							

Table 7.16. Designed classifiers

---

C1	HMM with 17 sequential geometrical features ( $G$ )
C2	MNN with 272 NMF coefficients of landmark coordinates trajectories ( $P$ )
C3	HCRF with 17 sequential geometrical features ( $G$ )
C4	MNN with 25 ICA coefficients of landmark coordinates trajectories ( $P$ )
C5	HMM with 75 sequential patch DCT coefficients ( $A$ )
C6	MNN with 238 DCT coefficients of landmark coordinates trajectories ( $P$ )
C7	HCRF with 75 sequential patch DCT coefficients ( $A$ )
C8	HCRF with 34 sequential landmarks coordinates trajectories ( $P$ )

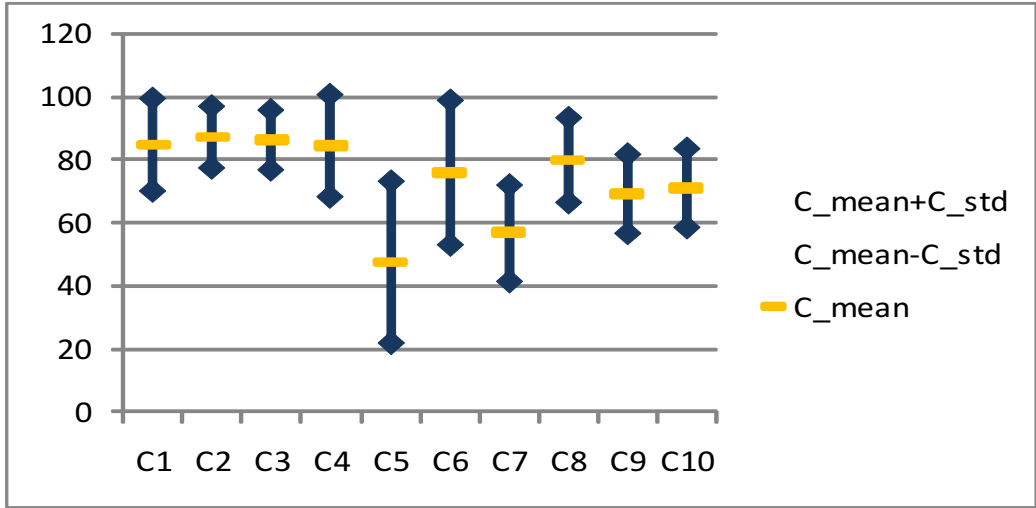
---

Figure 7.1 illustrates the performance statistics for the classification of gestures and the corresponding classifiers. In Figure 7.1-a, it is observed that classifier C2 and classifier C3 has similar classification performances over gestures. But classifier C3 (HCRF with geometrical features) has the smallest standard deviation over the classifiers whereas C2 is the best individual classifier with %87.3 classification accuracy. Here all classifiers surpass the classification performance of classifier C5 (HMM with patch DCT coefficients). Figure 7.1-b demonstrates the mean and standard deviations of classification rates of each gesture computed over 8 classifiers. It is observed that **G2** (head-up and eyebrow raise) is the best classified gesture category with smallest standard deviation over 8 classifiers. **G1** (head shaking) has a similar classification performance to **G2**. It is also observed that sadness (**G4**) and smile (**G6**) expression pair and Head U-D (**G5**) and Happy U-D (**G7**) gesture pair are the most misclassified gesture pairs.

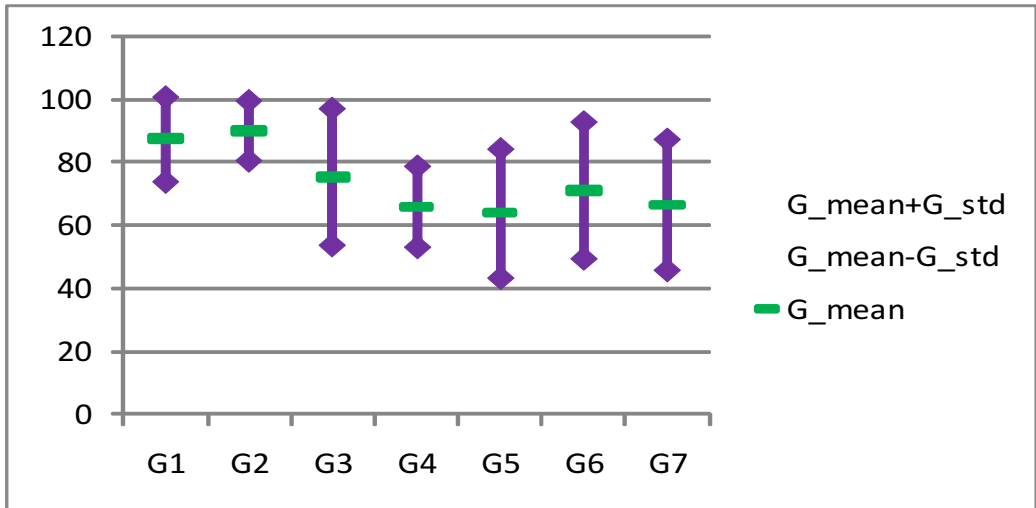
When we watch the videos of the gestures we observed that in most cases Happy U-D action (**G7**) is accompanied with a subtle smile different then smile alone gesture class (**G6**). Sadness is another difficult expression for classification, since acting of the sadness state by the subjects in the database differs significantly. For example, some of the subjects activated “raise inner eyebrow points” and “lip corners down” actions, while others activated “eyebrow fall” and “lip pucker” actions for sadness emotion.

### 7.1.1. Results of Classifier Fusion

When we analyze detection scores in Tables 7.4 to 7.12, we observe that the performances of gesture types vary significantly depending upon the feature and classifier combination. In particular, the overall scores (ARA) and the individual **G1** to **G7** performances are not proportional, that is an improvement in ARA does not mean a uniform improvement in all classes. Thus, a feature and classifier combination that yields lower score can have some gestures detected more accurately as compared to a higher ARA feature and classifier combination. This has given us a hint as for the potential use of classifier fusion.



(a)



(b)

Figure 7.1. (a) Mean and standard deviation of classifiers over 7 gestures , (b) Mean and standard deviation of gestures over 8 classifiers

Table 7.17. Comparison of Fusion Techniques (Each time one classifier is removed - PV: Plurality Voting, BC: Borda Count, WBC: Weighted Borda Count, MIN: Minimum, MAX: Maximum, WSUM: Weighted Sum, PROD: Product)

	Rank Based			Similarity Score Based					
	PV	BC	WBC	MIN	MAX	MEDIAN	SUM	WSUM	PROD
All	95.6	93	95.3	84.4	87.5	95.1	95.6	94.8	90.7
All - C1	95.1	92	95.1	84.2	87.5	94.8	95.1	95.1	90.4
All - C2	94	90.1	94	83.6	85.7	95.3	94	94.8	90.4
All - C3	95.3	92.5	95.6	84.4	85.2	95.6	95.3	94.3	90.4
All - C4	94.3	91.7	95.1	83.1	83.9	94.8	94.3	94	89.9
All - C5	<b>96.1</b>	93.5	94.8	88.6	90.9	95.3	<b>96.1</b>	95.8	94.5
All - C6	94.3	93	95.3	82.9	86.2	95.1	94.3	<b>95.8</b>	89.9
All - C7	94.5	93	94.8	84.4	86.2	95.6	94.5	95.1	90.1
All - C8	94.5	91.4	94.5	84.4	86.2	94	94.5	94.5	90.7
Best Fusion	94.8	92.2	<b>96.4</b>	74.8	89.6	94	94.8	<b>96.9</b>	80.3

Decision combination of eight individual classifiers are implemented using various decision fusion techniques as discussed in Section 6.3. Table 7.17 displays the decision fusion results with the ensemble of eight as well as with a subset of seven classifier. Our first observation is that combination of all available classifiers does not necessarily give the best performance. Exhaustive evaluation of all subsets of the eight classifiers (C1 to C8) has revealed that the best fusion accuracy is obtained when seven of the classifiers are fused with HMM and 17-G combination left out. It is known that fusion techniques perform better when different modalities are used, either different classifiers or different data representations. In our case, HMM and HCRF are sequence classifiers operating on geometric data (G), and the inclusion of both deteriorates in fact the fusion performance.

It is encouraging to observe that all fusion schemes, i.e., the three rank-based schemes and the three similarity score based ones (median, sum, weighed sum) bring invariably seven to nine point improvements (Table 7.17). The best result occurs when the decisions of the seven classifiers (HMM with geometric features, that is left out,

Table 7.18. Best decision fusion results

	<b>G1</b>	<b>G2</b>	<b>G3</b>	<b>G4</b>	<b>G5</b>	<b>G6</b>	<b>G7</b>
<b>G1</b>	<b>100</b>	0	0	0	0	0	0
<b>G2</b>	0	<b>98.2</b>	1.8	0	0	0	0
<b>G3</b>	0	0	<b>98.2</b>	1.8	0	0	0
<b>G4</b>	0	0	1.8	<b>94.6</b>	0	0	3.6
<b>G5</b>	0	0	1.8	0	<b>96.4</b>	0	1.8
<b>G6</b>	0	0	0	0	0	<b>94.6</b>	5.4
<b>G7</b>	0	0	1.8	0	1.8	0	<b>94.6</b>
Average Recognition Accuracy (ARA): <b>96.9</b>							

Table 7.18) are fused using either WBC or sum of weighted similarity scores. Note that, we also analyzed the recognition performance of individual facial patches and their combinations in addition to using all four facial patches. We have found out that the best fusion result (96.9%) is obtained when the DCT-coefficients extracted only from the nose and mouth components (Patch-2 with 2 blocks and Patch-3 with 4 blocks, see Figure 5.4), with HMM classifier are used in the decision fusion.

## 7.2. Classification Methodology for LILir TwoTalk Database

The database contains 6 males and 2 females from various cultures, all of whom were native English speakers. Annotation of natural conversations is a challenging task since annotation can become person specific due to many factors such as annotator’s feelings to persons in a social situation or to the discussed topic, annotators emotional mood during the annotation, etc. To avoid this situation many annotators reviewed the video sequences to obtain a consensus on the labels of the segmented clips [10, 11]. Another challenging task is to split a longer video into clips. Therefore videos were manually splitted into clips by an observer by considering that each clip was a sample of one of the defined classes. The classes in the LILir TwoTalk corpus [10] are thinking, understanding, agreeing and questioning which frequently occur in natural conversations. Note that the length of the clips ranged from length  $l = 0.6$  to 10 seconds ( $\mu_l = 4.2\text{s}$ ,  $\sigma_l = 2.5\text{s}$ ) [11].

To compare the results of Sheerman and et al. [11] with our results, we used the same test and training organizations. The following steps are applied to select the test and training set for each non-verbal message category:

- (i) Choose the highest rated 25 clips among 527 clips as the positive set for that category.
- (ii) Choose the lowest rated 25 clips among 527 clips as the negative set for that category.
- (iii) Divide the positive and negative sets into two groups for a two-fold cross-validation test scheme.
- (iv) For each fold, use one group as the training set and the remaining group as the test set.

Among 527 clips, there were 109 agreement or disagreement, 140 understanding, 93 thinking and 65 questioning clips. Notice that a group of 120 clips were selected randomly in order to increase the variety of non-verbal messages, since they do not belong to any of the categories. Multi-class classification is not employed for this dataset since there are correlations between most of the categories, albeit weak (see Table 7.19). Note that correlation coefficient tending to 1 indicates that both communication signals occur together consistently and correlation that tends to -1 indicates that the expression of one signal can be occurred without observing the other category. A correlation of zero indicates that these two categories are independently occurring [11].

Two-class classification is carried out between positive and negative samples of a category instead of multi-class classification of the four categories. Because of the correlation between the categories some of the non-verbal message categories, e.g. agreeing and understanding, may include the same clip in their positive sample set. In the two-class classification scheme, the classifier of a certain non-verbal message category classifies the given test data as whether positive class or negative class of that category.

Clips extracted from the LIlir Twotalk corpus [10] do not start and end with neutral state as in BUHMAP database [9]. Therefore we need to define temporal time

Table 7.19. Calculated correlation coefficients using the average ratings for each category (taken from [11])

	Agreeing	Understanding	Thinking	Questioning
Agreeing	1			
Understanding	.46	1		
Thinking	-.21	-.23	1	
Questioning	-.18	-.40	0.06	1

windows for an effective feature extraction and mental state inference. Kaliouby [27] investigated the effect of inter-expression dynamics on people’s ability to recognize complex mental states. In her study, she found that two seconds is the minimum time required for a human to reliably infer a mental state. Consequently, one can conjecture that video segments of duration less than two seconds may induce inaccurate recognition results for humans.

By considering the inference given in the previous paragraph, we chose 2.5 seconds (approximately 64 frames at 25 fps) temporal windows for feature extraction and classification for the non-verbal message clips of LILir database [10]. The features extracted from the 64 frame long temporal window are given to the classifiers independently from the other frames. A 64 frame long temporal window slides 8 frames (0.32 seconds) at a time until the last frame of the clip is reached. By this way, we do not need to make time normalization for feature-subspace classification. However, for the clips lasting less than 2.5 seconds, we interpolate the extracted features to the desired frame length (64). Figure 7.2 illustrates the general flowchart for temporal feature extraction and classification of the clips.

### 7.3. Performance Results for the LILir TwoTalk Database

Table 7.20 summarizes the feature extraction methods and the classifiers employed. These features and classifiers are as follows: (i) HCRF sequential classifier with geometric features  $G$ , (ii) MNN based classification with ICA subspace projected landmark trajectory matrix  $P$  and (iii) MNN based classification with NMF subspace

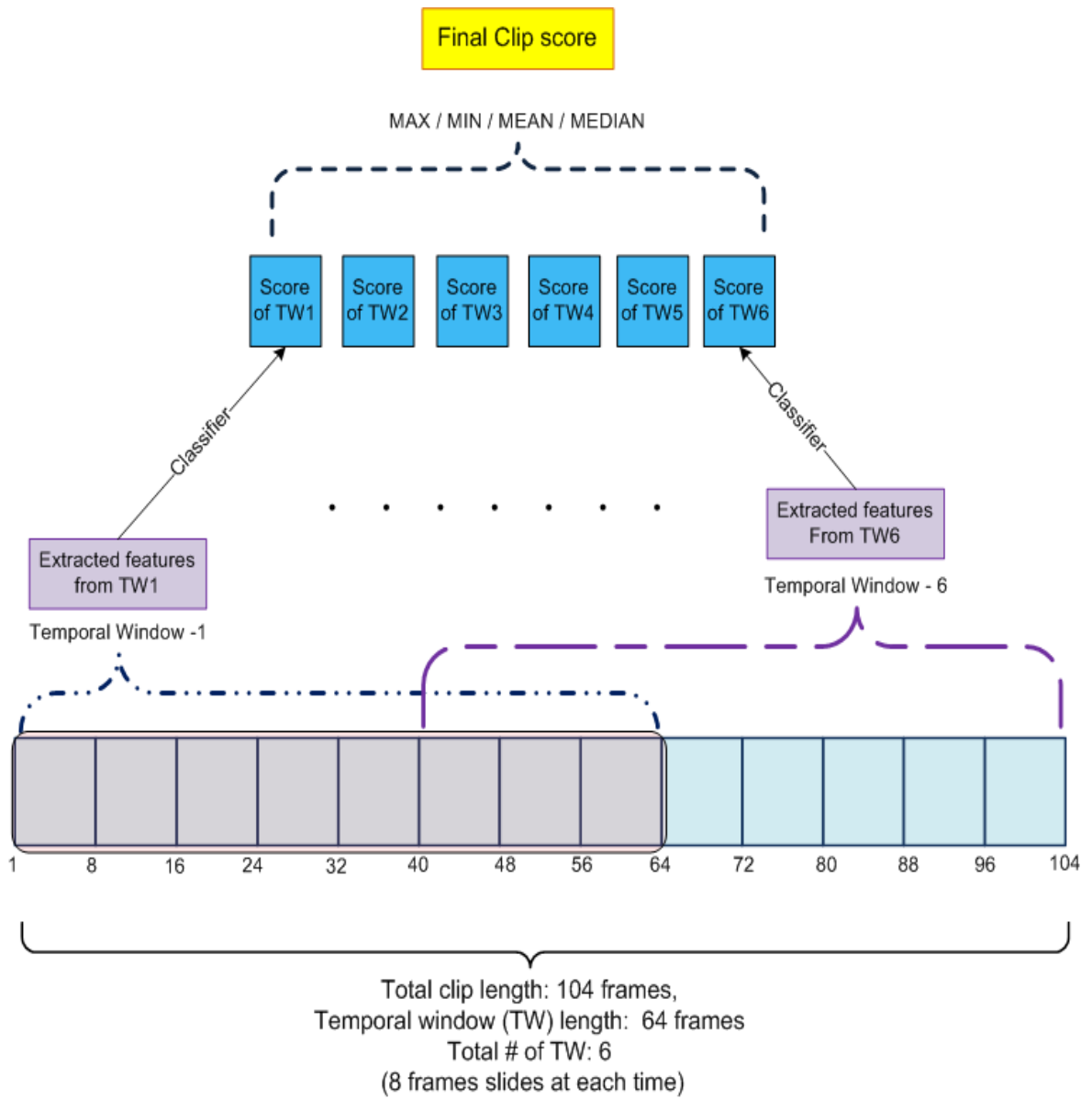


Figure 7.2. Temporal feature extraction and classification for a 104-frame long sample clip.

Table 7.20. Data types, extracted features and the classifiers for the LILir database

Data Type	$i^{th}$ time instance of sequence feature (Dimension)	Classifier
Landmark trajectories	Geometric Features (17x1)	HCRF
Data Type	Spatiotemporal Features (Dimension)	Classifier
Landmark trajectories	$P$ matrix (34x64)	ICA (75)
		NMF (350)
		MNN

projected landmark trajectory matrix  $P$ . These features and their accompanying classifiers are chosen for the non-verbal message analysis of LILir database [10] since these selected features and the classifiers perform better than the other features and classifiers for the BUHMAP database [9].

A two-class classification method is employed for each experiment set, because of the correlation between the non-verbal message categories. Experiment sets are summarized in Table 7.21. Classification performances of the classifiers are evaluated using the area under Receiver Operating Characteristic (ROC) curves, called as AuC (Area under the Curve). The AuC concept is more convenient for comparing classifiers as it reduces the multiclass performance to a single number, which otherwise would be given by the confusion matrices. Furthermore AuC is equivalent to the theoretical maximum achievable correct rate of a binary classification problem. In this way, we avoid measures like correct recognition, hit and false alarm rates which can sometimes be quite misleading since they depend on the operation threshold. When we formulate the FHG classification problem as one gesture - versus - all others, we have the two class classification problems. Thus, for each test clip we have two classifier scores, one positive class and one negative class (lowest rated samples). By using these scores we generate ROC curve of a classifier over a given test set as follows:

- (i) We subtract the score of the negative class from the positive class for each test sample. Store the difference of the scores in a vector.
- (ii) Then we normalize the elements of this vector between zero and one.
- (iii) Threshold values are set between zero and one with equal step sizes. Here we set 200 threshold values between zero and one.

Table 7.21. Experiment sets for LILir Database

Experiment Set	Category	# of training and test set
I	Agreeing	25 strongly rated positive samples and 25 lowest rated negative samples
II	Questioning	
III	Thinking	
IV	Understanding	

- (iv) For each threshold value we calculate the true positive and false positive rates. Figure 7.3 illustrates some of the defined threshold values and the corresponding true positive and false positive rates on a sample test set with its corresponding ROC curve.
- (v) At the end we plot the true positive versus false positive rates for each threshold value and hence we obtained the ROC curve.

Note that two classification paradigms are considered, namely: clip based classification (entire clip) and temporal window based classification. For clip based classification paradigm, only one inference is made for each clip after observing the all temporal window scores spanning that clip. Note that, for each temporal window a classification score is obtained. Final decision for a video clip is made by combining the similarity scores of the temporal windows spanning the entire clip by using the MIN, MEAN, MAX and MEDIAN similarity score fusing methods discussed in Section 6.3. For the temporal window based paradigm, classification is done for each temporal window separately. Thus each temporal window is classified independently from the other temporal windows and hence the whole clip (Fig. 7.2).

We get the following observations for the LILir TwoTalk Database [10]:

- (i) Clip based classification paradigm: Both feature-subspace classification and feature-sequence classification methods achieve similar classification performances for clip based classification paradigm. ICA features of landmark trajectory matrix with MNN classifier achieves 70 % average classification performance, the rest of the classifiers achieves 68.12% and 69.26 % average classification accuracy, which are

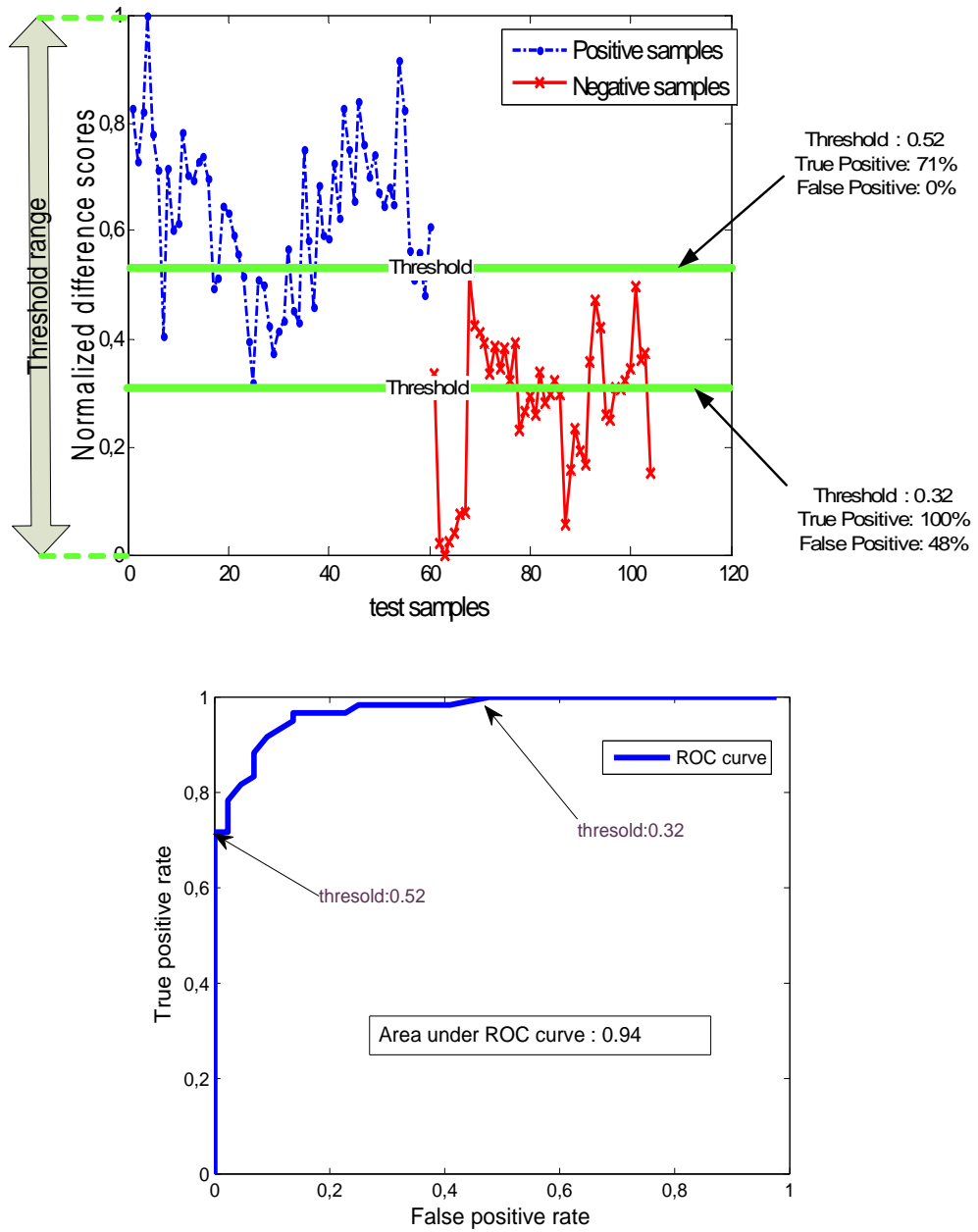


Figure 7.3. Normalized difference scores for a sample test set and the corresponding ROC curve

very close to each other’s performance (see Table 7.22). It is observed that “understanding” non-verbal message category is the best classified class among the others with 77.57 % and 79.31 % classification accuracy. On the other hand, “questioning” class seems more difficult for classification with 64.04 average classification accuracy over all classifiers.

- (ii) Temporal window based classification paradigm: HCRF sequential classifier with geometric features achieves 77.41% best individual average classification performance for temporal window based classification. Subspace-feature classification methods MNN with ICA and MNN with NMF performs similar average classification accuracy (74.54 % and 72.43 %) as summarized in Table 7.23. Again “questioning” category is the most confused class when it is averaged over the classifiers performance. It is also observed that MNN with ICA features achieves 84.57 % best classification accuracy for the “agreeing” non-verbal message category over all categories and classifiers.
- (iii) Clip based versus temporal window based classification: Average classification based on the temporal window paradigm outperforms the average classification based on clip paradigm (see Table 7.24). All the average classification accuracies of the classifiers are improved by about 4 percentage points to 8 percentage points per classifier. However, MNN classifier with ICA features based on temporal windows shows about 4 percentage points of performance drop for the classification of “understanding” category when compared to clip based classification paradigm and MNN with NMF features shows a similar performance drop for “questioning” category. But when we look at the overall classification performances averaged over the categories and the classifiers, we can see that temporal window based paradigm achieves higher classification accuracy compared to clip based classification paradigm. One possible reason of this result is that long clips can be classified more accurately than short clips with the designed classifiers.

There may be two possible explanation for the last observation. The first explanation is that the time interpolation of extracted features for the clips lasting less than 2.5 seconds may be disturbing the extracted features which are inherent to the underlying non-verbal message. Notice that time domain normalization process applied for

Table 7.22. Classification performances averaged over non-verbal message clips

	MNN with ICA ( $P$ )	MNN with NMF ( $P$ )	HCRF (17, $G$ )
Agreeing	71.53	66.29	69.83
Questioning	61.90	66.17	64.04
Thinking	69.01	66.77	63.84
Understanding	77.57	69.15	79.31
Total Average	70	68.12	69.26

Table 7.23. Classification performances averaged over temporal windows

	MNN with ICA ( $P$ )	MNN with NMF ( $P$ )	HCRF (17, $G$ )
Agreeing	84.57	75.79	78.73
Questioning	67.64	60.01	74.97
Thinking	83.49	76.09	73.48
Understanding	74.29	77.83	82.46
Total Average	74.54	72.43	77.41

the BUHMAP dataset [9] does not make an appreciable performance drop. The second explanation is that the non-verbal message clips lasting less than 2.5 seconds may be annotated incorrectly. Due to the experimental results of Kaliouby's study [27]; it is more efficient to assess a humans emotion by looking at the persons face historically at least over a two second window. This finding suggests that annotation of clips lasting less than two seconds may be annotated incorrectly.

The fusion schemes employed for the classifiers designed for the BUHMAP database [9] improves the classification performance about 9%. Therefore we employed the same fusion scheme to the classifiers designed for the LILir database [10]. Decision combination of the three individual classifiers are implemented using various decision fusion

Table 7.24. Comparison of average classification performances over message categories for two classification paradigms

	MNN with ICA ( $P$ )	MNN with NMF ( $P$ )	HCRF (17, $G$ )
Temporal Window based	77.54	72.43	77.41
Clip based	70	68.12	69.26

techniques that are described in Section 6.3. In Table 7.25, decision fusion results with the ensemble of three classifiers are displayed and compared with Sheerman-Chase's results [11]. Sheerman-Chase and et. al [11] extract four static feature types by only considering information from a single frame of video based on tracking of landmark coordinates as follows:

- (i) Tracking PCA values: First  $2J$  principal component vectors (eigenspace) are learned for the  $2J \times T$  training data where  $2J$  is the x and y coordinates of the tracked landmarks and  $T$  is the total number of frames. The tracking information for each frame was then projected into the eigenspace to give a  $2J$  vector (without dimension loss) that represent the deformation of the underlying head pose and facial expression. This  $2J$  vector was taken as the static features for that frame.
- (ii) Geometric features: They defined geometric features ( $g=12$ ) for head yaw, head pitch, head roll, eyebrow raise, lip pull/pucker and lips part as static features.
- (iii) Levenberg-Marquardt head pose estimation: Levenberg-Marquardt minimization was used to determine pose from a cloud a of  $J$  points.
- (iv) Affine head pose estimation: Affine head pose estimation was based on a transformation from the tracking positions of current video frame to a frame showing a frontal view of the face.

A polynomial equation is fitted to each static feature described above within a temporal window. Then the polynomial parameters are determined by regression and then used as temporal features that describe the evolution of the value of that static feature. Four different temporal window sizes are chosen. Feature selection and recognition is accomplished by using AdaBoost algorithm [142]. The classification results of Sheerman-Chase's algorithm is presented in Table 7.25.

It is observed that for temporal window-based classification paradigm about 5 percentage points performance improvement is obtained and for clip-based classification paradigm about 6 percentage points performance improvement is achieved. Notice that, in this case sum of the similarity scores gives the best classification fusion results. Consequently, highest classification performance (82.88 %) is achieved via temporal-

Table 7.25. Comparison of decision fusion results and Sheerman-Chase's results [11]

Decision fusion	Temporal Window based	Clip based	Sheerman-Chase	human
Agreeing	85.92	77.79	70	70
Questioning	78.18	67.12	73	93
Thinking	83.78	76.85	81	77
Understanding	83.65	83.09	80	82
Total Average	82.88	76.21	76	80.5

window based classification paradigm. Figures 7.4 and 7.5 illustrate the ROC curves for the best fusion results obtained by the fusion of temporal window based classification and clip base classification, respectively.

As a side remark, Sheerman-Chase and et al. [11] also presented the human performance assessment on clips containing strongly rated samples of non-verbal messages. Human performance assessment was accomplished by taking each rating of the clip by a human classifier output and comparing to the average of the other user ratings on that clip. The human classification method achieves 70% agreeing, 77 % thinking, 82% understanding and 93% questioning classification accuracy. Table 7.26 presents the comparison of human performance and clip-based decision fusion performance. Note however that it is not possible to make a fair comparison between human performance and the machine performance because human annotators were required to score each clip for four defined categories. Therefore, human performance represents multi-class (here four classes) classification strategy. On the other hand, our proposed method and the Sheerman-Chase method use a two-class classification strategy. It is observed that human performance on questioning category surpasses the performances of all classifiers. Another important difference is that the annotators not only watched the clips but they could also listen to the audio data. this is especially important since the clips are segmented according to the conversation between two persons. In this case, the audio data can give more information about the context of the conversation and it may be a good clue to make an inference about the affective message of the clips. Due to this fact, human classification performance especially for questioning category is higher than all other automatic classifiers. The advantage of human annotators is

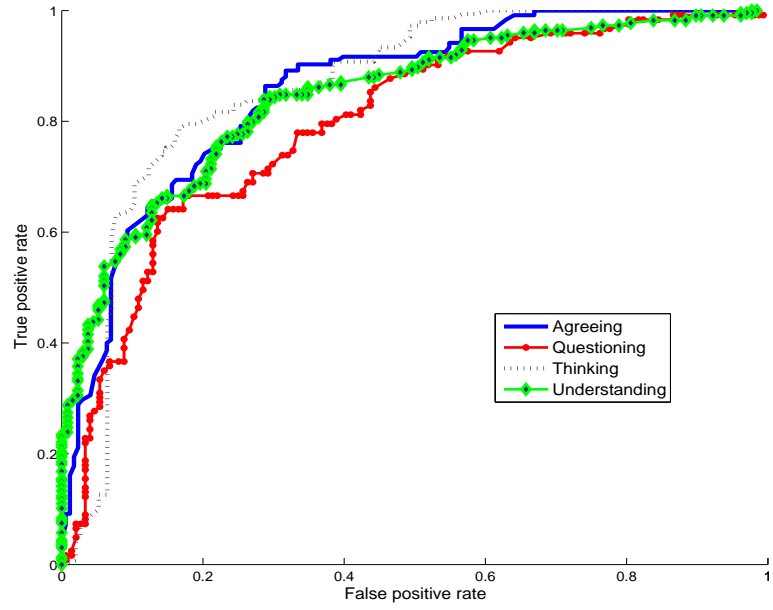


Figure 7.4. ROC of decision fusion for the classification of four categories via temporal window based classification.

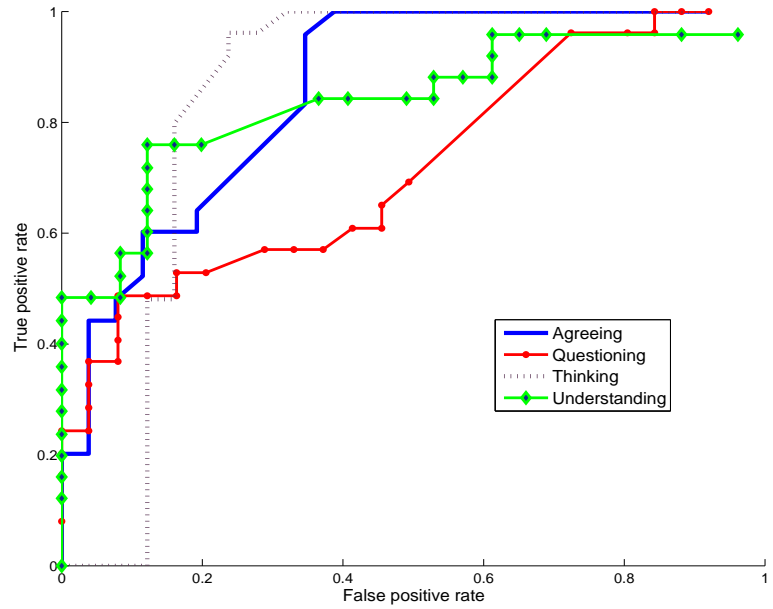


Figure 7.5. ROC of decision fusion for the classification of four categories via clip based classification

Table 7.26. Performance comparison of clip based decision fusion classification and human classification [11]

	Clip based decision fusion performance	Human performance
Agreeing	77.79	70
Questioning	67.12	93
Thinking	76.85	77
Understanding	83.09	82
Total Average	76.21	80.5

that they have the the audio information but on the other hand, they have to rate each clip for four categories.

#### 7.4. Performance Results for Cohn-Kanade Facial Expression Database

Cohn-Kanade facial expression database [71] is composed of six basic (prototypical) emotions namely: anger, disgust, fear, joy, sadness, and surprise almost without noticeable head pose. Experiments are conducted using leave-one-group-out cross-validation testing scheme to determine the performance of the volumetric 3D face data form on the novel subjects. Notice that, recognition results reported in this study are computed as the average of 10-fold testing. We designated five classifiers to take roles in fusion scheme as follows:

*Classifier - I:* MNN with 20 ICA coefficients of landmark trajectory matrice  $P$

*Classifier - II:* MNN with 280 global 3D DCT coefficients of  $V_{64 \times 32 \times 16}$

*Classifier - III:* MNN with 280 global 3D DCT coefficients of  $V_{64 \times 48 \times 16}$

*Classifier - IV:* MNN with 200 ICA features extracted from 3D DCT coefficients of face blocks of  $V_{64 \times 32 \times 16}$  (280 x 21 DCT coefficients reduced to 200 ICA )

*Classifier - V:* MNN with 200 ICA features extracted from 3D DCT coefficients of face blocks of  $V_{64 \times 48 \times 16}$  (280 x 35 DCT coefficients reduced to 200 ICA )

The recognition results of the individual classifiers are presented in Table 7.27. Table 7.28 summarizes the confusion matrix obtained by fusing the similarity scores of the

Table 7.27. Comparative results of individual classifiers

Class	Surprise	Happiness	Sadness	Fear	Disgust	Anger	Total
# of sequences	(70)	(83)	(49)	(45)	(40)	(35)	(322)
Classifier - I	84.3	80.7	87.8	46.7	87.5	77.14	78.3
Classifier - II	94.3	100	85.7	51.1	87.5	77.14	85.7
Classifier - III	95.7	97.6	81.6	57.8	90	80	86.3
Classifier - IV	85.7	80.7	83.7	80	92.5	82.9	83.9
Classifier - V	87.1	78.3	79.6	84.4	90	85.7	83.5

Table 7.28. Confusion matrix of 6-class facial expression recognition after decision fusion of five classifiers (overall recognition rate is 95.34)

	Surprise/70	Happiness/83	Sadness/49	Fear/45	Disgust/40	Anger/35
Surprise	100					
Happiness		97.6	1.2			1.2
Sadness			98			2
Fear	2.2	6.7		86.7	4.4	
Disgust				2.5	95	2.5
Anger		2.9	2.9		5.6	88.6

five classifiers with different feature and data types. Table 7.28 demonstrates that the fusion of five basic classifiers increases the overall facial expression recognition performance to 95.34 % where as the best individual classifier achieves only 86.3% recognition accuracy.

Table 7.29 lists the best recognition performances of our approach and other methods all conducted on Cohn–Kanade facial expression database [71]. We cannot make a direct and fair comparison between different approaches due to different experiment setups and preprocessings (e.g. manual or automatic registration and alignment of face images). We can conclude that our approach gives the second best reported overall recognition results on Cohn – Kanade facial expression database [71]. Notice that our proposed expression recognition system achieves slightly better recognition rates for surprise, happiness, sadness and disgust expressions when compared with Zhao’s [38] recognition rates with the same expressions. It is also observed that fear

Table 7.29. Comparison with other results from the literature. Since experimental conditions vary slightly, with each method we are reporting the number of subjects, the number of sequences and the number of classes analyzed(SI:Subject Independent,

	P:# of person, S:# of sequence, C:# of class			
ref.	Zhao07 [38]	Shan09 [40]	Yang09 [30]	<b>Ours</b>
(P,S,C)	(97,374,6)	(96,320,7)	(96,300,6)	(92,322,6)
test	10-fold SI	10-fold SI	1-fold SI	10-fold SI
Surprise	98.65	92.5	99.8	<b>100</b>
Happiness	96.04	90.1	<b>99.1</b>	97.6
Sadness	95.89	61.2	<b>97.8</b>	<b>97.8</b>
Fear	<b>94.64</b>	70	91.6	86.7
Disgust	94.74	92.5	94.1	<b>95</b>
Anger	<b>97.88</b>	66.6	97.3	88.6
Neutral	—	95.2	—	—
Total	<b>96.26</b>	95.1(6 class)	—	<b>95.34</b>
		92.6(7 class)		

and anger expressions are the most difficult categories for the classification. One of the reason is that, when we check the annotations made by different research groups, it is observed that annotators are not always agree for the label of fear and anger videos. The same expression video is inferred as different emotional expressions from different observers. Therefore, interpretation of fear and anger expressions is not an easy task even for humans.

We have the following observations:

- Global DCT features can discriminate the happiness and surprise expressions better than local block based DCT features. This is probably due to the fact that the lower frequency coefficients do not contain much discriminative information for facial expressions which involves only subtle appearance changes but on the other hand they are quite robust for the recognition of relatively more dominant and recognizable facial expressions such as joy and surprise.

- Local block-based DCT features are more effective for the classification of more easily confused expressions such as fear and anger. Indeed, partitioning the face into 21 or 35 blocks enable us to capture information more locally, which helps discriminate subtle local appearance changes. Furthermore the bigger number of block DCT coefficients means that more of the higher frequency coefficients were picked up.
- It is also observed that spatiotemporal 3D DCT features provide better overall recognition performance than landmark coordinate features. Also notice that landmark coordinate features can recognize sadness class slightly better than the 3D DCT features.
- Decision fusion of the classifiers outperforms the overall recognition rate of the best individual classifier about 9 percentage points. It is observed that all classifiers play a positive role in the decision fusion stage.

## 8. CONCLUSIONS

In this thesis, we have developed a fully automatic facial expression and head gesture analysis system based on visual information. We have concentrated on three tasks that are necessary for an affective human computer interface. These tasks are: (i) Accurate facial landmark localization, (ii) Robust and accurate facial landmark tracking and (iii) Classification of trajectories into face and head gestures. Following sections summarize the main conclusions extracted from each proposed approach.

### 8.1. Contributions of the Thesis

A summary of the main contributions of the thesis work is given in the following sections.

#### 8.1.1. Reliable Facial Landmark Detection

Robust and accurate localization of facial landmarks is essential for affective human computer interfaces based on facial behaviors. The main challenge in automatic landmark localization is the extreme variability of the configurations and of the local appearances and the training data dependency which handicaps database to database portability. We have addressed the requirements of robustness and accuracy as follows:

#### 8.1.2. Robustness by graph-based regularizer

- (i) Robustness by graph-based regularizer: Using images at coarse scale, that is low resolution images and the aid of structural regularization. Essentially regularization embodies anthropomorphic information. This structural correction is called Probabilistic Graph Model - I (*PGM-I*), and achieves robustness by eliminating unlikely landmark locations. We have observed that, PGM-I decreases the average coarse landmark localization error about 0.27 IOD.
- (ii) Accuracy by graph-based refinement and SVM-based search: Probabilistic Graph

Model - II (*PGM-II*) improves the landmark location accuracy by fusing information of the companion landmarks. The *PGM-II* improves the average accuracy of the landmarks about 20%. Final refinement with SVM classifiers improves the average accuracy of the coarsely localized landmarks about 15%.

In conclusion, the proposed two-stage facial landmarking algorithm maintains satisfactory performance face images under facial expressions and slight head rotations. It is characterized by:

- It relies on a subset of 7 fiducial landmarks and augments them reliably to 17 facial landmarks.
- The combinatorial search in *PGM-I* provides robustness, albeit at the cost of heavy processing.
- The step-by-step SVM search and *PGM-II* aid in improving accuracy.

The output of this landmark initialization algorithm is then used in facial landmark tracking algorithm (Section 4.4).

### 8.1.3. Facial Landmark Tracker

We have developed a multi-step tracking algorithm to successfully track facial landmarks under natural head movements and facial expressions. The algorithm is capable of tracking facial landmarks even under large head rotations (up to  $\pm 45^\circ$  yaw angles and about up to  $\pm 20^\circ$  tilt angles) and under facial expressions from subtle to strong valence. The tracker has generalization potential in that it performed satisfactorily in test sequences of unseen subjects under varying head rotations and facial movements.

We have found the following points worth emphasizing:

- Dynamically replenished template library: Since landmark appearances vary significantly, it is a good idea to start from one landmark template, typically the

one extracted during initialization, and keep updating the template library per landmark as significantly different appearances are encountered. A further consideration is to divide the template library into subsets. Since the factor that most affects the template shape is the yaw rotation, it is a good idea to divide the library into three libraries subtending three angle ranges.

- Shape regularization: Regularization of landmark configuration (shape regularization) is important to prevent drifting of landmarks into positions that make the ensemble configuration unlikely. This regularization is effected via PCA. However, shape regularization can also damage tracking precision, especially for lip and eyebrow landmarks.
- Refinement by local search: We have found that after all the operations of prediction and regularization one needs a final polishing step, that of local search by template matching. For example we have found in our work that the final refinement step improves the precision of the tracking algorithm about 19 % in terms of  $\Delta_{AUCDC}$  (see Section 4.4.3 for details).

The Kalman prediction was not very effective and at least for video cases we testes could be removed.

## 8.2. Facial Expression and Head Gesture Analyzer

The third contribution of the thesis is the FHG analyzer.

The main lessons learned during the design of the FHG classifier are the following:

- Facial landmark trajectories subjected to dynamic analysis via HMM or HCRF and the image of spatiotemporal matrices subjected to subspace analysis yield comparable performance, the latter being slightly better. This means that the subspace methods can extract the essence of the FHG action by considering the spatiotemporal totality of evidences. Note that sequential classifiers do inherent nonlinear time warping while in the case of subspace classifiers all sequences were linearly scaled for uniform duration.

- Between sequential classifiers HCRF performs slightly better than HMM for any chosen feature. For subspace classifiers, NMF is a better feature as compared to DCT and ICA.
- Subspace methods are best implemented using temporal sliding windows as the expression evolves from onset to apex and to offset. The sliding window based approach also enables local score or decision fusion.
- Decision fusion of the different feature sets and/or of the different classifiers improves the correct recognition performance significantly, that is by 10%.

### 8.2.1. Generalization to Use Cases

Our proposed automatic FHG analyzer can be easily adapted to many intelligent human-computer applications. A case in point is the automatic drowsiness detection of car drivers [24]. Current studies indicate that drowsiness reflect to both facial expressions as well as head gestures. We can conclude that the proposed automatic FHG analyzer can be effectively utilized for many affective and intelligent human-computer applications.

## 8.3. Future Work

There are several avenues of research possible to our FHG analysis scheme.

- (i) Cooperative tracking: Presently a separate Kalman filter is used for predicting the motion of each landmark. Instead, the joint dynamics of facial landmarks can be modeled and introduced into a multi-landmark configuration. This will also incorporate the structural information, as used in PGM-I, so that path prediction and shape regularization can be made jointly.
- (ii) Landmark Shedding: For severe yaw and tilt angles some landmarks become invisible or become too risky, that is, noisy, we can remove and add some certain landmarks according to the severity of the pose. For example, the outer eyebrow corners, suffer for large yaw angles and their uncertainty affects negatively upon the accuracy of the neighboring landmarks. Additional landmarks can be designated

between nose and mouth corners to encode furrow.

- (iii) Eye gaze analysis: Gaze tracking is an integral [143]. Gaze analysis can provide clues for focus of attention, a reflecting mind, surprise etc. Eye pupils can be detected and tracked accurately by using either near infrared cameras or by higher resolution face images [88, 102].
- (iv) Alternative feature extraction methods: The following alternative facial features remain to be investigated:
  - (a) Data matrices (appearance, gesture and landmark trajectory) (Section 5.2.3) were processed as intensity images via subspace projection algorithms. Alternatively, these data matrices can be analyzed by row by row so that a separate time series analysis can be carried out per row of the matrix. Recall that each row corresponds to a specific landmark, a specific geometric feature or a specific DCT coefficient of a patch. The multitude of classifications can then be fused via some score or decision fusion.
- (v) Adaboost [142] based feature selection algorithms can be employed followed by the Adaboost classifier itself or an SVM, KNN, MNN etc. type of classifier operating on the Adaboost-selected features. Finally, a recently announced extension of the Cohn-Kanade database [144] remains as an important testbed for the performance of our FHG analyzer.
- (vi) Multi-modal FHG analysis: Our FHG analysis system utilizes only visual information extracted from the affective state videos. However, it is obvious that, supporting more modalities such as speech data accompanying the facial behavior, whenever available, will improve the recognition power of the system. Context clue is also important in order to make a correct interpretation of the facial expressions and head gestures. Context may include many useful evidences such as the time and location of an event, personal information about the user etc.
- (vii) Temporal affective segmentation: In this thesis, we have only studied the classification of manually segmented videos. A very relevant future challenge is classification with automatic temporal segmentation of the videos. Determining the onset, apex and offset phases of an expression within a time series can be useful for continuous and spontaneous facial expression analysis.
- (viii) 3D tensor ICA decomposition of spatiotemporal data: Presently spatiotemporal

feature sequence of a whole shot is organized as a single vector. Then, the data matrix is formed as a spatiotemporal features x actors. Instead, we can investigate 3D Tensor ICA decomposition [145, 146] to take into account more explicitly the temporal evolution by representing the data matrix as 3D Tensor such as (spatial domain) x (temporal domain) x (actors).

## REFERENCES

1. Cootes, T. F. and C. Taylor, “Statistical Models of Appearance for Medical Image Analysis and Computer Vision”, *In Proc. SPIE Medical Imaging*, pp. 236–248, 2001.
2. Dutagacı, H., *Object Recognition in Subspaces: Applications in Biometry and 3D Model Retrieval*, Ph.D. thesis, Boğaziçi University, 2009.
3. Tian, Y., T. Kanade, and J. Cohn, “Recognizing action units for facial expression analysis”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 2, pp. 97–115, Feb. 2001.
4. Akakin, H. C., L. Akarun, and B. Sankur, “Robust 2D/3D Landmarking”, *3DTV Conference*, pp. 1–4, Kos Island, Greece, 2007.
5. Lades, M., J. C. Vorbruggen, J. Buhmann, J. Lange, C. V. D. Malsburg, R. P. Wrtz, and W. Konen, “Distortion Invariant Object Recognition in the Dynamic Link Architecture”, *IEEE Transactions on Computers*, Vol. 42, pp. 300–311, 1993.
6. Wiskott, L., J. Fellous, N. Krger, and C. der Malsburg, “Face recognition by elastic bunch graph matching”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 775–779, July 1997.
7. Demirkir, C. and B. Sankur, “Face Detection Using Look-up Table Based Gentle AdaBoost”, *Audio and Video-based Biometric Person Authentication*, 2005.
8. Savran, A., N. Alyuz, H. Dibeklioglu, O. Celiktutan, G. B., B. Sankur, and L. Akarun, “Bosphorus Database for 3D Face Analysis”, *The First COST 2101 Workshop on Biometrics and Identity Management*, May 2008.
9. Aran, O., I. Ari, M. A. Guvensan, H. Haberdar, Z. Kurt, H. I. Turkmen, A. Uyar,

- and L. Akarun, “A Database of Non-Manual Signs in Turkish Sign Language”, *IEEE 15th Signal Processing and Communications Applications Conference (SIU '07)*, June 2007.
10. Bowden, R., *LILiR Twotalk Corpus*, University of Surrey, 2010, [http://www.ee.surrey.ac.uk/Projects/LILiR/twotalk\\_corpus](http://www.ee.surrey.ac.uk/Projects/LILiR/twotalk_corpus).
  11. Sheerman-Chase, T., E.-J. Ong, and R. Bowden, “Feature Selection of Facial Displays for Detection of Non Verbal Communication in Natural Conversation”, *IEEE International Workshop on Human-Computer Interaction*, Kyoto, Oct 2009.
  12. Pantic, M. and L. J. M. Rothkrantz, “Toward an Affect-Sensitive Multimodal Human-Computer Interaction”, *Proceedings of the IEEE*, pp. 1370–1390, 2003.
  13. Zeng, Z., M. Pantic, G. Roisman, and T. Huang, “A Survey of Affect Recognition Methods: Audio, Visual, and Spontaneous Expressions”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 31, No. 1, pp. 39–58, January 2009.
  14. Vinciarelli, A., M. Pantic, and H. Bourlard, “Social signal processing: Survey of an emerging domain”, *Image and Vision Computing*, Vol. 27, No. 12, pp. 1743 – 1759, 2009.
  15. Baron-Cohen, S., A. Riviere, M. Fukushima, D. French, J. Hadwin, P. Cross, C. Bryant, and M. Sotillo, “Reading the Mind in the Face: A Cross-cultural and Developmental Study”, *Visual Cognition*, Vol. 3, pp. 39–60(22), 1 March 1996.
  16. Back, E., T. R. Jordan, and S. M. Thomas, “The recognition of mental states from dynamic and static facial expressions”, *Visual Cognition*, Dec. 2008.
  17. Knapp, M. L. and J. A. Hall, *Nonverbal communication in human interaction*, Belmont, CA : Wadsworth/Thomson Learning, 6th ed edition, 2006.

18. Mehrabian, A. and S. Ferris, “Inference of attitude from nonverbal communication in two channels”, *Journal of Counseling Psychology*, Vol. 31, No. 3, pp. 248–252, June 1967.
19. Picard, R. W., *Affective computing*, MIT Press, Cambridge, MA, USA, 1997.
20. Duric, Z., W. D. Gray, R. Heishman, S. Member, F. Li, A. Rosenfeld, M. J. Schoelles, C. Schunn, and H. Wechsler, “Integrating Perceptual and Cognitive Modeling for Adaptive and Intelligent Human-Computer Interaction”, *Proc. of the IEEE*, pp. 1272–1289, 2002.
21. Gatica-Perez, D., “Automatic nonverbal analysis of social interaction in small groups: A review”, *Image and Vision Computing*, Vol. 27, No. 12, pp. 1775 – 1787, 2009.
22. Sebe, N., M. Lew, Y. Sun, I. Cohen, T. Gevers, and T. Huang, “Authentic facial expression analysis”, *Image and Vision Computing*, Vol. 25, No. 12, pp. 1856–1863, December 2007.
23. Yang, J. H., Z.-H. Mao, L. Tijerina, T. Pilutti, J. F. Coughlin, and E. Feron, “Detection of driver fatigue caused by sleep deprivation”, *Trans. Sys. Man Cyber. Part A*, Vol. 39, No. 4, pp. 694–705, 2009.
24. Vural, E., M. Çetin, A. Erçil, G. Littlewort, M. S. Bartlett, and J. R. Movellan, “Drowsy Driver Detection Through Facial Movement Analysis”, *ICCV-HCI*, pp. 6–18, 2007.
25. Ji, Q., P. Lan, and C. Looney, “A probabilistic framework for modeling and real-time monitoring human fatigue”, *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, Vol. 36, No. 5, pp. 862–875, 2006.
26. Kapoor, A., W. Burleson, and R. W. Picard, “Automatic prediction of frustration”, *International Journal of Human-Computer Studies*, Vol. 65, No. 8, pp. 724

- 736, 2007.
27. Kaliouby, R. A., “Mind-reading machines: automated inference of complex mental states”, Technical report, UCAM-CL-TR-636, 2005.
  28. Picard, R. W., , and M. Curie, “Affective Computing”, Technical Report 321, M.I.T Media Laboratory Perceptual Computing Section, 1995.
  29. Viola, P. and M. Jones, “Robust Real-time Object Detection”, *International Journal of Computer Vision*, 2001.
  30. Yang, P., Q. Liu, and D. N. Metaxas, “Boosting encoded dynamic features for facial expression recognition”, *Pattern Recognition Letters*, Vol. 30, No. 2, pp. 132 – 139, 2009.
  31. Vukadinovic, D. and M. Pantic, “Fully Automatic Facial Feature Point Detection Using Gabor Feature Based Boosted Classifiers”, *In SMC05*, pp. 1692–1698, 2005.
  32. Cristinacce, D., T. Cootes, and I. Scott, “A Multi-Stage Approach to Facial Feature Detection”, *15<sup>th</sup> British Machine Vision Conference, London, England*, pp. 277–286, 2004.
  33. Eckhardt, M., I. R. Fasel, and J. R. Movellan, “Towards Practical Facial Feature Detection.”, *Int. Journal of Pattern Recognition and Artificial Intelligence*, Vol. 23, No. 3, pp. 379–400, 2009.
  34. Akakin, H. C. and B. Sankur, “Analysis of Head and Facial Gestures Using Facial Landmark Trajectories”, *COST 2101/2102 Conference*, pp. 105–113, 2009.
  35. Bartlett, M. S., G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan, “Recognizing Facial Expression: Machine Learning and Application to Spontaneous Behavior”, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 2, pp. 568–573, IEEE Computer Society, Los Alamitos, CA, USA, 2005.

36. Littlewort, G., M. S. Bartlett, I. Fasel, J. Susskind, and J. Movellan, “Dynamics of Facial Expression Extracted Automatically from Video”, *Journal of Image and Vision Computing*, pp. 615–625, 2004.
37. Tian, Y., “Evaluation of Face Resolution for Expression Analysis”, *CVPR Workshop on Face and Video*, p. 82, 2004.
38. Zhao, G. and M. Pietikainen, “Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 6, pp. 915–928, June 2007.
39. Sheerman-Chase, T., E.-J. Ong, and R. Bowden, “Online Learning of Robust Facial Feature Trackers”, *3rd IEEE On-line Learning for Computer Vision Workshop*, Kyoto, Oct 2009.
40. Shan, C., S. Gong, and P. McOwan, “Facial Expression Recognition Based on Local Binary Patterns: A Comprehensive Study”, *Image and Vision Computing*, Vol. 27, No. 6, pp. 803–816, May 2009.
41. Anderson, K. and P. McOwan, “Real-Time Automated System for the Recognition of Human Facial Expressions”, *Systems, Man and Cybernetics - B*, Vol. 36, No. 1, pp. 96–105, February 2006.
42. Baskan, S., M. M. Bulut, and V. Atalay, “Projection based method for segmentation of human face and its evaluation”, *Pattern Recognition Letters*, Vol. 23, No. 14, 2002.
43. Wong, K., K. Lam, and W. Siu, “An efficient algorithm for human face detection and facial feature extraction under different conditions”, *Pattern Recognition*, Vol. 34, No. 10, pp. 1993–2004, October 2001.
44. Arca, S., P. Campadelli, and R. Lanzarotti, “A face recognition system based

- on automatically determined facial fiducial points”, *Pattern Recognition*, Vol. 39, No. 3, 2006.
45. Colbry, D., G. Stockman, and A. Jain, “Detection of Anchor Points for 3D Face Verification”, *Proceedings of Workshop on Advanced 3D Imaging for Safety and Security*, 2005.
  46. Feris, R. S., J. Gemmell, K. Toyama, and V. Krger, “Hierarchical Wavelet Networks for Facial Feature Localization”, *In ICCV01 Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-Time Systems*, pp. 118–123, 2002.
  47. Ryu, Y.-S. and S.-Y. Oh, “Automatic extraction of eye and mouth fields from a face image using eigenfeatures and multilayer perceptrons”, *Pattern Recognition*, Vol. 34, No. 12, pp. 2459 – 2466, 2001.
  48. F., S. and J. Bigun, “Retinal vision applied to facial features detection and face authentication”, *Pattern Recognition Letters*, Vol. 23, No. 4, pp. 463–475, February 2002.
  49. Cinar Akakin, H., A. Salah, L. Akarun, and B. Sankur, “2D/3D facial feature extraction”, *Proceedings of the SPIE*, pp. 441–452, 2006.
  50. Salah, A. A., H. Çınar-Akakin, L. Akarun, and B. Sankur, “Robust facial landmarking for registration”, *Annals of Telecommunications*, Vol. 62, No. 1-2, pp. 1608–1633, 2007.
  51. Boehnen, C. and T. Russ, “A Fast Multi-Modal Approach to Facial Feature Detection”, *Seventh IEEE Workshops on Application of Computer Vision*, 2005.
  52. Zhu, X., J. Fan, and A. Elmagarmid, “Towards facial feature extraction and verification for omni-face detection in video/images”, *Int. Conf. on Image Processing*, pp. II: 113–116, 2002.

53. Wang, S.-L., W.-H. Lau, A. W.-C. Liew, and S.-H. Leung, “Robust lip region segmentation for lip images with complex background”, *Pattern Recognition*, Vol. 40, No. 12, pp. 3481–3491, 2007.
54. Wang, S., W. Lau, and S. Leung, “Automatic lip contour extraction from color images”, *Pattern Recognition*, Vol. 37, No. 12, pp. 2375–2387, December 2004.
55. Antonini, G., V. Popovici, and J. Thiran, “Independent Component Analysis and Support Vector Machine for Face Feature Extraction”, *Audio and Video Based Biometric Person Authentication*, pp. 111–118, 2003.
56. Shih, F. Y. and C.-F. Chuang, “Automatic extraction of head and face boundaries and facial features”, *Information Sciences*, Vol. 158, No. 1, 2004.
57. Martiriggiano, T., M. Leo, P. Spagnolo, and T. d’Orazio, “Facial feature extraction by kernel independent component analysis”, *IEEE Conf. on Advanced Video and Signal Based Surveillance*, pp. 270–275, 2005.
58. Zobel, M., A. Gebhard, D. Paulus, J. Denzler, and H. Niemann, “Robust Facial Feature Localization by Coupled Features”, *Fourth IEEE Int. Conf. on Automatic Face and Gesture Recognition*, 2000.
59. Gourier, N., D. Hall, and J. L. Crowley, “Facial features detection robust to pose, illumination and identity”, *IEEE International Conference on Systems, Man & Cybernetics*, pp. 617–622, 2004.
60. Nguyen, M. H., J. Perez, and F. D. la Torre Frade, “Facial Feature Detection with Optimal Pixel Reduction SVMs”, *8th IEEE International Conference on Automatic Face and Gesture Recognition*, September 2008.
61. Wang, Q., C. Zhao, and J. Yang, “Efficient Facial Feature Detection Using Entropy and SVM”, *Int. Symposium on Visual Computing*, pp. I: 763–771, 2008.
62. “Face detection and facial feature localization without considering the appearance

- of image context”, *Image and Vision Computing*, Vol. 25, No. 5, pp. 741 – 753, 2007.
63. Cootes, T., C. Taylor, D. Cooper, and J. Graham, “Active shape models - their training and application”, *Computer Vision Image Understanding*, Vol. 61, No. 1, pp. 38–59, 1995.
  64. Cootes, T., G. J. Edwards, and C. J. Taylor, “Active appearance models”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 6, pp. 681–684, June 2001.
  65. Dryden, I. and K. Mardia, *Statistical Shape Analysis*, John Wiley and Sons, 1998.
  66. Xue, Z., S. Z. Li, and E. K. Teoh, “Bayesian Shape Model for Facial Feature Extraction and Recognition”, *Pattern Recognition*, Vol. 36, pp. 2819–2833, 2003.
  67. Jesorsky, O., K. J. Kirchberg, and R. W. Frischholz, “Robust Face Detection Using the Hausdorff Distance”, pp. 90–95, Springer, 2001.
  68. Messer, K., J. Matas, J. Kittler, J. Luettin, and G. Maitre, “Extended M2VTS Database”, *Proceedings 2nd Conference on Audio and Video-base Biometric Personal Verification*, 1999, <http://www.ee.surrey.ac.uk/Research/VSSP/xm2vtsdb>.
  69. Phillips, P., H. Moon, S. Rizvi, and P. Rauss, “The FERET Evaluation Methodology for Face-Recognition Algorithms”, *Pattern Analysis and Machine Intelligence*, Vol. 22, No. 10, pp. 1090–1104, October 2000.
  70. Bailly-Baillire, E., S. Bengio, F. Bimbot, M. Hamouz, J. Mariethoz, J. Matas, K. Messer, F. Poree, and B. Ruiz, “The BANCA Database and Evaluation Protocol”, Technical report, IDIAP, IRISA, EPFL, University of Carlos, University of Surrey, 2003.
  71. Kanade, T., J. Cohn, and Y.-L. Tian, “Comprehensive Database for Facial Expression Analysis”, *Proceedings of the 4th IEEE International Conference on Au-*

- Automatic Face and Gesture Recognition (FG'00)*, pp. 46 – 53, March 2000.
72. Zhu, Z. and Q. Ji, “Robust Pose Invariant Facial Feature Detection and Tracking in Real-Time”, *International Conference on Pattern Recognition*, pp. I: 1092–1095, 2006.
  73. Asteriadis, S., N. Nikolaidis, and I. Pitas, “Facial feature detection using distance vector fields”, *Pattern Recognition*, Vol. 42, No. 7, pp. 1388–1398, 2009.
  74. Salah, A. and L. Akarun, “3D Facial Feature Localization for Registration”, *Int. Workshop on Multimedia Content Representation, Classification and Security*, pp. 338–345, 2006.
  75. Lu, X. and A. K. Jain, “Multimodal facial feature extraction for automatic 3D face recognition”, Technical report, MSU-CSE-05-22, Michigan State University, 2005.
  76. Wang, Y., C.-S. Chua, and Y.-K. Ho, “Facial feature detection and face recognition from 2D and 3D images”, *Pattern Recognition Letters*, Vol. 23, No. 10, pp. 1191–1202, 2002.
  77. Romero, M. and N. Pears, “Landmark Localisation in 3D Face Data”, *IEEE Conference on Advanced Video and Signal Based Surveillance*, pp. 73–78, IEEE Computer Society, Los Alamitos, CA, USA, 2009.
  78. Tomasi, C. and T. Kanade, “Detection and Tracking of Point Features”, *International Journal of Computer Vision*, 1991.
  79. Feris, R., C. Junior, and R.M., “Tracking Facial Features Using Gabor Wavelet Networks”, *Proc. IEEE Sibgraphi*, pp. 22–27, 2000.
  80. Buenaposada, J. M., E. Muñoz, and L. Baumela, “Efficient illumination independent appearance-based face tracking”, *Image Vision Computing*, Vol. 27, No. 5, pp. 560–578, 2009.

81. McKenna, S., R. Gong, J. Wurtz, and D. Tanner, "Tracking facial feature points with Gabor wavelets and shape models", *Proc. of Int. Conf. on Audio-and Video-based Biometric Person Authentication*, 1997.
82. Cristinacce, D. and T. F. Cootes, "Facial Feature Detection and Tracking with Automatic Template Selection", *Int. Conf. On Automatic Face and Gesture Recognition (FGR)*, 2006.
83. F., D. and F. Davoine, "Online Appearance-based Face and Facial Feature Tracking", *Proc. of the 17th Int. Conf. on Pattern Recognition*, 2004.
84. Kanaujia, A., Y. Huang, and D. Metaxas, "Emblem Detections by Tracking Facial Features", *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, p. 108, 2006.
85. Tong, Y., Y. Wang, Z. Zhu, and Q. Ji, "Robust facial feature tracking under varying face pose", *Pattern Recognition*, Vol. 40, pp. 3195–3208, 2007.
86. Tsalakanidou, F. and S. Malassiotis, "Real-time 2D+3D facial action and expression recognition", *Pattern Recognition*, Vol. 43, No. 5, pp. 1763 – 1775, 2010.
87. Bailenson, J. N., E. D. Pontikakis, I. B. Mauss, J. J. Gross, M. E. Jabon, C. A. C. Hutcherson, C. Nass, and O. John, "Real-time classification of evoked emotions using facial feature tracking and physiological responses", *Int. J. Hum.-Comput. Stud.*, Vol. 66, No. 5, pp. 303–317, 2008.
88. Zhang, Y. and Q. Ji, "Active and Dynamic Information Fusion for Facial Expression Understanding from Image Sequences", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 27, No. 5, pp. 699–714, May 2005.
89. Hupont, I., E. Cerezo, and S. Baldassarri, "Facial Emotional Classifier For Natural Interaction", *Electronic Letters on Computer Vision and Image Analysis*, Vol. 7, No. 4, pp. 1–12, 2008.

90. Dornaika, F. and F. Davoine, “Simultaneous Facial Action Tracking and Expression Recognition in the Presence of Head Motion”, *International Journal of Computer Vision*, Vol. 76, No. 3, pp. 257–281, March 2008.
91. Wang, T. and J. J. James Lien, “Facial expression recognition system based on rigid and non-rigid motion separation and 3D pose estimation”, *Pattern Recognition*, Vol. 42, pp. 962–977, 2009.
92. Pantic, M. and L. Rothkrantz, “Facial Action Recognition for Facial Expression Analysis from Static Face Images”, *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, Vol. 34, No. 3, pp. 1449–1461, June. 2004.
93. Ekman, P. and W. Friesen, *Facial Action Coding System: A Technique for the Measurement of Facial Movement.*, Consulting Psychologists Press, Palo Alto, 1978.
94. Pantic, M., “Machine analysis of facial behaviour: naturalistic and dynamic behaviour”, *Philosophical Transactions of the Royal Society B: Biological Sciences*, Vol. 364, No. 1535, pp. 3505–3513, December 2009.
95. Russell, J. A. and J. M. Fernandez Dols, *The psychology of facial expression*, Cambridge University Press, 1997.
96. Ambadar, Z., J. Schooler, and C. J. F., “Deciphering the enigmatic face: The importance of facial dynamics in interpreting subtle facial expressions”, *Psychological Science*, Vol. 16, pp. 403–410, 2005.
97. Ekman, P. and E. Rosenberg, “What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS)”, *Oxford University Press*, 2004.
98. Cunningham, D. W. and C. Wallraven, “Dynamic information for the recognition of conversational expressions”, *Journal of Vision*, Vol. 9, No. 13:7, pp. 1–17, 12

- 2009.
99. Kapoor, A. and R. W. Picard, "A Real-Time Head Nod and Shake Detector", *Proceedings from the Workshop on Perspective User Interfaces*, 2001.
  100. Kang, Y. G., H. J. Joo, and P. K. Rhee, "Real Time Head Nod and Shake Detection Using HMMs", *Knowledge-Based Intelligent Information and Engineering Systems*, Vol. 4253, pp. 707–714, Springer Berlin / Heidelberg, 2006.
  101. Morency, L.-P., C. Sidner, C. Lee, C. Lee, and T. Darrell, "Contextual Recognition of Head Gestures", *Proceedings of the Seventh International Conference on Multimodal Interfaces*, pp. 18–24, 2005.
  102. Maat, L. and M. Pantic, "Gaze-X: Adaptive affective multimodal interface for single-user office scenarios", *Proc. ACM Intl Conf. Multimodal Interfaces*, pp. 171–178, 2006.
  103. Kamarainen, J., V. Kyrki, and H. Kalviainen, "Invariance Properties of Gabor Filter-Based Features: Overview and Applications", *Image Processing*, Vol. 15, No. 5, pp. 1088–1099, May 2006.
  104. Liu, C. and H. Wechsler, "Independent Component Analysis of Gabor Features for Face Recognition", *IEEE Transactions on Neural Networks*, Vol. 14, pp. 919–928, 2003.
  105. Daugman, J., "Complete Discrete 2D Gabor Transforms by Neural Networks for Image Analysis and Compression", *IEEE Trans. Acoustics, Speech, and Signal Processing*, Vol. 36, No. 7, pp. 1169–1179, July 1988.
  106. Ekenel, H. and B. Sankur, "Feature selection in the independent component subspace for face recognition", *Pattern Recognition Letters*, Vol. 25, No. 12, pp. 1377–1388, September 2004.
  107. Bartlett, M., J. Movellan, and T. Sejnowski, "Face recognition by independent

- component analysis”, *IEEE Transactions On Neural Networks*, Vol. 13, No. 6, pp. 1450–1464, November 2002.
108. Hyvarinen, A., “Fast and robust fixed-point algorithms for independent component analysis.”, *IEEE Trans Neural Networks*, Vol. 10, No. 3, pp. 626–34, 1999.
109. Oja, E., “Independent component analysis: algorithms and applications”, *Neural Networks*, Vol. 13, pp. 411–430, 2000.
110. Burges, C. J. C., “A Tutorial on Support Vector Machines for Pattern Recognition”, *Data Min. Knowl. Discov.*, Vol. 2, No. 2, pp. 121–167, 1998.
111. Chang, C.-C. and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001, <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
112. Sanderson, C. and K. K. Paliwal, “Features for robust face-based identity verification”, *Signal Processing*, Vol. 83, No. 5, pp. 931 – 940, 2003.
113. Ekenel, H. K. and R. Stiefelhagen, “Analysis of Local Appearance-Based Face Recognition: Effects of Feature Selection and Feature Normalization”, *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, p. 34, IEEE Computer Society, Washington, DC, USA, 2006.
114. Phillips, P. J., P. J. Flynn, T. Scruggs, K. W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, “Overview of the Face Recognition Grand Challenge”, *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1*, pp. 947–954, IEEE Computer Society, Washington, DC, USA, 2005.
115. Welch, G. and G. Bishop, “An Introduction to the Kalman Filter”, Technical report, 1995, <http://www.cs.unc.edu/~welch/kalman/kalmanIntro.html>.
116. Seong, C., B. Kang, J. Kim, and S. Kim, “Effective Detector and Kalman Filter Based Robust Face Tracking System”, *The Pacific-Rim Symposium on Image and*

- Video Technology*, pp. 453–462, 2006.
117. Ziliani, F. and F. Moscheni, “Kalman filtering motion prediction for recursive spatio-temporal segmentation and object tracking”, *Proc. of The Workshop on Image Analysis for Multimedia Interactive Services*.
  118. Lee, S., J. Kang, J. Shin, and J. Paik, “Hierarchical active shape model with motion prediction for real-time tracking of non-rigid objects”, *IET - Computer Vision*, Vol. 1, No. 1, pp. 17–24, March 2007.
  119. Welch, G. F., “History: The use of the kalman filter for human motion tracking in virtual reality”, *Presence: Teleoperators and Virtual Environments*, Vol. 18, No. 1, pp. 72–91, 2009.
  120. Doucet, A. and A. M. Johansen, “A tutorial on particle filtering and smoothing: fifteen years later”, *In Handbook of Nonlinear Filtering (eds, University Press, 2009*.
  121. La Scala, B. F., R. R. Bitmead, and M. R. James, “Conditions for stability of the extended Kalman filter and their application to the frequency tracking problem”, *Mathematics of Control, Signals, and Systems*, Vol. 8, No. 1, pp. 1–26, March 1995.
  122. Murphy, K., *Kalman filter toolbox for Matlab*, 2005, <http://www.cs.ubc.ca/~murphyk/Software/Kalman/kalman.html>.
  123. Valstar, M. F., M. Pantic, Z. Ambadar, and J. F. Cohn, “Spontaneous vs. posed facial behavior: automatic analysis of brow actions”, *Proceedings of the 8th international conference on Multimodal interfaces*, 2006.
  124. Cohn, J. F. and K. L. Schmidt, “The timing of facial motion in posed and spontaneous smiles”, *Journal of Wavelets, Multi-resolution & Information Processing*, Vol. 2, pp. 1–12, 2004.

125. Quattoni, A., S. Wang, L.-P. Morency, M. Collins, and T. Darrell, “Hidden Conditional Random Fields”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 10, pp. 1848–1852, 2007.
126. Morimoto, C., Y. Yacoob, and L. Davis, “Recognition of Head Gestures Using Hidden Markov Models”, *In Proceeding of ICPR*, pp. 461–465, 1996.
127. Lafferty, J. D., A. McCallum, and F. C. N. Pereira, “Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data”, *Proceedings of the 18. International Conference on Machine Learning*, pp. 282–289, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001.
128. Wallach, H. M., “Conditional random fields: An introduction”, Technical Report MS-CIS-04-21, University of Pennsylvania, 2004.
129. Murphy, K., *Hidden Markov Model (HMM) Toolbox for Matlab*, 2005, <http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html>.
130. Gunawardana, A., M. Mahajan, A. Acero, and J. C. Platt, “Hidden conditional random fields for phone classification”, *in Interspeech*, pp. 1117–1120, 2005.
131. Wang, S. B., A. Quattoni, L.-P. Morency, and D. Demirdjian, “Hidden Conditional Random Fields for Gesture Recognition”, *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1521–1527, 2006.
132. Morency, L. P., *HCRF Library*, 2007, <http://sourceforge.net/projects/hcrf/>.
133. Lee, D. D. and H. S. Seung, “Algorithms for Non-negative Matrix Factorization”, *NIPS*, pp. 556–562, 2000.
134. Lin, C.-J., “Projected Gradient Methods for Non-negative Matrix Factorization”, *Neural Computation*, Vol. 19, pp. 2756–2779, 2007.
135. Hollander, M. and D. A. Wolfe, *Nonparametric Statistical Methods, 2nd Edition*,

Wiley-Interscience, 2 edition, 1999.

136. Ho, T., J. Hull, and S. Srihari, "Decision Combination in Multiple Classifier Systems", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 16, pp. 66–75, 1994.
137. Kittler, J., M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 20, pp. 226–239, 1998.
138. Gokberk, B., H. Dutagaci, A. Ulas, L. Akarun, and B. Sankur, "Representation plurality and fusion for 3-D face recognition", *IEEE Transactions on Systems Man and Cybernetics Part B*, Vol. 38, No. 1, pp. 155–173, February 2008.
139. Ong, E.-J. and R. Bowden, "Robust Lip-Tracking using Rigid Flocks of Selected Linear Predictors", *8th IEEE International Conference on Automatic Face and Gesture Recognition*, 2008.
140. Robinson, P. and R. el Kaliouby, "Computation of emotions in man and machines", *Philosophical Transactions of the Royal Society B: Biological Sciences*, Vol. 364, No. 1535, pp. 3441–7, 2009.
141. Boker, S. M., B.-J. T. Jeffrey F. Cohn and, I. Matthews, T. R. Brick, and J. R. Spies, "Effects of damping head movement and facial expression in dyadic conversation using real-time facial expression tracking and synthesized avatars", *Philosophical Transactions of the Royal Society B: Biological Sciences*, Vol. 364, No. 1535, pp. 3485–3495, December 2009.
142. Freund, Y. and R. E. Schapire, "Experiments with a New Boosting Algorithm", *International Conference on Machine Learning*, pp. 148–156, 1996.
143. Morimoto, C. H. and M. R. M. Mimica, "Eye gaze tracking techniques for interactive applications", *Computer Vision and Image Understanding*, Vol. 98, No. 1,

pp. 4–24, 2005.

144. Lucey, P., J. F. Cohn, T. Kanade, J. Saragih, and Z. Ambadar, “The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression”, *Third IEEE Workshop on CVPR for Human Communicative Behavior Analysis*, 2010.
145. Vasilescu, M. A. O. and D. Terzopoulos, “Multilinear (tensor) ICA and dimensionality reduction”, *ICA '07: Proceedings of the 7th international conference on Independent component analysis and signal separation*, pp. 818–826, Springer-Verlag, Berlin, Heidelberg, 2007.
146. Yilmaz, Y. K. and A. T. Cemgil, “Probabilistic Latent Tensor Factorization”, *LVA/ICA*, 2010.