FACIAL EXPRESSION RECOGNITION BASED ON FACIAL ANATOMY

KRİSTİN SURPUHİ BENLİ
B.S., Computer Engineering, IŞIK UNIVERSITY, 2005
M.S., Computer Engineering, IŞIK UNIVERSITY, 2007

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

IŞIK UNIVERSITY
2013

IŞIK UNIVERSITY
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING

FACIAL EXPRESSION RECOGNITION BASED ON FACIAL ANATOMY

KRİSTİN SURPUHİ BENLİ

APPROVED BY:

Assist. Prof. Mustafa Taner Eskil (Thesis Supervisor)	Işık University	___________
Prof. Yorgo Istefanopulos	Işık University	___________
Prof. Selahattin Kuru	Istanbul Kemerburgaz University	___________
Assoc. Prof. Mehmet Göktürk	Gebze Institute of Technology	___________
Assist. Prof. Devrim Akça	Işık University	___________

APPROVAL DATE: ..../..../....

FACIAL EXPRESSION RECOGNITION BASED ON FACIAL ANATOMY

Abstract

In this thesis we propose to determine the underlying muscle forces that compose a facial expression under the constraint of facial anatomy. Muscular activities are novel features that are highly representative of facial expressions. We model human face with a 3D generic wireframe model that embeds all major muscles. The input to our expression recognition system is a video with marked set of landmark points on the first frame. We use these points and a semi-automatic fitting algorithm to register the 3D face model to the subject's face. The influence regions of facial muscles are estimated and projected to the image plane to determine feature points. These points are tracked on the image plane using optical flow algorithm. We estimate the rigid body transformation of the head through a greedy search algorithm. This stage enables us to align the 3D face model with the subject's head in consecutive frames of the video. We use ray tracing from the perspective reference point and through the image plane to estimate the new coordinates of model vertices. The estimated vertex coordinates indicate how the subject's face is deformed in the progression of an expression. The relative motion of model vertices provides us an over-determined linear system of equations where unknown parameters are the muscle activation levels. This system of equations is solved using constrained least square optimization. Muscle activity based features are evaluated in a classification problem of seven basic facial expressions. We demonstrate the representative power of muscle force based features on four classifiers; Linear Discriminant Analysis, Naive Bayes, k-Nearest Neighbor and Support Vector Machine. The best performance on the classification problem of seven expressions including neutral was 87.1 %, obtained by use of Support Vector Machine. The results we attained in this study are close to the human recognition ceiling of 87-91.7 % and comparable with the state of the art algorithms in the literature.

YÜZ ANATOMİSİNE DAYALI İFADE TANIMA

Özet

Bu tezin amacı yüz ifadelerini oluşturan kas kuvvetlerinin yüz anatomisi kısıtı altında tespit edilmesidir. Kas aktivasyonları yüz ifadelerini büyük ölçüde temsil eden yeni özniteliklerdir. İnsan yüzü temel yüz kaslarını içeren üç boyutlu genel bir telkafes ile modellenmiştir. İfade tanıma sisteminin girdisi imge dizisinin ilk çerçevesi üzerinde işaretlenmiş olan nirengi noktalarıdır. İşaretlenmiş olan nirengi noktaları ve yarı–otomatik yüz modelleme algoritması kullanılarak üç boyutlu yüz modeli deneğe uyarlanır. Yüz kaslarının etki alanları tahmin edilir ve kamera düzlemine izdüşümleri öznitelik noktaları olarak belirlenir. Bu noktalar kamera düzleminde optik akış algoritması ile izlenir. Başın katı devinimi fırsatçı algoritma ile tahmin edilir. Bu aşama 3 boyutlu yüz modeli ile deneğin kafasının videonun ardışık çerçevelerinde hizalanmasını sağlar. Kamera referans noktasından kamera düzlemi boyunca ışın izleme yöntemi kullanılarak modelin düğüm noktalarının yeni koordinatları tahmin edilir. Tahmin edilen düğüm koordinatları ifade oluşumu sırasında deneğin yüzünün nasıl şekil değiştirdiğini gösterir. Modelin düğüm noktalarının bağıl hareketleri ile bilinmeyen değişkenleri kas aktivasyon seviyeleri olan artık-belirtilmiş denklemler sistemi elde edilir. Bu denklemler sistemi kısıtlı en küçük kareler yöntemi kullanılarak çözülür. Kas aktivasyonlarına dayalı öznitelikler yedi temel yüz ifadesinin sınıflandırılması probleminde kullanılır. Kas kuvvetlerine dayalı özniteliklerin temsili gücü Doğrusal Ayırtaç Analizi, Naive Bayes, En Yakın K Komşu ve Destek Vektör Makineleri sınıflandırıcıları ile gösterilir. Nötr ifade de dahil olmak üzere yedi ifadenin sınıflandırılmasında en iyi performans 87.1 % ile Destek Vektör Makineleri kullanılarak elde edilir. Bu çalışmada elde edilen sonuçlar insanın yüz ifadesi tanımadaki yetkinlik oranı olan 87-91.7 % aralığına yakın olup literatürde yer alan çalışmaların başarıları ile kıyaslanabilir durumdadır.

Acknowledgements

I am indebted to my advisor Assist. Prof. Taner Eskil for his patience, support and advices. He always shares his experiences with me. It was very big pleasure working with him.

I want to thank the members of my thesis monitoring committee: Prof. Yorgo Istefanopulos and Assist. Prof. Boray Tek for their valuable suggestions and comments. Also many thanks to my thesis jury members: Prof. Selahattin Kuru, Assoc. Prof. Mehmet Göktürk and Assist. Prof. Devrim Akça for their interest and support.

I also thank Prof. Bülent Sankur who contributed early stages of this study. I would also like to thank Prof. Mustafa Karaman for his support and assistance.

Finally, I want to give my special thanks to my family, Janet-Garabet Benli, Mari-Sirop Özçulcu, Surpuhi-Agop Benli, Linda-Melissa Gülağacı, Belinda Özçulcu, Hilda-Nerses-Maral Yağar. Thanks a lot for their unconditional love and support.

This research is part of project “Expression Recognition based on Facial Anatomy”, grant number 109E061, supported by The Support Programme for Scientific and Technological Research Projects of The Scientific and Technological Research Council of Turkey (TÜBİTAK).

To my parents…
Janet-Garabet Benli

Table of Contents

List of Tables

List of Figures

List of Symbols

v	Wireframe vertex
ϕ	Angular limit for region of influence
β	Angle of deviation
δ_A	Angular fading coefficient
δ_R	Radial fading coefficient
δ	Fading coefficient
$\mathbf{A}$	Anatomic muscle map
$\vec{\mathbf{f}}_{\mathbf{m}}$	Muscle force magnitudes
$\vec{\mathbf{f}}_{\mathbf{s}}$	Muscle forces on each vertex in each axis
x_{prp}	x-coordinate of the projection reference point
y_{prp}	y-coordinate of the projection reference point
z_{prp}	z-coordinate of the projection reference point
z_{vp}	z-coordinate of the view (camera) plane
f_p	Perspective projection
f_p^{-1}	Ray tracing
p_i^l	Facial landmarks
v_i^l	Wireframe landmarks
T	Translation
$v_{i,original}$	Wireframe vertex on original wireframe model
$v_{i,centered}$	Wireframe vertex on centered wireframe model
$v_{i,custom}^l$	Wireframe landmark on customized wireframe model
$v_{i,orig}^l$	Wireframe landmark on original wireframe model
Δv_i^l	Translation vector of wireframe landmarks
$v_{j,orig}^{nl}$	Non-wireframe landmark on original wireframe model

$d_{i,j}$	Euclidean distance
$v^{nl}_{j,custom}$	Non-wireframe landmark on customized wireframe model
$\vec{\mathbf{c}}^{t_k}_{i,j}$	Cross product at time t_k
θ	Rotation
b	Dynamics of face model
$\overrightarrow{\mathbf{F_i^{ij}}}$	3-dimensional force on vertex i in spring (i,j)
l_{ij}	Rest length of the spring
k_{ij}	Stiffness of the spring attached to vertices i and j
α_{ij}	Effective stiffness value
$\overrightarrow{\mathbf{F_i}}$	Total tensile force on vertex i
K	Stiffness matrix

List of Abbreviations

HCI	Human Computer Interaction
FACS	Facial Action Coding System
AU	Action Unit
CK Database	Cohn-Kanade Database
CK+ Database	Extended Cohn-Kanade Database
ASM	Active Shape Model
AAM	Active Appearance Model
LDA	Linear Discriminant Analysis
NB	Naive Bayes
SVM	Support Vector Machine
kNN	k-Nearest Neighbor
LLSAN	Levator Labii Superioris Alaeque Nasi
LAO	Levator Anguli Oris
ZMa	Zygomaticus Major
DAO	Depressor Anguli Oris
DLI	Depressor Labii Inferioris
HIGEM	HIgh resolution GEneric Model
NNWA	Nearest Neighbors Weighted Average Customization
GPA	Genaral Procrustes Analysis
PCA	Principal Component Analysis
H.O.T	Higher Order Terms

Chapter 1

Introduction

Communication is a word of Latin origin. Latin word *communicare* means to impart, share, or make common. According to Oxford dictionary the word communication means: the imparting or exchanging of information by speaking, writing, or using some other medium. Human beings do two-way communication through not only words but also facial expressions, gestures and posture. Mehrabian [1] reported that there are three elements in face to face communication: words, tone of voice and non-verbal behavior. Figure 1.1 depicts the percentages of importance of these elements in face to face communication as Mehrabian proposed.

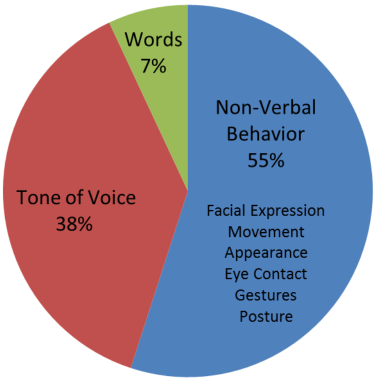


Figure 1.1: Elements of communication.

Mehrabian reported importance of verbal, vocal and visual elements of any message as 7 %, 38 % and 55 %, respectively. According to his findings tone of voice and non-verbal behavior are more effective than the “spoken words”. During communication if a person feels inconsistency between spoken words, tone of voice and non-verbal behavior, she/he prefers to believe tone of voice and non-verbal behavior (38 % and 55 %). This shows us that for effective communication, spoken words, tone of the voice and non-verbal behavior of the speaker must complement each other. Friedrich Nietzsche referred to this fact as:

> “The mouth may lie, alright, but the face it makes nonetheless tells the truth”.

Non-verbal behavior can also be called as body language that includes facial expression, eye, hand and head movements, appearance, posture etc. In person to person interaction facial expressions are frequently used. Communication over the Internet can be also a good example for explaining the importance of the facial expressions. In the beginning people expressed their feelings only by writing and send text messages to each other. Later they started to use particular symbols (smiley or sad face) in the conversation for improving the interpretation of the text. This indicates the difficulty of transferring feelings without facial expressions.

Face serves as an interface, making interpersonal communication possible. Due to this fact facial expressions constitute a popular field of research in many research domains, especially in psychology. Facial expressions are also studied in human-computer interaction (HCI). In the most general sense the goal of HCI is to analyze the emotional status of the human operator and use the obtained results in the decision processes of the machine for better adaptation to the user′s emotions. Among all gestures, facial expressions are the most direct, natural and most of the time involuntary expressions of the emotions. For that reason analysis of these expressions is one of the most prominent emotion analysis methods. Correct analysis of human faces will enable satisfactory human-computer interaction.

Latest efforts in HCI focus on automatic detection of physiological situations such as boredom, fatigue and stress. If researchers succeed in this front, unsuitable conditions of the staff who work in critical positions like pilots, drivers or system security, can be detected and accidents can be avoided ([2, 3]). If we can ever reach the success of human security experts, we may be able to detect suspicious people or deceptive facial expressions.

Research in facial expression recognition goes back to mid 1800s with the experiments of famous neurologist Duchenne de Boulogne [4]. He applied electric shocks to live subjects to observe how muscles produce facial expressions. A decade later, Charles Darwin [5] presented six universal facial expressions; anger, disgust, fear, happiness, sadness and surprise that are common among all cultures (Figure 1.2).

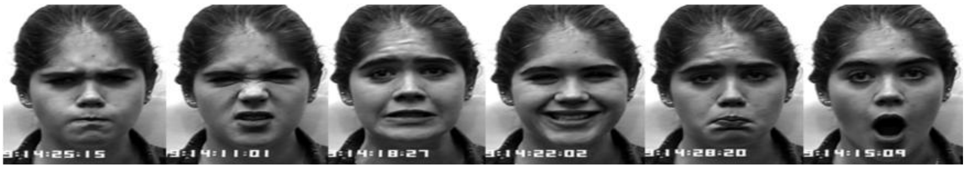


Figure 1.2: Six universal facial expressions.

About a century later, Ekman and Friesen [6] proposed a systematic method for measuring facial behaviors, namely Facial Action Coding System (FACS). They based these behaviors on action units (AUs) rather than muscle activations. Ekman and Friesen′s study drew attention to psychological studies while initiating the first steps in computer-based automated facial expression recognition.

An automatic facial expression recognition system includes three main stages. The first stage is detecting the human face in an input, which can be an image or the first frame of a video sequence. The second stage is extracting the features that discriminate facial expressions in the input. When the observation is video, we have the opportunity to make use of the dynamics of the expression by tracking the face and its features. The last stage is classification of the facial expression using the obtained numeric values of the features.

Face is a very sophisticated structure, and extracting meaningful information from it is a challenging task. Geometric, appearance-based and hybrid methods have been proposed for feature extraction. In this study we propose a new and anatomy based set of features that are derived from muscle activation levels. We demonstrate that these features are robust and representative of common facial expressions.

1.1 FACS Action Units as Features

Ekman and Friesen analyzed appearance changes on human face and defined them with action units (AUs). In FACS there is not an exact one-to-one mapping between facial muscles and AUs. There exist action units that correspond to the visual effect of a combined activity of muscles. For instance brow lowering action (AU 4) is observed as a result of the contraction of facial muscles Corrugator supercilii and Depressor supercilii (facial muscles which are located at the middle portion of the eyebrow). Conversely, one muscle may be a constituent of multiple AUs. For instance Orbicularis Oris muscle which encircles the mouth appears in the formation of six different facial actions.

Action units are annotated by the FACS experts. To become an expert, trainees attend 300 hours of training to learn rules of annotating action units and an expert can score one minute of a video clip in approximately two hours [7].

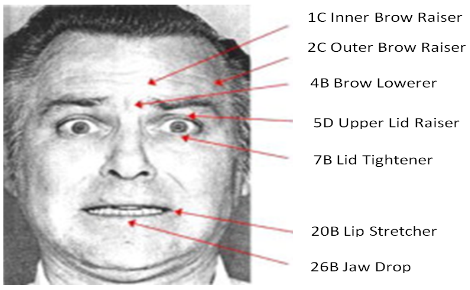
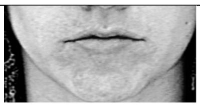
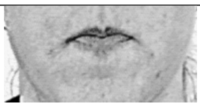


Figure 1.3: Fear expression and related action units [8].

Figure 1.3 presents an example FACS annotation. In figure subject performs Fear expression and experts determined the existence of seven AUs. AUs are labelled with numbers and letters for describing the face actions and intensity levels. Lowest and highest intensity values are represented with the letters A and E, respectively. For instance in figure subject slightly lowers his brow and experts coded this action with $4B$ where number 4 indicates brow lowerer action unit and letter B indicates slight intensity level.

The most current FACS categorization uses 46 action units for describing the facial actions (Table 1.1). Nine action units are related with the upper region of the face and 18 of them are related with the lower region [9]. There also exist additional action units for describing head and eye movements.

Table 1.1: Example action units used for facial expression recognition

Action Units (AU)	Description	Example Image	Related Expression
AU 5	Upper Lid Raiser		Surprise
AU 9	Nose Wrinkler		Disgust
AU 12	Lip Corner Puller		Happy
AU 15	Lip Corner Depressor		Sad
AU 16	Lower Lip Depressor		Happy
AU 20	Lip Stretcher		Fear
AU 23	Lip Tightener		Anger
AU 24	Lip Pressor		Anger

FACS experts also modelled facial expressions with combinations of AUs (Table 1.2). These rules form a basis for FACS-based facial expression recognition studies. However as can be seen from the Table 1.2 an action unit may appear in more than one combination. An example is AU 4, which exists in the formation of anger, fear and sadness expressions. There also exist action units (AU 5, 9, 12, 15, 16, 20, 23 and 24) that are responsible for producing single facial expression. Kotsia and Pitas [10] simplified these sets of rules in their facial expression recognition study.

Table 1.2: Description of six universal facial expression in terms of action units [11]

Expression	Action Unit Description
Anger	4+7+(((23 or 24) with or not 17) or (16+(25 or 26)) or (10+16+(25 or 26))) with or not 2
Disgust	((10 with or not 17) or (9 with or not 17))+(25 or 26)
Fear	(1+4)+(5+7)+20+(25 or 26)
Happy	6+12+16+(25 or 26)
Sad	1+4+(6 or 7)+15+17+(25 or 26)
Surprise	(1+2)+(5 without 7)+26

Majority of the current research efforts rely on extracting the FACS AUs as features [12, 13, 14]. Therefore the common focus is on more precise extraction of FACS AUs, automating FACS coding and improving the classification performance using FACS AUs. In this research, we argue that FACS has significant limitations. In the remaining of this chapter we will discuss these limitations, which serve as the motivation of our research. Next, we will discuss how we plan to address these limitations. The context of this discussion is critical in the development of the proposed approach.

1.2 Limitations of FACS AUs

We will be discussing the limitations of FACS AUs in conjunction with the anatomical structure of the face, specifically the layout of the facial muscles. Figure 1.4 depicts the layout of muscles on the human face. Note that many muscles can be involved in the progress of an action unit.

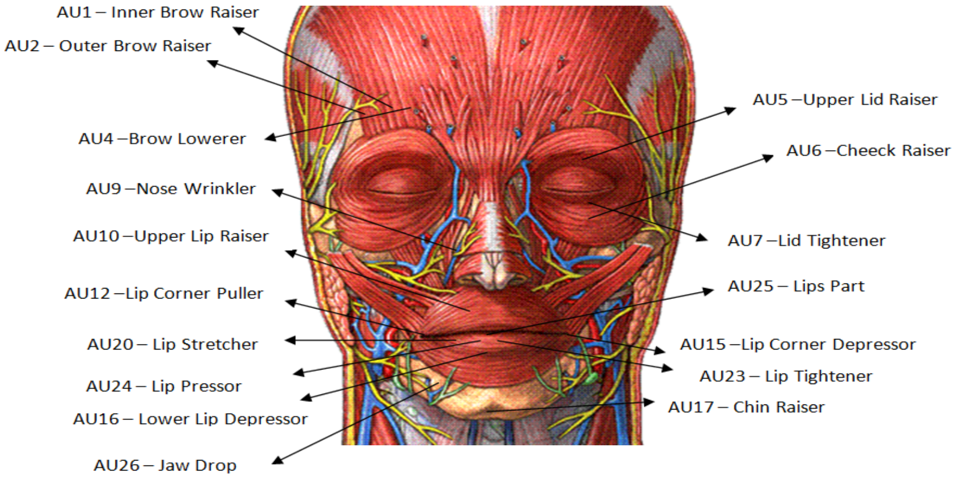


Figure 1.4: Facial muscles and FACS AUs.

In the rest of this section the limitations of the FACS AUs will be introduced.

a) Intensity scoring does not support detection of subtle changes.

FACS uses rules for scoring the intensities of AUs. When there is no evidence of an AU, the face is evaluated as neutral. AUs are scored for intensity using a five-point ordinal scale. The intensities are denoted by five letters that span the range of intensities from trace to maximum.

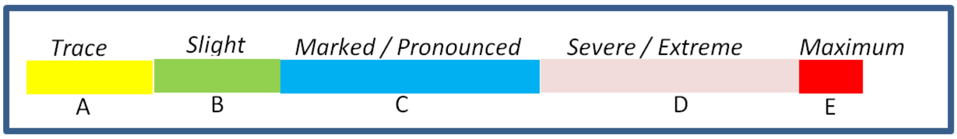


Figure 1.5: Relation between the scale of evidence and intensity scores.

Score intensity is indicated after the action unit code, i.e. 4B or 4E for indicating the intensity level of the action. Figure 1.5 indicates an unequal division of the intervals. Intensity levels C and D cover a larger activity range than the other levels. However, most of the expression activities fall in these ranges. An important limitation of FACS is the limited number of levels of quantification and their non-uniform distribution of range. Due to this scheme of scoring discrimination between intensities of expressions is a difficult problem.

Due to these disadvantages, many researchers preferred to employ a continuous scale of intensities for action units. Essa and Pentland [15], Kimura and Yachida, [16] and Lien et al.[17] defined continuous intensity levels to be used in classifying facial expressions. Tian et al. [18] used Gabor filters and artificial neural networks to quantify the action of eyelids. Pantic and Patras [19] and Valstar et al. [20] implemented a continuous scale using mid-level parameters.

b) Completely different sets of muscles can produce very similar vectoral displacements of feature points.

The FACS model focuses on facial feature points and their directions of motion. The compound effects of facial muscles may produce very similar activities of feature points. For instance, sadness and anger expressions are characterized by lip corner depressor and lip depressor actions, respectively (Figure 1.6). However, disjoint sets of muscles are active for these expressions. The sadness expression is realized by the Depressor Anguli Oris muscle (Section 3.1) whereas the muscles that are active in anger are Orbicularis Oris (facial muscle which encircles the mouth) and Incisivus Labii Inferioris (facial muscle which is located on the front of mandible).

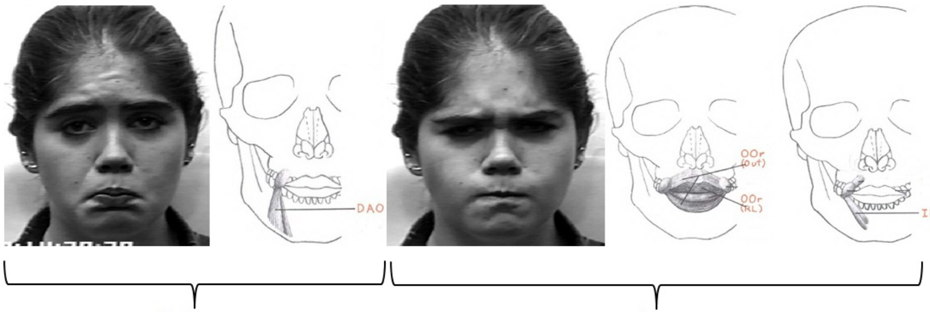


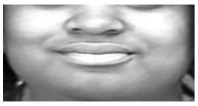
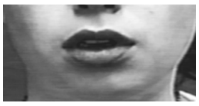
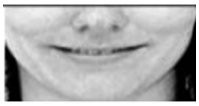
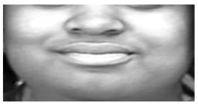
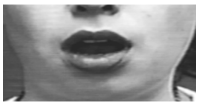
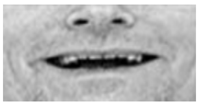
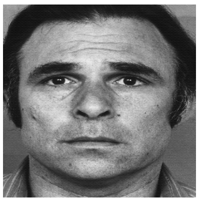
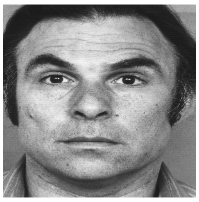
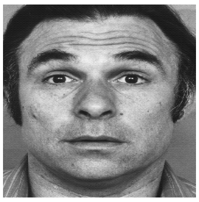
Figure 1.6: Expressions and related muscles from Goldfinger [21].

Which muscles are active during the construction of an expression is critical for expression recognition. As we stated before, there ceases to exist a one to one mapping between the vectorial displacement of a feature point (AU) and an expression. The decision for the facial action has to be supported by vectorial displacements of other feature points to obtain a sound classification.

c) It is hard to identify individual action units in compound expressions.

Action units compose facial expressions individually or in different combinations. The combinations can be additive or non-additive. Additive combinations, as the name suggests, do not affect the appearance of individual action units. Table 1.3 illustrates examples of additive action unit combinations.

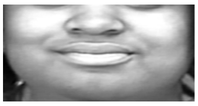
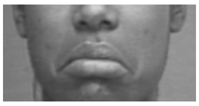
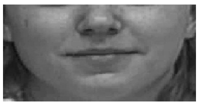
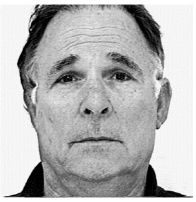
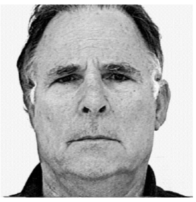
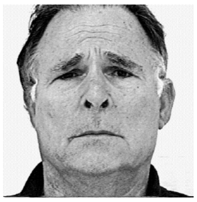
Table 1.3: Additive action units

Action Units (AU)	Action Units (AU)	Additive Combination
AU 12 Lip Corner Puller 	AU 25 Lips Parted 	AU 12+AU 25 Smiling 
AU 12 Lip Corner Puller 	AU 26 Jaw Drop 	AU 12+AU 26 Smiling 
AU 1 Inner Brow Raiser 	AU 2 Outer Brow Raiser 	AU 1+AU 2 Surprise 

The first two combinations are involved in the formation of smiling action. AU 12 lifts the corner of the lips upwards. AUs 25 and 26 present the different degrees of lip opening action. Individual presence of AUs 12, 25 and 26 can be easily detected in the additive combinations. In the last combination AUs 1 and 2 raise inner and outer parts of the eyebrows, respectively. This additive combination can be evaluated as a clue for surprise expression. As can be seen from the last frames individual action units do not change their appearance during the combinations.

In non-additive combinations, the compound effect alters the appearance of individual action units. Compound effects of action units are generally not linearly additive if they influence the same region of the face. Table 1.4 illustrates examples of non-additive action unit combinations.

Table 1.4: Non-additive action units

Action Units (AU)	Action Units (AU)	Non-additive Combination
AU 12 Lip Corner Puller 	AU 15 Lip Corner Depressor 	AU 12+AU 15 Embarrassment 
AU 1 Inner Brow Raiser 	AU 4 Brow Lowerer 	AU 1+AU 4 Sadness 

In the first example AUs 12 and 15 have almost opposite functionalities. One of them lifts the corner of the lips upwards and the other one pulls the lips downwards. According to Li and Jain [22] this combination often occurs during the embarrassment. As can be seen from the resultant frame, the individual action units are hard to recognize when the observation does not reflect a linear combination of their effects. In the second example AU 1 pulls the inner brow upward and AU 4 pulls the entire brow downward. Due to the nature of the non-additive combination, the appearance of AU 4 is modified in the combination. There may also exist additional appearance changes like wrinkles in the forehead. According to Darwin [5] resultant combination can be detected in the formation of the sadness expression.

Once AUs are compounded, it is extremely difficult to decompose an expression back to AUs unless a large rule base is made available. Tian et al. [13] also showed that approaches that focus on individual action units may fail on compound expressions.

d) It is hard to identify feature points when interpersonal variations are present.

Park and Kim [23] states that interpretations of AU′s have been difficult because of inter-person variations. As a result of this he argues that human emotions are interpreted by human experts, like psychologists, with much higher accuracy. The appearances of human faces differ remarkably between individuals.

1.2.1 Motivation for Muscle Based Features

In the paragraphs above we discussed four important limitations of the FACS based approaches. In this research we propose the following solutions to these limitations.

- We can estimate the layout of muscles by precisely fitting an anatomy-based generic wireframe model to the detected human face in observed scene.
- Once the model is customized the layout of muscles completely defines the muscular regions of influence.
- The displacement of each feature point that is carefully placed in the region of influence of muscles serves as an evidence of muscle activities.
- A set of n feature points carefully distributed in regions of influence of m muscles generates an over–determined system of equations if $n > m$. This system is solvable using convex optimization methods provided that the condition number of the coefficient matrix is low.

In this study we propose a set of new and robust features by mapping the motions of feature points due to the facial expressions to the underlying muscle forces. We aim to determine the muscle activations with high precision using a realistic and anatomy based human face model. Our novel contributions are a generic and anatomically accurate wireframe model and a set of new and robust features that are based on the facial anatomy. Our proposed system is different from the existing systems in the following aspects:

- In this study we propose and use an anatomy based high polygon wireframe model for extracting features that are based on muscle activities.
- We simultaneously track facial feature points that are distributed over muscular regions of influence on an expression video.
- We estimate head orientation and deformations on the wireframe model that constitute the observed facial expression.
- The deformation of the wireframe model is uniquely solved under the constraint of the facial anatomy to obtain muscle activation levels.
- The muscular activation levels are a set of new features, which can be used for static and dynamic classification purposes.

1.3 Problem Statement and Overview of the System

The ultimate goal of this study is to propose a set of new and robust features by mapping the deformations due to the facial expressions to the underlying muscle forces. As depicted in the figure below, we define our input as a video that contains a single human face and 32 landmark points on the first frame. We assume that the face is at a neutral expression state and is oriented towards the camera in the first frame of the video. Our approach is to dynamically track the face and its landmarks while deriving muscle activations at each state. The output of our system will be a classification of the displayed facial expression.

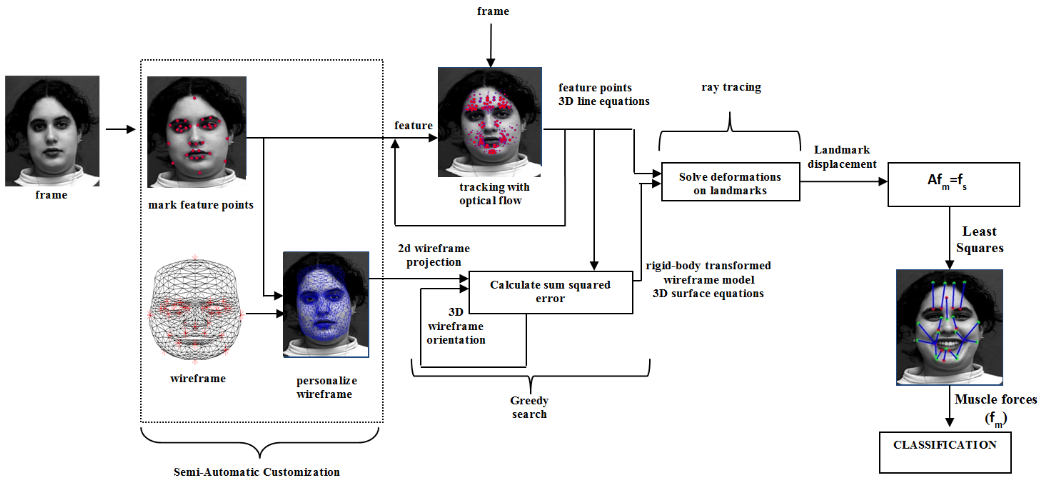


Figure 1.7: Structure of the facial expression recognition system.

The first steps of our work are to register the face image with a 3D generic face model and update the model to accommodate interpersonal variations (semi-automatic customization). In the next stage, we track the facial landmarks on the 2D video and find a rigid body transformation for the wireframe model that best describes the motion of the head (greedy search). When the orientation of the head in the 3D space is estimated, we proceed to solve for the displacements of the landmarks due to facial expression (ray tracing). These displacements will be used to solve a linear system of equations to obtain facial muscle activations. Primary focus and contribution of our research will be deriving the muscle activations that are to be used as new features. We will demonstrate the performance of these new features on a sample classification problem of seven expressions including neutral.

1.4 Organization of the Dissertation

The dissertation is organized as follows. In Chapter 2, we explore the state of the art studies in facial expression recognition field. In Chapter 3, we introduce our high polygon wireframe model and the layout of the muscles in accordance with the anatomical maps. We also present generation of the muscle map. In Chapter 4, we describe the customization of high-polygon generic wireframe model. The next stage in facial expression recognition, which is tracking of rigid body motion of the head is introduced in Chapter 5. Deformation of the wireframe model in accordance with the subject′s expression is detailed in Chapter 6. In Chapter 7 we present derivation of the stiffness matrix and discuss the constrained least squares solution to the system. We introduce the experiments and results in Chapter 8 and conclude this study in Chapter 9.

Chapter 2

Overview of Facial Expression Recognition

Human beings meet thousands of human faces throughout their lifes. They can learn and recognize faces even after many years. Over time visual characteristics of human face (ageing, glasses, beard, moustache, hair style) may change, however human beings will continue to recognize faces. It seems an effortless task for human beings to detect faces, identify them and recognize facial expressions. However computer-based automated facial expression recognition is still a challenging task. Bruce and Young [24] studied how people recognize face identity. They developed a model that is still valid in face recognition research (Figure 2.1). This model defines eight modularities for face recognition. Structural encoding generates descriptions of numerous faces. Face recognition units and person identity nodes include structural information about familiar faces. Name of the person is stored in name generation component. Cognitive system includes extra information (attractiveness, asymmetry), influencing other components. Specific facial information may be treated selectively in directed visual processing component [25]. Emotional states are recognized from facial features in expression analysis component. The observation of a speaker′s lip movements assists speech perception.

It is known fact that we mostly stare on the face of a speaker to infer the exact meaning of uttered words. In perception, visual component frequently dominates the auditory component even when the auditory information is clear; sometimes changing what we believe to hear. An intriguing illustration of this phenomenon is the McGurk effect. McGurk and MacDonald [26] demonstrated that when audio syllable /ba/ is dubbed upon videos of spoken /ga/, observers report to hear /da/. When the audio or video is presented in isolation, observers accurately identify audio /ba/ and video /ga/. This study is an excellent proof of the power of visual stimuli; they can make us believe to hear things that are merely suggested by vision.

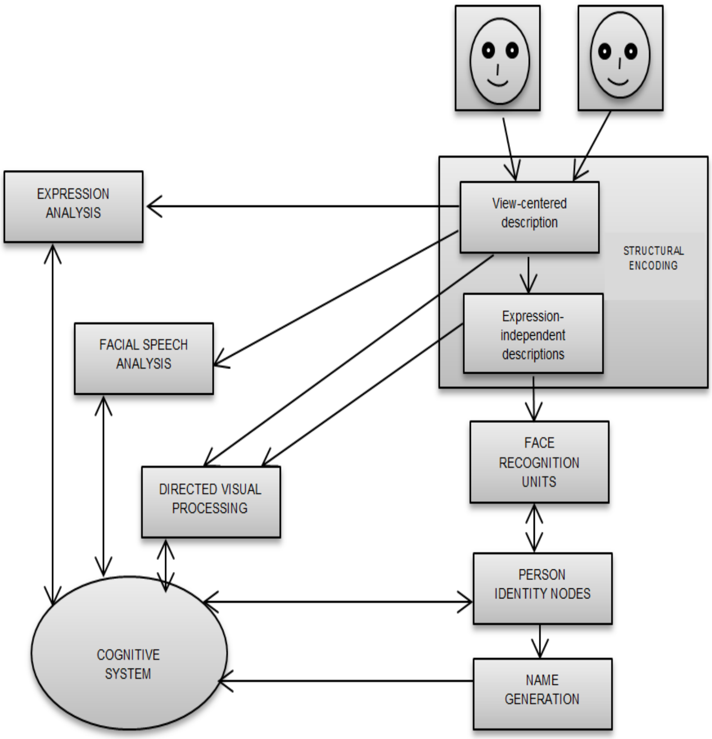


Figure 2.1: Bruce & Young model for studying faces [27].

Bruce and Young assumed that recognition of facial identities and expressions are realized separately. This idea is supported by the study of Humphreys et al. [28]. They studied with healthy and prosopagnosic participants (who can not recognize familiar faces, in some cases their faces in a mirror). Prosopagnosic participants have poor skills for recognizing familiar faces however in this study they recognized facial expressions, including the most subtle ones. They show comparable performances with the healthy participants.

Humans' recognition of facial expressions is not fully understood. For that reason this field attracted both computer scientists and psychologists. Computer scientists are working on faster and satisfactory fully automatic facial expression recognition systems while psychologists seek to reveal the mechanisms of emotions and expressions. In the next paragraphs we will introduce the stages of facial expression recognition system from the computer science perspective and explore the state of the art in this field.

2.1 Facial Expressions and Their Characteristics

Fasel and Luettin [29] define facial expressions as the contractions of facial muscles that result in deformation of facial features such as eyelids, eyebrows, nose, lips and skin texture. Most expressions are composed of a combination of these deformations. For that reason, quantifying an overall expression in terms of deformations is often a painstaking and cumbersome task.

Facial expressions are commonly described with three characteristics; location, intensity and dynamics. Intensity values of facial expressions can be measured using the geometric deformations of the face and wrinkles. For instance, the intensity level of smile expression can be measured using the magnitude of cheek, lip corner raising and wrinkles.

The dynamics (timing and duration) of facial actions are as meaningful as the actual deformations of the facial features. Static images clearly do not provide the dynamics of the face, nevertheless it is important to assess the dynamics of the facial expressions for accurate classification. The importance of the expression dynamics is addressed by researchers but there exist a limited number of studies in this topic. For instance, Messinger et al. [30] investigate and compare the timing of Duchenne (which comprises cheek raising) and non-Duchenne (which does not involve cheek raising) smiles in their study.

Facial expressions can be described with the three temporal parameters: onset (attack), apex (sustain) and offset (relaxation). Onset refers the point when expression begins to appear. Apex indicates the peak point of the expression; and Offset is the point when expression starts to fade. Until recently, these temporal parameters were coded only by human experts. Pantic and Patras [31] worked on the automatic detection of facial action units and their temporal dynamics. They propose a rule based method and achieved higher recognition rates.

Facial actions are described by their locations and intensities and sometimes dynamics. A widely used method for this purpose is the facial action coding system (FACS). FACS models facial expressions with combinations of action units (Table 1.2). First studies on automatic encoding of AUs in images of faces were reported by Bartlett et al. [32], Lien et al. [17], and Pantic et al. [33].

Current studies mostly focus on automatic coding of FACS AUs on genuine and synthetic facial expressions. Although significant success has been achieved on synthetic facial expressions, the performance of automated expression recognition systems is subpar compared to human experts, especially on genuine expressions.

2.2 Facial Expression Recognition

An automatic facial expression recognition system includes three main stages (Figure 2.2). First stage is detecting the human face from an input which may be a still image or a video. The second stage is extracting the features that describe a facial expression in the observed input. When the observed input is a video, we have the opportunity to make use of the dynamics of the expression by tracking the face and its features. The last stage is classification of the facial expression using the obtained numeric values of the features. In the following paragraphs we will describe these facial expression recognition stages.

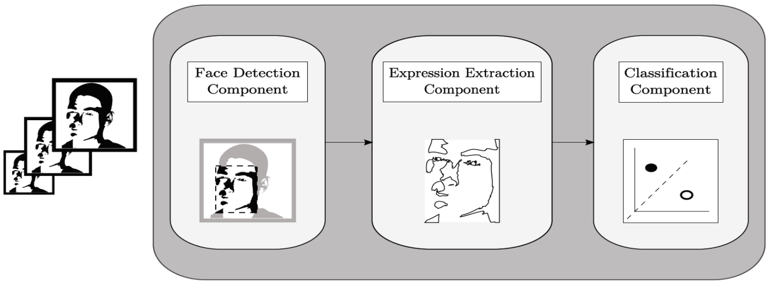


Figure 2.2: Facial expression recognition system [34].

2.3 Face Databases

Most facial expression recognition algorithms need a large collection of training and test data. For this purpose databases have been developed. Table 2.1 shows an overview of the existing databases that can be used in automatic facial expression analysis. These databases and their important properties will be introduced below.

Table 2.1: Overview of the existing Face Databases

Name	Number of Subjects	Number of Expressions	Type of Expression
Cohn-Kanade [35]	97	6	Posed
Extended Cohn-Kanade [36]	123	7	Posed and non-posed
PICS [37]	35	3	Posed
JAFFE [38]	10	7	Posed
AR [39]	126	4	Posed
PIE [40]	68	4	Subtle
Muti-PIE [41]	337	6	Subtle
MMI [42], [43]	52	9	Posed and Spontaneous
RU-FACS [44]	100	Not mentioned	Spontaneous
UT-Dallas [45]	284	11	Spontaneous

Cohn-Kanade (CK or DFAT) Database: Developed in the Carnegie Mellon University, it is the most widely used database in facial expression analysis research. This database has 2105 image sequences and contains both single action unit examples and emotional expressions like joy, surprise, sadness, disgust, anger and fear. But it has two limitations. First, each recording ends at the apex of the shown expression and this limits analysis of facial expression temporal activation patterns to onset (attack), apex (sustain). The second limitation of this database is that the date/time stamps of the recordings of videos are displayed over the chin of the subject. As a result of this, the appearance changes of the chin become less visible, making it difficult to track the deformations of the chin.

Extended Cohn-Kanade (CK+) Database: It is an extension of the CK database. The dataset contains 593 sequences from 123 subjects, with increased number of sequences (additional 107 sequences) and subjects (additional 26 subjects). As in the original CK, the image sequence starts with the neutral expression and ends at the peak of the facial expression. FACS experts annotated 327 of the 593 sequences according to the peak frames of the expressions. Subjects performed 7 facial expressions; 6 basic emotions and contempt.

PICS Database: It is an image database developed in the Stirling University and stands for Psychological Image Collection at Stirling. It contains grayscale and color images of both female and male subjects. There are multiple views of subjects such as profile view and frontal view. Subjects performed smile, surprise and disgust expressions.

JAFFE Database: JAFFE is a project of Kyushu University and ATR Human Information Processing Research Laboratory and it stands for the Japanese Female Facial Expression database. It contains 219 static images. 10 Japanese females display 7 facial expressions; 6 basic emotions and a neutral expression. Each image is rated by 60 Japanese subjects.

AR Database: It is created in Computer Vision Center (CVC) of Universitat Autonoma de Barcelona. It contains over 4000 color images of 126 subjects (70 male and 56 female). Subjects performed neutral, anger, smile and scream expressions. Also, experiments are done under varying illumination conditions. Few images are collected with subjects wearing sun glasses and scarf.

PIE Database: It is the image database of Carnegie Mellon University Robotics Institute and stands for CMU Pose, Illumination, and Expression (PIE) database. It contains 41368 images of 68 people. The images of participants are collected under 13 different poses, 43 different illumination conditions and with 4 different expressions including neutral, smile, and blink. The images include talking subjects.

Multi-PIE Database: The original PIE database has few limitations such as limited number of subjects, single recording session and few number of expressions. Multi-PIE database is an improved version of PIE database with increased number of subjects (337 people), recording sessions (increased to 4 sessions) and number of expressions (increased to 6 expressions). Subjects performed neutral, smile, surprise, squint, disgust and scream actions. Images are taken under 15 view points and 19 different illumination conditions.

MMI Database: It is developed in Man-Machine Interaction (MMI) group of Delft University of Technology. It consists of two parts. First part of MMI contains deliberately displayed facial expressions; there are over 4000 videos and 600 static images in this category. This database includes facial expressions of single AU activation, multiple AU activations, and emotions such as anger, disgust, fear, happiness, sadness, surprise, scream, boredom and sleepiness. FACS coding is done by two certified coders. Second part of MMI contains 65 videos for spontaneous facial expressions. This part is also coded by two certified coders.

RU-FACS Database: RU-FACS database contains spontaneous facial expressions that are coded by FACS experts. Subjects participate a false opinion paradigm. They filled a questionnaire and attempted to persuade an interviewer (retired police and FBI) that he/she is telling the truth. The dataset contains 100 subjects, 33 of which are annotated by FACS experts.

UT-Dallas Database: UT-Dallas is collected in the University of Texas. It contains static images and video sequences. There are 284 subjects performing 11 different expressions (happiness, sadness, fear, disgust, anger, puzzlement, laughter, blank stare, surprise, boredom, disbelief). It includes videos of more than one expression such as subject starting her/his performance with puzzled expression, changing the expression to surprise or disbelief and concluding with laughter. FACS coding of this database is not available.

There are also a significant number of 3 dimensional data sets available for research. A brief summary of a subset of existing 3 dimensional face data sets is proved in Table 2.2.

Table 2.2: 3 dimensional face data sets. Variations include (E)xpression, (I)llumination, (O)cclusion, (P)ose, (S)peech.

Database Name	Subjects	Resolution	Variation
3DRMA [46]	120	240 x 320	P
Bosphorus 3D [47]	105	1128 x 1374	E, O, P
BU-3DFE [48]	100	1049 x 1329	E, P
BU-4DFE [49]	101	1040 x 1329	E, P
CASIA [50]	100	640 x 480	E, I, P
CAS-PEAL [51]	1040	360 x 480	E, I, P
FRGC-v2.0 [52]	466	1704 x 2272	E, I
GavabDB [53]	61	240 x 320	E, P
Max Plank Inst. [54]	200	786 x 576	E, P
ND 2006 [55]	888	240 x 320	E, P
Photoface [56]	453	1280 x 1024	E, I
Texas 3DFRD [57]	284	720 x 480	E, P
XM2VTS [58]	295	720 x 576	R, S
York [59]	350	240 x 320	E, P

2.4 Face Detection

The initial step of facial expression recognition study is face detection. Face detection is the special case of object detection. Object detection methods can achieve successful results on simpler objects. However face detection is significantly more complex than detection of simple and rigid objects. Identifying human faces in an observed scene was an important challenge for algorithmic approaches for decades (Figure 2.3).

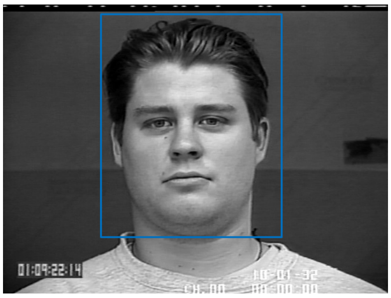


Figure 2.3: Face Detection.

There are a significant number of studies in this topic. The problem of face detection is now considered to be solved with works of Rowley and Viola Jones. Rowley et al. [60] used neural networks for detecting upright, frontal views of faces. They examined small windows of gray-scale images and detected face and non-face patters. Schneiderman and Kanade [61] proposed a statistical method for detection of faces with out-of plane rotation. Yang et al. [62] presented a method that uses SNoW (Sparse Network of Winnows) learning architecture. Romdhani et al. [63] used non-linear support vector machines. Most commonly used face detection algorithm was proposed by Viola and Jones [64]. In this method cascade of rectangular box based classifiers are constructed and trained by AdaBoost.

2.5 Feature Extraction and Tracking

After the face detection stage, the next step is extracting discriminative information for recognizing expressions. Face is a very sophisticated structure, and extracting meaningful information from it is a challenging task. Geometric, appearance-based and hybrid methods have been proposed for feature extraction. The geometry of facial feature points and deformations on the skin are important visual cues for facial expression recognition. Appearance based features are derived from the texture of a facial image. Model based methods are proposed to derive a mathematical model of variation modes of geometric or appearance based features. In this section we will discuss these methods and address to their connections with our approach.

Geometric features are driven from the facial components (e.g. eyes, mouth) and the pixel coordinates of facial fiducial points (e.g. corners of the eyes, mouth). Pantic and Patras [19] proposed a method for detecting FACS AUs and their temporal dynamics from profile-view image sequences. They tracked a set of facial points using particle filtering, observed the position changes of the tracked points, and calculated the relative changes in feature coordinates, which they call as mid-level parameters. These mid-level parameters are used to determine action units.

Valstar et al. [20] presented a method for discriminating the posed and spontaneous facial expressions by analyzing the brow actions. They used a semi-automatic method for initializing feature points on input face image. Feature points are tracked with two standard tracking algorithms and mid-level feature parameters are calculated. Gentle Boost method is used to select more informative features and the expressions are classified with a probabilistic decision function.

Valstar and Pantic [65] employed an automatic feature extraction system. Facial feature points are automatically detected with Gabor feature-based classifiers and tracked with Particle Filtering with Factorised Likelihoods algorithm. Polynomial mid-level parameters are classified with combined HMM and SVM.

Seyedarabi et al. [66] presented a facial expression analysis and synthesis system. They manually marked 14 facial feature points in the first frame and estimated motion of these points with an improved cross-correlation based motion tracking algorithm. They extracted features (width of eye and mouth, height of eyebrows, openness of mouth, nose tip-lip corner distance and eye-cheek distance) from facial feature points and classified them with probabilistic neural network (PNN) classifier.

Kotsia and Pitas [10] proposed a facial expression recognition system based on geometric deformation features. They initialized Candide grid with a semi-automatic fitting approach and tracked the model in consecutive frames by a pyramidal variant of Kanade-Lucas-Tomasi tracker [67]. They calculated geometrical displacements of grid nodes between first and last frames of the expression. Extracted features are classified with multi-class SVM.

Lu and Zhang [68] used optical flow to track feature points in subsequent frames. They calculated displacements of feature points and classified facial expressions to one of the basic emotions with discriminative analysis of canonical correlations.

Valstar and Pantic [69] introduced a fully automatic expression analysis method to recognize action units and their temporal characteristics. They used Gabor-feature-based boosted classifier for detecting facial feature points. They utilized particle filtering for tracking these points in consecutive frames and applied a combination of GentleBoost, SVM and HMM for classifying action units and their temporal dynamics.

Appearance based methods deal with the texture of the facial skin including wrinkles, bulges and furrows. Valstar et al. [70] proposed an AU recognition system based on Multilevel Motion History Images. They examined the performances of temporal templates in AU detection with a combined (kNN and rule based) and SNoW classifier.

Guo and Dyer [71] manually marked fiducial points on the face image, applied Gabor filters, used the amplitude values of fiducial points as features and compared the performances of several classification methods.

Bartlett et al. [72, 73] also used Gabor filters and presented expression recognition results of different machine learning methods. Whitehill and Omlin [74] proposed Haar features for FACS AU recognition. They also examined the performance differences between introduced Haar features with Adaboost method and Gabor features with SVM approach and obtained similar recognition results. Jiang et al. [75] examined the local binary pattern descriptors for facial action unit detection.

Zhang et al. [76] classified expressions using two types of features: geometric positions and Gabor wavelet coefficients of the fiducial points. These features are given as input to a two-layer perceptron. They demonstrated that Gabor wavelet coefficients achieves better recognition performance than geometric coordinates.

Chen et al. [77] combined both geometric and appearance based features in their facial expression recognition study. They used extended active shape model to extract facial feature points. Facial feature point displacements and local texture differences between the neutral and peak frames are computed. Obtained hybrid features are classified with SVM.

Model based approaches can be an alternative to appearance based approaches. Typical examples are Active Shape Model [78] and Active Appearance Model [79]. This is a parametric deformable model. They are used to create models of human hearts, hands and faces. A statistical shape model of the face object is built using a set of training examples. Pose and shape parameters are iteratively modified for a better fit.

Gang et al. [80] introduced a geometric feature extraction method for facial expression recognition study. They applied ASM based method for detecting the coordinates of facial fiducial points and computed distances between each facial fiducial point and the center of the gravity of the face shape. Geometric deformation in the neutral and peak frames are extracted and classified with SVM.

Active Appearance Model (AAM) is also proposed by Cootes et al. [79] for matching a generic face model to input face image. AAM combines the statistical model of the shape and the gray-level appearance of the object of interest. The synthesized model is projected onto the face image and matching is done iteratively.

Lucey et al. [81] derived features based on Active Appearance Model (AAM) and employed them for facial action recognition task. Lucey et al. [36] introduced Active Appearance Model based system to classify seven emotions. They tracked the face and extracted geometric shape and canonical appearance features. They demonstrated classification results of the individual and combined features. According to the results combined features reached higher accuracy than individual features.

2.6 Facial Expression Classification

In its early stages, research on automatic facial expression recognition focused on static images. Studies in this context can be grouped in two: facial action unit (AU) detection and facial expression recognition. Table 2.3 presents recognition of facial action units (AUs) on static images. Studies focused on detecting action units alone or in combinations. Tian et al. [13] grouped AUs as upper and lower face and detected them using neural networks. Pantic and Rothkrantz [11, 82] applied a rule based method and detected large numbers of AUs with high accuracy.

Table 2.3: AU recognition studies on static images. ^{u} Upper-face FACS AU. ^{l} Lower-face FACS AU.

Study	**Methodology**	**Success**	**FACS AUs**	**Database**
Tian et al. [13]	NN	95.4 %	6^u	CK
		95.6 %	10^l	Ekman & Hager [83]
Pantic & Rothkrantz [11]	Rule based	89.0 %	31	Pantic & Rothkrantz [11]
Pantic & Rothkrantz [82]	Rule based	86.0 %	32	Pantic & Rothkrantz [82]

As aforementioned in Table 1.2, most facial expressions can be described with a combination of action units. Therefore facial action unit detection can also be viewed as a preliminary stage of facial expression recognition. There are quite a few studies that extract AUs and utilize them for classifying basic facial expressions.

Table 2.4 presents selected facial expression recognition studies on static images. In these studies artificial neural networks [76], linear discriminant analysis [84] and rule based classifiers [11] among others are applied on single frames. In static analysis, time information is not used, therefore the dynamics of facial expressions are ignored. The experimental studies showed that, formation stages of facial expressions are as important as the peak appearance of facial expressions [85].

Table 2.4: Selected facial expression recognition studies on static images.

Study	**Methodology**	**Success**	**Emotion**	**Database**
Zhang et al. [76]	NN	90.1 %	6	JAFFE
Lyons et al. [84]	LDA	75-92 %	6	JAFFE Ekman & Friesen [86]
Fellenz et al. [87]	PCA+MLP	60.0 %	4	CMU
Pantic & Rothkrantz [11]	Rule based	91.0 %	6	Pantic & Rothkrantz [11]
Littlewort et al. [88]	SVM	92.0 %	7	CK
Wen & Huang [89]	NN+GMM	71.0 %	6	CK

Recent FACS AU detection and facial expression recognition studies are done with image sequences. Table 2.5 presents AU classification studies on image sequences. In latest studies number of detected AUs are increased. However none of the studies can recognize the full set of FACS AUs. The largest AU set is utilized by Pantic and Patras [19]. Bartlett et al. [44] recognized spontaneous facial actions with high accuracy. There also exist few studies that are focused on detecting temporal dynamics (onset, apex, offset stages) of the facial action units [19, 69].

Table 2.5: Dynamic facial action unit classification. Explanations of the abbreviations: ^{sf}: shape features, ^{af}: appearance features, ^{cf}: combined features

Study	**Methodology**	**Success**	**FACS AUs**	**Database**
Lien et al. [17]	HMM	85.0 %	3	Frank & Ekman [90]
	PCA+HMM	93.0 %	3	
	Variation finding+ HMM	85.0 %	4	
Bartlett et al. [91]	SVM+HMM	98.0 %	2	Frank & Ekman
		70.0 %	3	
Cohn et al. [92]	Rule Based	57.0 %	3	Frank & Ekman
Moriyama et al. [93]	Rule Based	98.0 %	2	Frank & Ekman
Bartlett et al. [44]	Adaboost	91.0 %	20	CK Hager & Ekman [83]
	Adaboost	93.0 %	19	RU-FACS
Pantic & Patras [19]	Rule Based	87.0 %	27	MMI
Lucey et al. [36]	SVM	90.0 % ^{sf}	17	CK+
		91.4 % ^{af}	17	
		94.5 % ^{cf}	17	
Valstar & Pantic [69]	GentleSVM & HMM	95.3 %	22	MMI
		91.7 %	22	CK

Table 2.6 shows a categorization of recent studies in facial expression recognition. As can be seen from the table most of the studies included neutral expression in their emotion sets. Sebe et al. [94] created an authentic facial expression database and applied several classifiers to recognize neutral, joy, surprise and disgust emotions. They also utilized Cohn and Kanade database. Park and Kim [23] introduced a motion magnification based method to recognize subtle facial expressions. They used SFED2007 database that contains four facial expressions (neutral, smile, surprise and anger) of 20 subjects and obtained the highest accuracy with SVM classifier.

Table 2.6: Recent facial expression recognition studies and their results. Explanations of the abbreviations: ^{pd}: person dependent experiment, ^{pi}: person independent experiment,^{sf}: shape features, ^{af}: appearance features, ^{cf}: combined features, ^{n}: neutral class.

Study	**Methodology**	**Success**	**Emotion**	**Database**
Bartlett et al. [14]	Adaboost	90.1 %	7^n	CK
	SVM	88.0 %		
	AdaSVM	93.3 %		
	LDA_{PCA}	80.7 %		
Sebe et al. [94]	NB	75.6 %	4^n	CK
	C4.5	83.9 %		
	SVM	75.4 %		
	kNN	93.0 %		
Sebe et al. [94]	NB	91.5 %	4^n	Sebe et al. [94]
	C4.5	91.6 %		
	SVM	86.8 %		
	kNN	95.6 %		
Kotsia et al. [95]	DNMF	74.3 %	7^n	CK
	SVM	84.8 %		
	MRBF NN	92.3 %		
Park & Kim [23]	Nearest Neighbor	71.9 %	4^n	SFED2007 [96]
	PNN	74.4 %		
	LDA	79.4 %		
	QDA	77.5 %		
	SVM	88.1 %		
Sung & Kim [97]	Layered GDA	98.2 %	4^n	Sung & Kim [97]
	Single layer GDA	96.7 %		
Gang et al. [80]	SVM	89.5 % ^{pd}	7^n	JAFFE
		68.5 % ^{pi}		
Lucey et al. [36]	SVM	50.0 % ^{sf}	7	CK+
		66.7 % ^{af}		
		83.3 % ^{cf}		
Lu & Zhang [68]	Canonical Correlations	90.0 %	6	CK
Chen et al. [77]	SVM	95.0 %	7	CK+

Sebe et al. [94] and Park and Kim [23] worked on a simpler classification problem. Both of them classified 4 emotions including neutral and excluding fear and sadness. This selection is done in accordance with the complexity of emotions.

Emotions may differ in degree of complexity in terms of the number of muscles involved. From this point of view, happiness can be described as a simple emotion, whereas fear is one of the most complex emotions.

Kotsia et al. [95], Lucey et al. [36] and Chen et al. [77] utilized hybrid features for classifying facial expressions. Kotsia et al. [95] classified seven emotions including neutral with texture-based (74.3 %) and shape-based features (84.8 %). They also introduced the recognition results for combined features (92.3 %). Lucey et al. [36] classified seven emotions with shape-based (50 %), appearance-based (66.7 %) and combined features (83.3 %). Obtained results showed that hybrid features achieve better recognition performances than individual features. Chen et al. [77] also combined geometric and appearance based features and classified seven basic emotions with high accuracy (95 %). In literature instead of using geometric and appearance-based features alone, researchers proposed to combine them. Obtained results also support the effectiveness of this idea.

Gang et al. [80] applied person-dependent and person-independent tests to evaluate performance of their approach. In person-dependent experiment a subject′s image samples of the same expression can be appear both in training and testing data set. In person-independent experiment a subject′s image samples of the same expression can appear in training or testing data but not both of them. In both of the experiment approaches same features, dataset and classifier are used but different recognition results are obtained. As expected, person dependent test showed better performance than the independent. However it is important to note that the actual performance of a system must be considered on person independent test.

Many studies in the literature do not clearly introduce the testing procedures. Most of the times distinct observations from the same expression of a specific subject may appear both in training and testing data sets. Recognition performances of those studies that utilize person-dependent approach can be misleading.

Another challenge in facial expression recognition study is a lack of standardized performance evaluations. Studies are done using different databases and different facial expressions; therefore it is hard to pass judgement on the most successful approach.

In dynamic analysis acceptable success rates were achieved only when basic emotions were targeted. These results suggest that there are open issues for research in facial expression analysis. In the past ten years research in facial expression recognition is increasingly getting attention. However each advancement in this field brings about new and challenging research problems. One issue open for debate is how to deal with an increased number of classes. Another open question is how to scale the intensity of facial expressions, if any. In the following chapters we will introduce our approach to the facial expression recognition problem.

Chapter 3

Face Model

There are more than 600 muscles in human body and approximately 52 of them form facial expressions [98]. Muscles are responsible for posture, body movements and facial expressions. Facial muscles produce over 10,000 facial expressions 3,000 of which can be recognized from other people. In smiling action almost 15 facial muscles become active. Our approach in this study capitalizes on the anatomical structure of the human face. We propose to identify the activation levels of facial muscles in the progress of an expression by observing the displacements of feature points that are distributed over a region. Obviously the actual forces applied by the facial muscles are not observable unless electrical activity sensors such as those employed in electroencephalograph (EEG) are utilized. However, it is possible to derive them given an accurate model of the face, the physical properties of skin and muscles, and the observed displacements of carefully selected feature points. Hence, having an accurate anatomical model of the human face is critical for our research. In this Chapter we will discuss face anatomy, which will serve as a reference for the development of a face model. This chapter is organized as follows: In Section 3.1 we introduce major facial muscles. In Section 3.2 we present existing face models and in Section 3.3 we describe our high polygon face model (HIGEM). In Section 3.4 we introduce placement of facial muscles on HIGEM face model and in Section 3.5 we present generation of muscle map.

3.1 Facial Anatomy

All fiber muscles on the human face have an origin (**O**) and an insertion (**I**) point. The origin point of a muscle is the point where it attaches to a skull. Since the muscle is anchored to the bone, origin point does not move by the contraction of the other muscles. Insertion point is distinct from the origin point, it attaches to the skin.

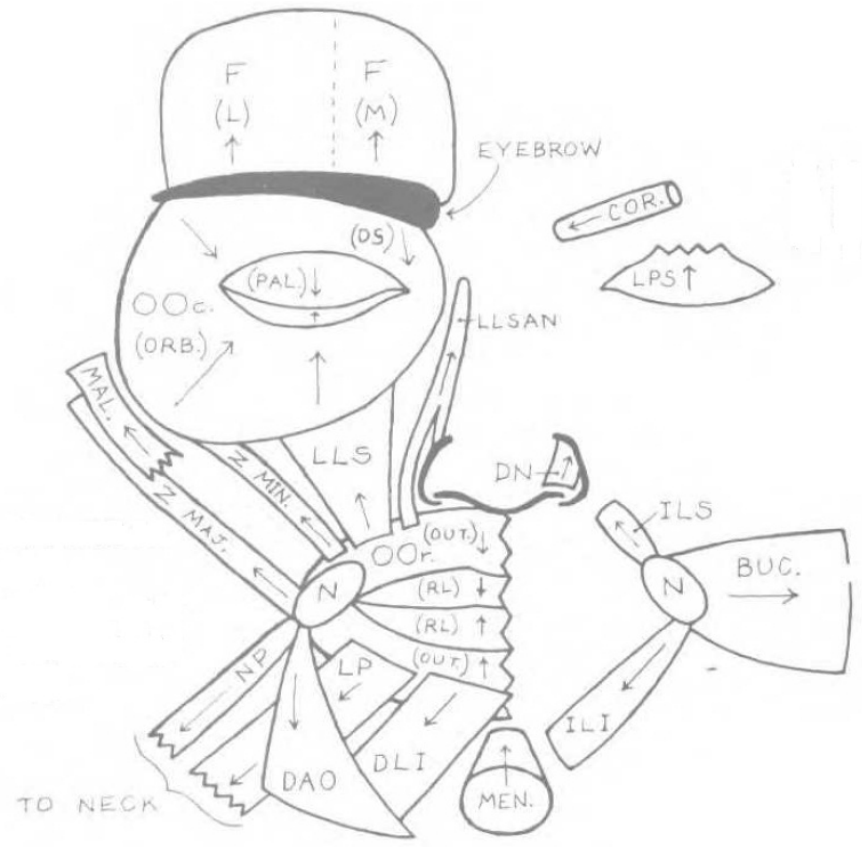


Figure 3.1: Facial muscles and their directions [21].

In the following paragraphs we introduce major facial muscles, their functionalities and layout on face. Muscles that are discussed in this section will be implemented on the HIGEM wireframe model.

1. **Frontalis:** It covers the forehead. It is composed of medial and lateral fibers. The layout of the muscle with its origin and insertion points is shown in Figure 3.2.

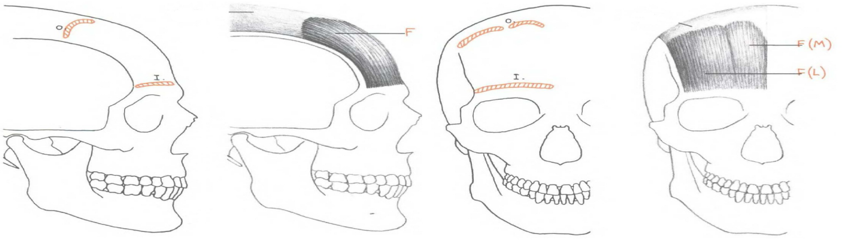


Figure 3.2: Origin and insertion points of Frontalis [21].

These two constituents of the Frontalis muscle may act together and alone. When they become active they pull different regions of the eyebrows upwards. For instance, medial Frontalis raises the medial end whereas lateral Frontalis raises the middle and lateral ends of the eyebrow. When both parts become active, they express surprise and fear expressions (Figure 3.3). Medial Frontalis is active in the formation of the sadness expression.

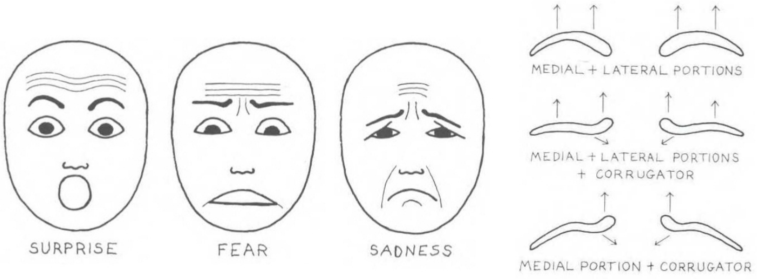

SURPRISE
FEAR
SADNESS
MEDIAL + LATERAL PORTIONS
MEDIAL + LATERAL PORTIONS + CORRUGATOR
MEDIAL PORTION + CORRUGATOR

Figure 3.3: Frontalis activity [21].

2. **Procerus or Pyramidalis Nasi:** It covers the nasal bones and the skin between lower region of the forehead and between eyebrows. It draws the medial ends of the eyebrows downward. It forms anger expression and

contributes to disgust expression. The layout of the muscle with origin and insertion points is shown in Figure 3.4.

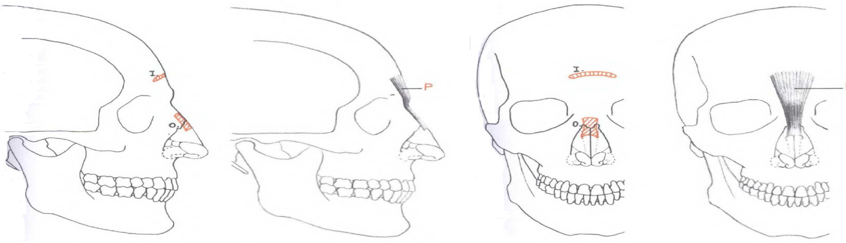


Figure 3.4: Origin and insertion points of Procerus [21].

3. **Levator Labii Superioris Alaeque Nasi (LLSAN):** It covers the upper lip and lateral side of the nasal bones. It raises upper lip upward and enlarges wing of the nose. It forms the snarl expression. Since this expression was often performed by Elvis Presley, this muscle is also known as "The Elvis muscle". The layout of the muscle with origin and insertion points is shown in Figure 3.5.

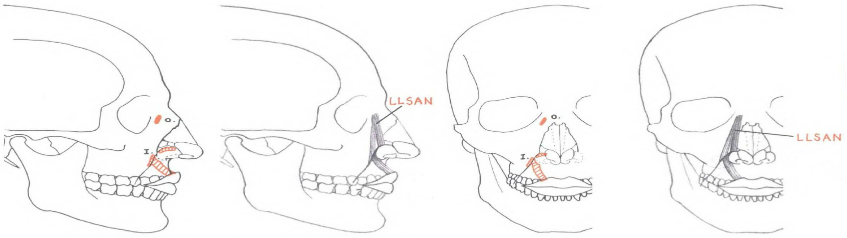


Figure 3.5: Origin and insertion points of LLSAN muscle [21].

4. **Levator Anguli Oris (LAO):** It is positioned below the lower part of the orbit. It raises the corner of the mouth. It is a constituent of the smile expression. The layout of the muscle with origin and insertion points is shown in Figure 3.6.

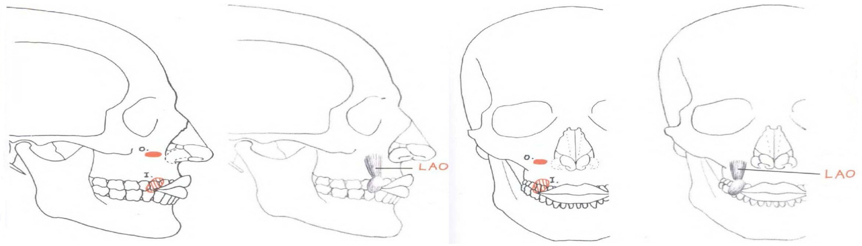


Figure 3.6: Origin and insertion points of LAO muscle [21].

5. **Zygomaticus Major (ZMa):** It covers the cheek bone and the muscles near the corner of the mouth, lifting the corner of the lips upwards. It forms expressions such as happiness, joy-smiling and laughing. The layout of the muscle with origin and insertion points is shown in Figure 3.7.

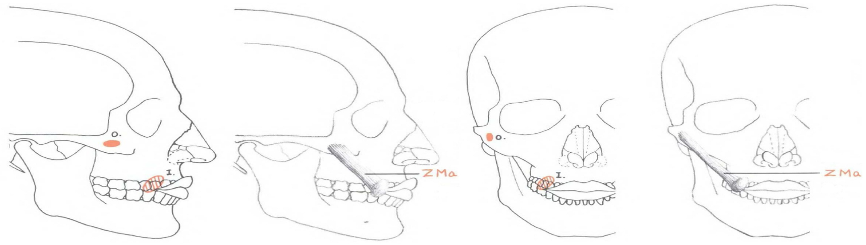


Figure 3.7: Origin and insertion points of ZMa muscle [21].

6. **Risorius:** It is positioned around the mouth. It pulls the corner of the mouth backward and outward. It contributes to the expression of happiness. The layout of the muscle with origin and insertion points is shown in Figure 3.8.

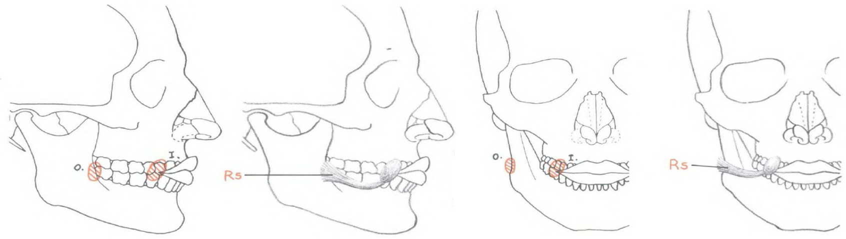


Figure 3.8: Origin and insertion points of Risorius [21].

7. **Depressor Anguli Oris (DAO):** It is located in the area starting from the mandible and ending with the corner of the mouth. It pulls the mouth

downward and is a major constituent of the sadness expression. The layout of the muscle with origin and insertion points is shown in Figure 3.9.

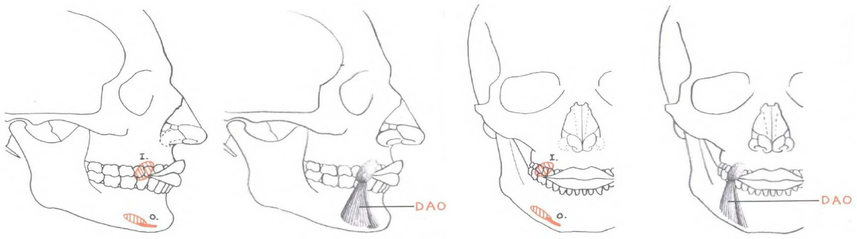


Figure 3.9: Origin and insertion points of DAO muscle [21].

8. **Depressor Labii Inferioris (DLI):** It is positioned in the jaw area. It pulls the lower lip downwards. It forms expressions that are primarily used in speaking. The layout of the muscle with origin and insertion points is shown in Figure 3.10.

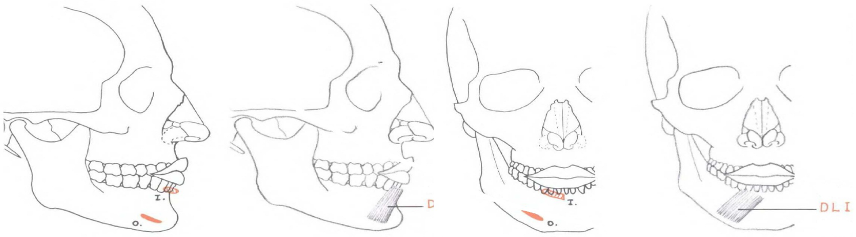


Figure 3.10: Origin and insertion points of DLI muscle [21].

9. **Mentalis:** It winds up the chin. It forms expressions such as sadness, grief, anger, disdain, and disgust. The layout of the muscle with origin and insertion points is shown in Figure 3.11.

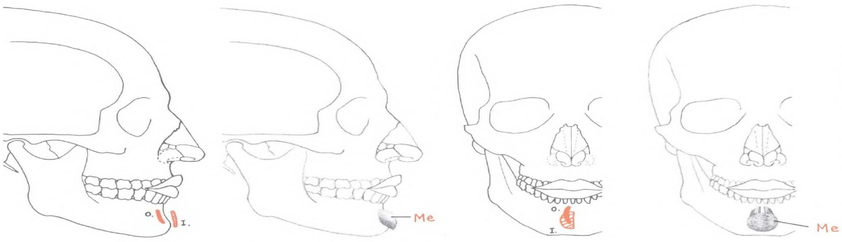


Figure 3.11: Origin and insertion points of Mentalis [21].

3.2 Types of Face Models

The first stage of our study is developing a 3D generic wireframe model according to the anatomy of the human face. The Candide model [99] has been widely used in the literature for modelling and animation of the faces. It was developed in Linköping University for model-based coding of human faces. The original Candide model contained 75 vertices and 100 triangular faces. After its first introduction, modified versions were created with increased vertex and polygon numbers (Figure 3.12).

A common version of Candide also defined 11 action units. The vertices of the Candide model are carefully selected to be able to track, identify, or synthesize the prominent deformations of the face. As such, Candide is not intended to recognize subtle expressions or life-like facial expression simulations. Such studies of the face require a more precise model through increased number of polygons.

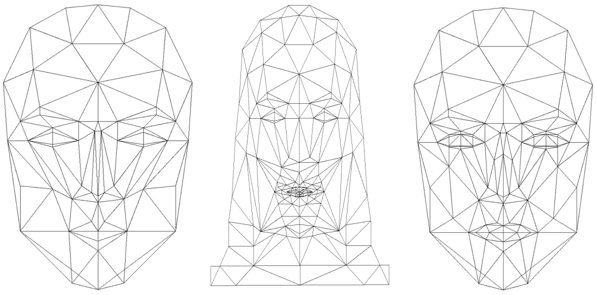


Figure 3.12: Different versions of Candide [99, 100, 101] .

There also exist other face models in the literature. The first 3D parametric face model is developed by Parke [102]. Parke constructed a face model (Figure 3.13) for expression and speech animation that contains about 400 vertices and 300 polygons.

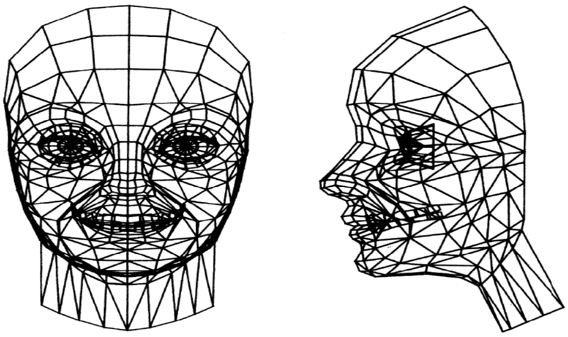


Figure 3.13: Face model proposed by Parke [102].

Waters [103] proposed another face model for animating facial expressions that consists of 512 vertices and 878 polygons. This polygonal face model is also utilized in the studies [104, 105, 106]. Erol [106] initialized nine symmetric pairs of muscles on the generic face model to represent facial deformations. Upper and lower part of the face contains 10 and 8 muscles, respectively. Figure 3.14 depicts face model and utilized facial muscles. Note that a few major muscles which appear in mandible like Mentalis and DLI are not included in face model.

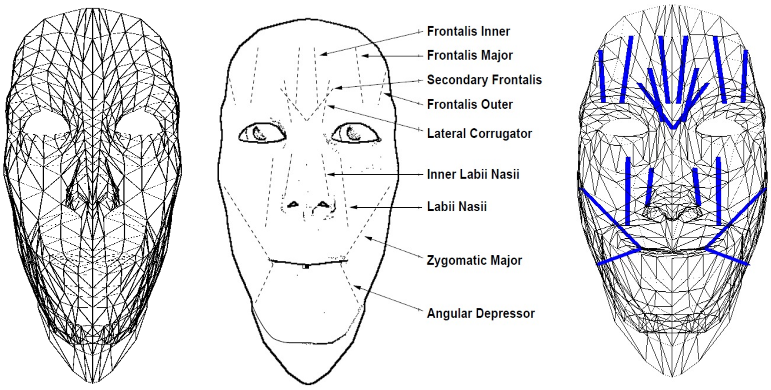


Figure 3.14: Face model and muscle layout proposed by Waters [103].

Essa and Pentland [15, 107] utilized extended face model of Platt and Badler [108] in facial expression recognition study. The introduced face model consists of 1226 nodes and 80 facial regions. They [107] used the method of Waters and Terzopoulos [109] and muscle data of Pieper et al. [110] for constructing an anatomical muscle model of the face. Figure 3.15 depicts face model and defined 22 muscles.

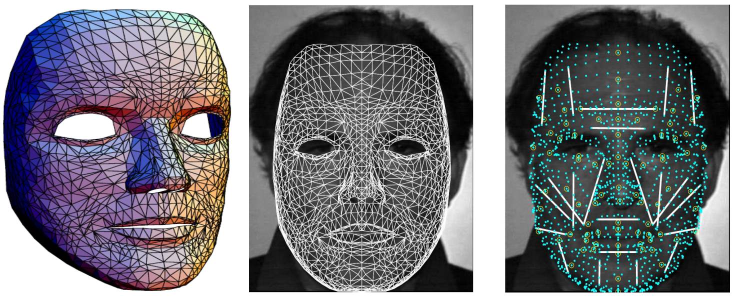


Figure 3.15: Face model and muscle layout proposed by Essa and Pentland [107].

Breton et al. [111] defined 25 muscles for facial expression animation. Upper part of the face contains 8 muscles for controlling the movement of eyebrow and forehead. Lower part of the face contains 17 muscles for controlling the movement of lips. Figure 3.16 presents face model and muscle layout.

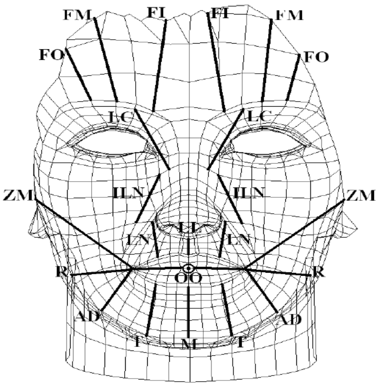


Figure 3.16: Muscle layout in the model proposed by Breton [111].

Zhang [112] developed a 3D face model for facial expression animation that consists of 753 vertices and 1394 faces. Zhang et al. [113] selected 23 facial muscles to animate facial expressions. The visual representation of face model and selected muscles is shown in Figure 3.17.

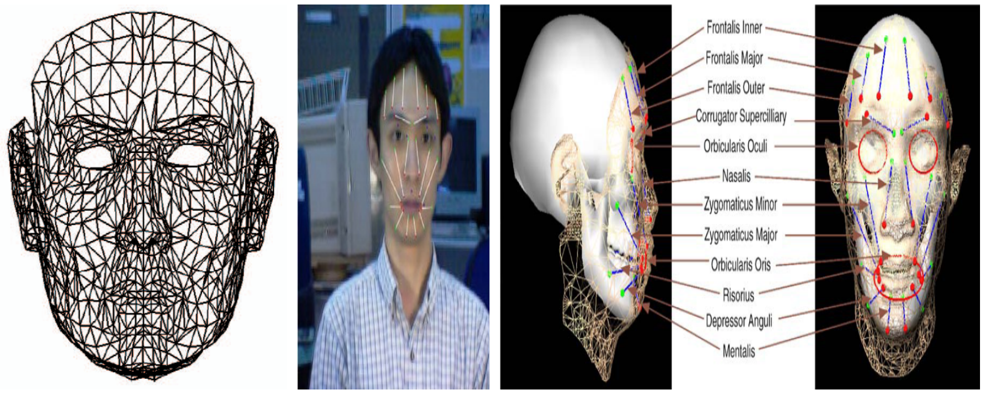


Figure 3.17: Face model [112] and muscle layout [113] proposed by Zhang.

3.3 HIgh resolution GEneric Model – HIGEM

In this study we propose a generic wireframe model that conforms to the human face anatomy. HIgh polygon Generic face Model or HIGEM is presented in Figure 3.18. It comprises of 612 nodes and 1128 polygonal surfaces, which are known as faces in computer graphics terminology.

HIGEM includes all major muscles of the human face. Each muscle is represented by an insertion point (on the skin) and an origin point (on the skull). These points are used to determine the directions of the muscle forces. The placement of muscles on HIGEM is detailed in the next section.

HIGEM is made available to researchers at Işık University Pattern Recognition and Machine Intelligence Laboratory web site [114].

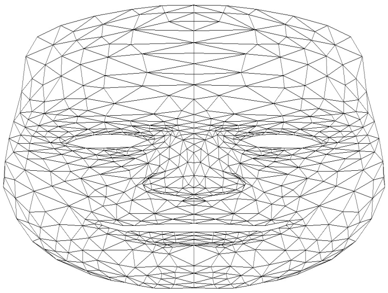


Figure 3.18: HIGEM face model.

3.4 Placement of Muscles

We assigned on the wireframe model the origin and insertion points for each muscle described in the previous section. This assignment step is done once in the beginning of the study, then the muscle-based wireframe model will be used for all subjects after a simple customization step. In the wireframe model, each muscle is represented by an origin and insertion vertex, which describe a ray in the 3 dimensional space. The information on muscle directions is used to solve for muscle activation levels in the final step of our analysis.

In this study we followed the study of Breton et al. [111] and graphical sketches of Goldfinger [21] in placement of muscles on the wireframe model. We define 18 muscles (Medial Frontalis (×2), Lateral Frontalis (×2), Procerus, Levator Labii Superioris Alaeque Nasi (×2), Levator Anguli Oris (×2), Zygomatic Major (×2), Risorius (×2), Depressor Anguli Oris (×2), Depressor Labii Inferioris (×2), Mentalis) on the model. We categorize the facial muscles into two groups. The first group controls the movement of the eyebrow and forehead and the second group is related with lower region of the face. Muscle layout on HIGEM is presented in Figure 3.19. We marked the muscle-skin connection points and directions of the muscles under the constraint of facial anatomy. In figure red and green points stand for the muscle insertion and origin points, respectively. Blue lines in this figure represent muscles, which are modelled as linear fibers.

3.5 Muscle Model

Muscles are fiber structures that can only contract, producing pulling forces on the attached skin. The attachment of a muscle to the skin is not a single spot but a region. This is the *region of influence*, where the muscular force is distributed in varying intensities. We model the human face with a wireframe, and the muscle-skin connection points and directions of the muscles are marked under the constraint of facial anatomy.

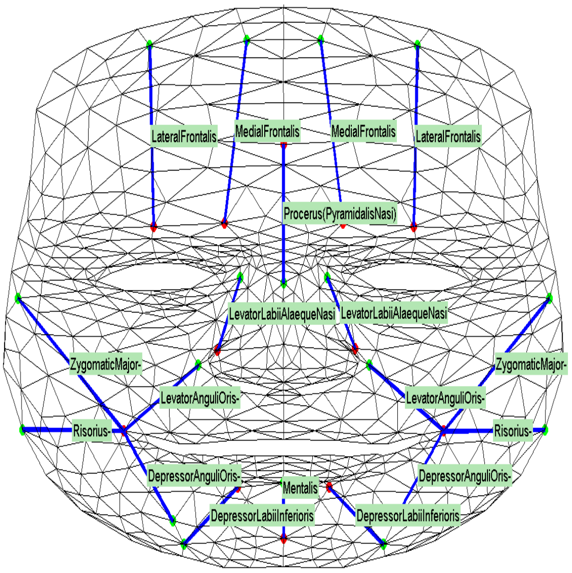


Figure 3.19: Muscle layout on the HIGEM model.

Based on Waters′ research on 3 dimensional animation of facial expressions [103], we define our muscles as linear springs with distributed forces in their regions of influence. A fiber muscle can be shown with vector $\vec{\mathbf{OI}}$ as depicted in Figure 3.20. The contraction of a muscle affects all vertices of the wireframe model that fall in this bounded region.

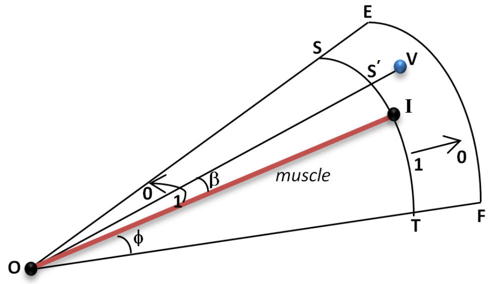


Figure 3.20: Waters′ muscle model.

In Figure 3.20, $\mathbf{V}$ is any wireframe vertex, β is the angle of deviation from the muscle fiber, ϕ is the angular limit for region of influence. The force exerted by the muscle on wireframe vertices is faded as the angle of deviation β approaches angular limit ϕ. On the radial axis, the muscle force increases until the insertion point $\mathbf{I}$ and fades back to zero in the outer band of the region.

We define force fading coefficients based on the angle of deviation and the length of $\vec{\mathbf{OI}}$ vector. The angular fading coefficient δ_A is computed with;

$$\cos\beta = \vec{\mathbf{OI}} \cdot \vec{\mathbf{OV}} / (||\vec{\mathbf{OI}}|| \times ||\vec{\mathbf{OV}}||)$$
$$\delta_A = \begin{cases} \frac{cos\beta - cos\phi}{1 - cos\phi} & \text{if } cos\beta \geq cos\phi \\ 0 & otherwise \end{cases} \tag{3.1}$$

Denoting the fiber length $||\vec{\mathbf{OI}}||$ and maximum radial distance of influence $||\vec{\mathbf{OE}}||$ with r and r_{max} respectively, we define radial fading coefficient δ_R as;

$$\delta_R = \begin{cases} cos(\frac{r - ||\vec{\mathbf{OV}}||}{r} \frac{\pi}{2}) & \text{if } ||\vec{\mathbf{OV}}|| \leq r \\ cos(\frac{||\vec{\mathbf{OV}}|| - r}{r_{max} - r} \frac{\pi}{2}) & \text{if } r < ||\vec{\mathbf{OV}}|| \leq r_{max} \\ 0 & otherwise \end{cases} \tag{3.2}$$

The overall fading in the region of influence is the product of angular and radial fading coefficients;

$$\delta = \delta_A \cdot \delta_R \tag{3.3}$$

This model distributes muscle forces conforming with the physical reality of spreading muscle fibers beneath a region of influence. A sample distribution of force for the Procerus muscle is illustrated in Figure 3.21. The sizes of the red dots represent the magnitude of forces exerted by the Procerus muscle. The muscle forces are always directed towards vertex **O**, where the muscle attaches to the skull. The range of the muscle effects can be defined as $[0, 1]$.

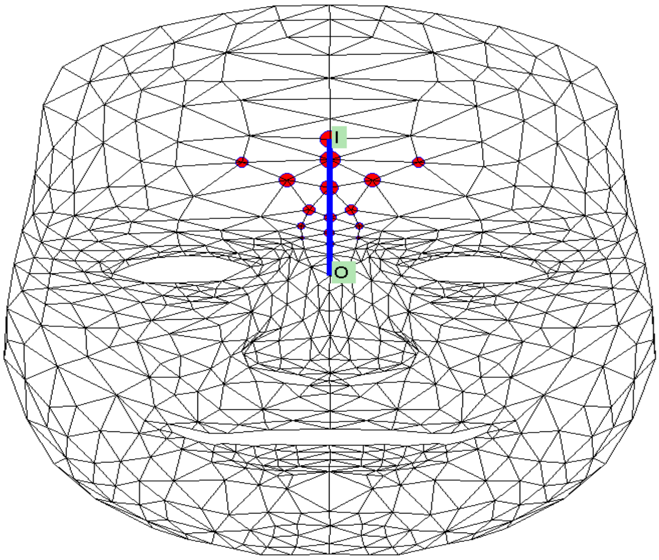


Figure 3.21: Distribution of forces exerted by the Procerus muscle.

We consolidate the effects of muscles on wireframe vertices in matrix **A**, which serves as our muscle map. The muscle map is solely dependent on the anatomical structure of human face. This is a $3n \times m$ matrix where n is the number of vertices and m is the number of muscles. The first three rows of matrix **A** stand for the effect of each muscle on the first vertex in x, y and z axes for a unit muscle force. The product of **A** with the $m \times 1$ vector of muscle activations ($\vec{\mathbf{f}}_{\mathbf{m}}$) gives us $3n \times 1$ vector of muscle forces on each vertex in each axis ($\vec{\mathbf{f}}_{\mathbf{s}}$) (Eq. 3.4).

$$\mathbf{A}\vec{\mathbf{f}}_{\mathbf{m}} = \vec{\mathbf{f}}_{\mathbf{s}} \tag{3.4}$$

Chapter 4

Wireframe Customization

A typical model-based facial expression recognition system incorporates the following five stages; detecting the human face in the input, fitting a 3D model onto face region, tracking the rigid body motion of the head and the deformations of the face, extracting features and classifying the facial expression. As one of the earlier stages, registering the subject's face with a generic 3D face model is critical for the overall success of the system. The error introduced in this stage may accumulate and thus amplify in the later stages. The adaptation of the generic face model must be precise and flexible enough to accommodate interpersonal variations. This stage is referred to as *adaptation*, *initialization* or *customization* by researchers.

The wireframe adaptation step can be done manually or automatically. In manual fitting user drags the vertices of the wireframe in the 3D coordinate system and visually matches their projections to the camera plane with facial landmark points. Semi-automatic fitting requires the user to identify a subset of landmark points in the input image. In this scenario, vertices of the wireframe are automatically translated to match with the identified landmark points. In automatic fitting feature detection algorithms are run to extract important landmarks in the image input and the generic 3D face model is deformed to match the estimated positions of these landmarks.

Essa [15] used Pentland and Moghaddam′s [115] view-based and modular eigenspace methods for fitting 3-D face model to a face in an image. The positions of the eyes, nose and lips are extracted automatically. According to the positions of these features the canonical face mesh is deformed and matched with the face image.

Active Shape Models (ASM) approach is proposed by Cootes et al. [78]. These models are parametric deformable models. They are used to create models of human hearts, hands and faces. A statistical shape model of the face object is built using a set of training examples. Pose and shape parameters are iteratively modified for a better fit. Lu et al. [116] used ASM algorithm for detecting facial feature points and silhouettes. Head pose is determined by solving the point-to-point and point-to-curve correspondences. Iterative closest point method is used for converting the point-to-curve correspondences to the point-to-point correspondences.

Active Appearance Model (AAM) was also proposed by Cootes et al. [79] for matching a generic face model to input face image. AAM combines the statistical model of the shape and the gray-level appearance of the object of interest. The synthesized model is projected onto the face image and matching is done iteratively. Ahlberg [117] used a color-based algorithm for detecting the size and the position of the face, and applied AAM search to do the fitting. Dornaika and Ahlberg [118] ultimately proposed two appearance-based fitting methods. In their first method they did a locally exhaustive and directed search in parameter space, and in the second one they decoupled the estimation of head and facial feature motion. They demonstrated the robustness of their fitting method on video sequences. Also, Dornaika and Ahlberg [119] designed a fast and reliable active appearance model search for face tracking.

Krinidis and Pitas [120] used a semi-automatic approach for fitting the wireframe model to a face image. The face model is a 2D mesh whose elements are springs with stiffness. In the first step they coarsely initialized the wireframe on the face image, and then manually matched model nodes with the corresponding positions of the face image. Using these correspondences driving force values that will cause the deformation of the wireframe are calculated. Proposed spring-mesh model is also used in the studies of Kotsia and Pitas [10], Kotsia et al. [121], Vretos et al. [122], Krinidis and Pitas [123].

We propose a semi-automatic model fitting method for wireframe deformation. The problem of projecting facial landmarks from 2D images to 3D space is underdetermined and for that reason most semi-automatic and automatic fitting approaches employ an iterative error minimization scheme. The real challenge in the customization task is to estimate the 3D locations of vertices using the 2D input. We studied ray tracing method which will be presented in Section 4.1. In Sections 4.2 and 4.3 we will introduce our manual wireframe fitting and semi-automatic wireframe fitting studies.

4.1 Ray Tracing

In this study we aim to detect and track facial feature points on the subject′s face, which exist in real world. The input to our system is a 2D observation of the subject. We would like to trace facial features in the given observation into a virtual space where our anatomical face model is positioned. In Computer Graphics, one of the methods to project a scene in 3D world to the camera view plane is perspective projection. Our goal on the other hand is to estimate the 3D position of a point given its location in screen coordinates. We will start our discussion with perspective projection and continue with our assumptions to do a variant of the well known ray tracing method.

Perspective projection is a method for mapping a 3D object onto 2D camera plane. Following equations show the general perspective-transformations.

$$x_p = x(\frac{z_{prp} - z_{vp}}{z_{prp} - z}) + x_{prp}(\frac{z_{vp} - z}{z_{prp} - z}) \tag{4.1}$$

$$y_p = y(\frac{z_{prp} - z_{vp}}{z_{prp} - z}) + y_{prp}(\frac{z_{vp} - z}{z_{prp} - z}) \tag{4.2}$$

In these equations, z_{vp} stands for the z-coordinate of the view (camera) plane and $(x_{prp}, y_{prp}, z_{prp})$ point stand for the projection reference point.

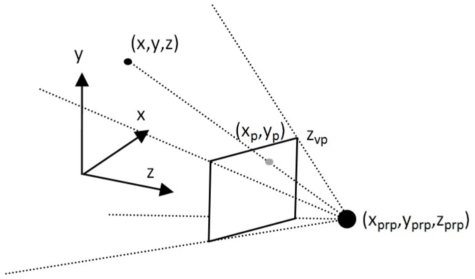


Figure 4.1: Perspective projection and ray tracing.

The projection reference point is chosen on the z-axis to simplify the calculations of perspective projection. When the projection reference point is fixed on z axis ($x_{prp} = y_{prp} = 0$), Eq.4.1 and 4.2 can be rewritten as in Eq.4.3.

$$f_p(x, y, z) = (x_p, y_p) = \begin{bmatrix} x(\frac{z_{prp}-z_{vp}}{z_{prp}-z}) \\ \\ y(\frac{z_{prp}-z_{vp}}{z_{prp}-z}) \end{bmatrix} \tag{4.3}$$

This mapping projects each 3D point to the camera view plane as depicted in Figure 4.1. Similarly, any point on the camera view plane is a projection of a 3D point that lies in the ray that emanates from the projection reference point and passes through the point on the camera view plane. Given the depth (z) of the

3D point, we can invert Eq.4.3 to find its exact location on the 3D coordinate system. This is a variant of ray tracing method that will be exploited in the wireframe customization and wireframe deformation calculation stages.

$$f_p^{-1}(x_p, y_p, z) = (x, y, z) = \begin{bmatrix} x_p(\frac{z_{prp}-z}{z_{prp}-z_{vp}}) \\ \\ y_p(\frac{z_{prp}-z}{z_{prp}-z_{vp}}) \\ \\ z \end{bmatrix} \tag{4.4}$$

4.2 Manual Fitting of the Wireframe Model to the Subject′s Face

The first task for model based facial expression study is wireframe customization. In initial project work plan we planned to achieve this step manually. In manual fitting all vertices on the wireframe model are manually marked on face image. Since we are marking all the vertices on wireframe model, we need to reduce our original wireframe model′s polygon size. The new generated low polygon wireframe model contains 64 vertices. In manual fitting we marked 64 facial landmarks on face image. Figure 4.2 represents the marking process.

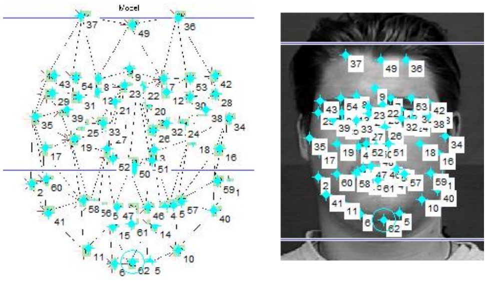


Figure 4.2: Marking facial landmarks.

After the marking process, feature points on video frame are translated to the corresponding vertices on wireframe model through ray tracing. As described in Section 4.1. Figure 4.3 shows the obtained results in manual wireframe fitting study. Original wireframe model is shown on the left, customized wireframe model is presented in the middle and the result of wireframe projection to the input image is shown on the right.

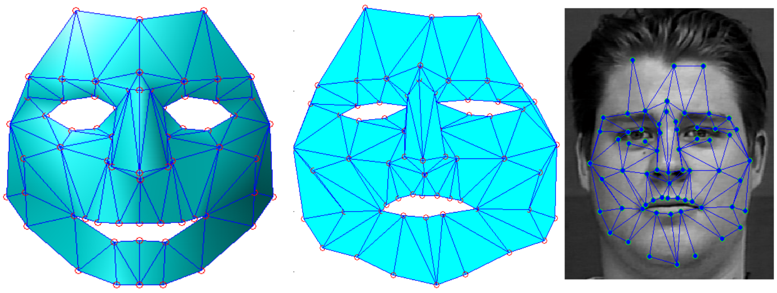


Figure 4.3: Test results.

4.3 Semi-Automatic Fitting of the Wireframe Model to the Subject′s Face

Precise registration of a generic 3D face model with the subject′s face is a critical stage for model based analysis of facial expressions. In this study we propose a semi-automatic model fitting algorithm to fit a high-polygon wireframe model to a single image of a face (Figure 4.4). We manually mark important facial landmarks both on the wireframe model and the face image. We carry out an initial alignment by translating and scaling the wireframe model. We then translate the wireframe landmarks in the 3D wireframe model so that their perspective projections coincide with the image facial landmarks. The vertices that are not manually labelled as landmark are translated with a weighted sum of vectorial displacement of k neighboring wireframe landmarks, inversely weighted by their 3D distances to the vertex. Our experiments indicate that we can fit a high-polygon model to the subject′s face with modest computational complexity.

4.3.1 Selection of Facial Landmarks

A facial landmark is a point that represents an important feature on the face of the subject such as eye corner, tip of the chin or top of the forehead. Facial landmarks

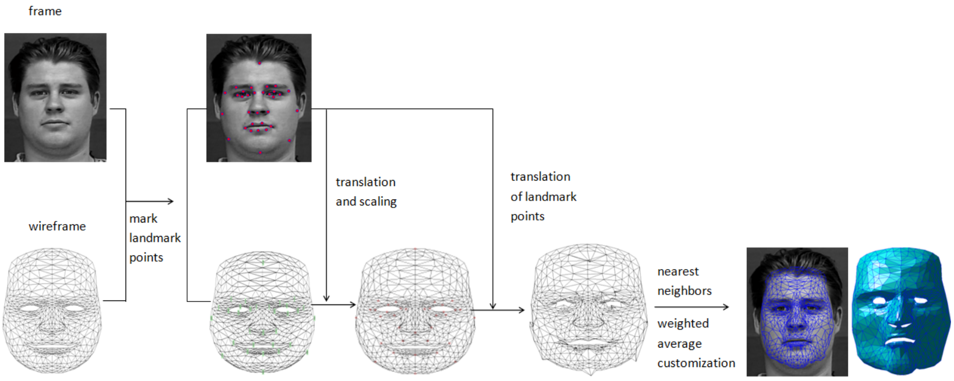


Figure 4.4: Semi-automatic fitting method.

can be marked manually on the subject′s face or can be detected automatically. In semi-automatic fitting methods facial landmarks are located manually. In this study we marked 32 facial landmarks both on face image and wireframe model. Figure 4.5 shows the selected facial and wireframe landmarks.

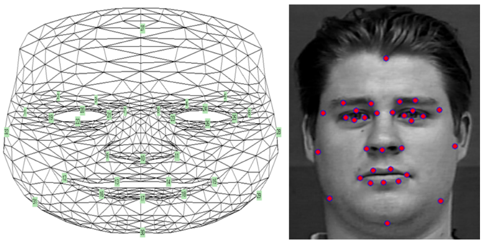


Figure 4.5: Marked facial landmarks.

In this study the points that are considered more informative than the others are selected as facial landmarks. For instance, the top of the forehead and the tip of the chin are selected as facial landmarks in order to assess the height of the wireframe. We also selected points on the extremities of the left and right cheeks as an indication of the width of the wireframe. The other features, such as those around the eyes and eye sockets, are carefully selected to sketch the general shape of subject′s face. These landmarks are consistent with those proposed by Cootes [79].

4.3.2 Wireframe Customization through Ray Tracing

We aim to reshape the generic wireframe model based on the facial landmarks selected on the 2D image. In order to calculate translation vectors for the wireframe vertices we need to estimate the 3D coordinates of these facial landmarks. Facial landmarks on face image are translated to 3D wireframe vertices using the ray tracing method that was introduced in Section 4.1.

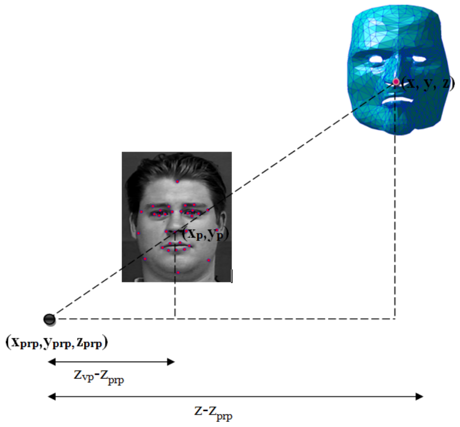


Figure 4.6: Ray tracing.

Our generic wireframe model is fixed with all of its wireframe landmarks. In other words, the only input to our system is the facial landmarks on the 2D facial image. When the user selects a facial landmark on the face, we know the depth of the corresponding vertex in the wireframe. We apply ray tracing keeping the depth of the vertex fixed (Figure 4.6).

4.3.3 Wireframe Alignment

For an initial alignment of the wireframe model with the facial landmarks, we will translate the wireframe and scale it in x, y and finally z directions. For rigid body translation of the entire wireframe, we map each facial landmark to the 3D space through ray tracing and calculate the mean shift for wireframe vertices. (Eq.4.5).

$$T = \frac{1}{n}\sum_{i=1}^{n} f_p^{-1}(p_i^l) - v_i^l \tag{4.5}$$

In Eq.4.5, n is the number of facial landmarks, f_p^{-1} is ray tracing, p_i^l and v_i^l stand for facial landmarks and wireframe landmarks, respectively. Computed translation vector is applied to all vertices of the wireframe model.

$$v_{i,centered} = v_{i,original} + T \tag{4.6}$$

We applied varying scaling factors on the x, y and z axes of the wireframe model to achieve a better initial alignment to subject′s face. The scaling factor in the x axis is found by the ratio of inverse projected width of the face to the original width of the wireframe model. To estimate the inverse projected width, we used the facial landmarks that lie on the left and right cheeks (Figure 4.7). Similarly, the forehead and chin facial landmarks are used to scale the wireframe in y axis. Scale factor for the depth of the head is assumed to be 1.15 times the scaling factor of the width of the head. This ratio was empirically found and reported by Luximon et al. [124].

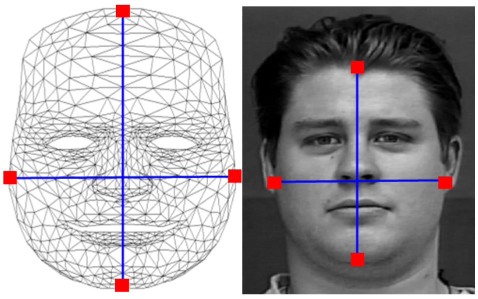


Figure 4.7: Width and height values of wireframe model and face image.

4.3.4 Nearest Neighbors Weighted Average Customization (NNWA)

When the translation and scaling stages are complete we have an initial alignment of the wireframe with the face image. To determine the final coordinates of wireframe landmarks we apply ray tracing method on each facial landmark of the face. Here the depth of a facial landmark is assumed to be the same as the z coordinate of the corresponding wireframe landmark, as shown in Eq.4.4. After this operation we have the customized coordinates of the wireframe landmarks on the wireframe model and are ready to customize the non-wireframe landmarks.

For each wireframe landmark v_i^l we computed the translation vector by subtracting its initial position in original wireframe from the customized position in the deformed wireframe model.

$$\Delta v_i^l = v_{i,orig}^l - v_{i,custom}^l \tag{4.7}$$

For each non-wireframe landmark in the original wireframe we calculated the Euclidean distances between the vertex and the wireframe landmarks. These distances are to be used to determine the nearest k wireframe landmarks and their weights in calculating the displacement vector for the vertex.

$$d_{i,j} = \left\| v_{i,orig}^l - v_{j,orig}^{nl} \right\| \qquad i = 1...32, j = 1...580 \tag{4.8}$$

In Eq.4.8, $v_{j,orig}^{nl}$ and $v_{i,orig}^l$ represent the coordinates of a non-landmark and landmark vertices on the original wireframe model, respectively. In our generic wireframe model we have 32 wireframe landmarks and 580 non-wireframe landmarks.

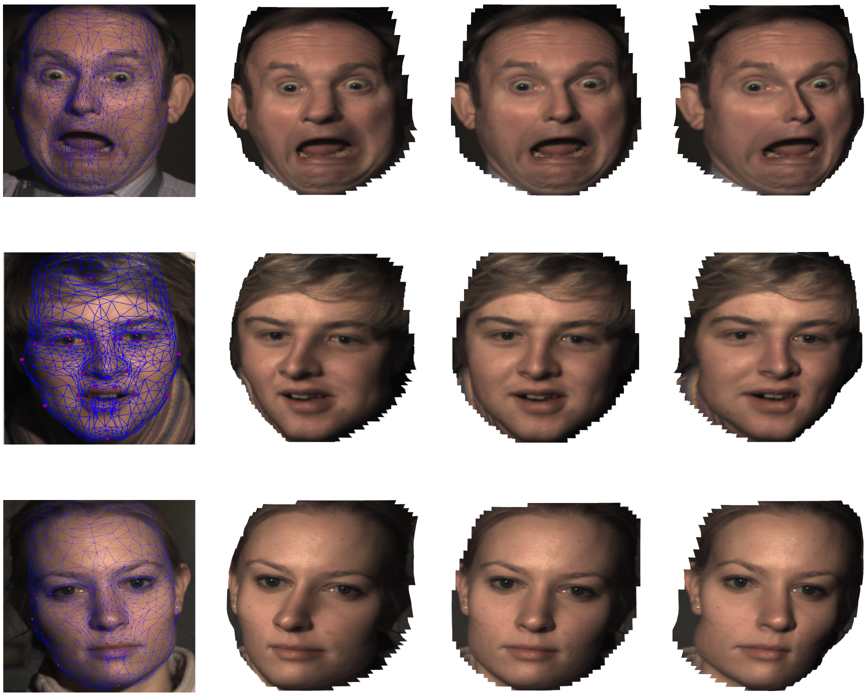
Each non-wireframe landmark is translated with a sum of translation of k nearest-neighbor wireframe landmarks, weighted by the inverse of their distances to the vertex.

$$T_j = \frac{\sum_{i=1}^{k} \frac{\Delta v_i^l}{d_{ij}^2}}{\sum_{i=1}^{k} \frac{1}{d_{ij}^2}} \tag{4.9}$$

$$v_{j,custom}^{nl} = v_{j,orig}^{nl} + T_j \tag{4.10}$$

4.3.5 Customization Experiments and Results

We evaluated the performance of the NNWA algorithm using the Photoface and Bosphorus databases. We visually examined the customization performance on chosen subjects of the Photoface database. For this experiment, we annotated 32 landmarks on the chosen images. Once face modelling is complete, we mapped the texture of the face image to wireframe faces through interpolation. The snapshots of the customized models are presented in Figure 4.8.



(i) Wireframe (j) Left (k) Front (l) Right

Figure 4.8: Modelling examples from the Photoface data set. The left column illustrates the projection of the customized wireframe model onto the image. Next 3 columns illustrate different views of the models with mapped texture.

We conducted our next batch of experiments on the Bosphorus 3 dimensional face data set. Bosphorus data set comprises of 4666 2 dimensional images and their corresponding 3 dimensional data clouds. The data set provides various facial expressions of 105 subjects. We carried out our experiments on 104 subjects since the 3 dimensional data cloud for one subject (subject #7) was not correctly parsed. In our experiments we utilized the neutral poses of subjects.

The purpose of our next experiment is to perform a quantitative evaluation of the NNWA algorithm. For that reason, we manually marked the facial landmarks on both the facial images and data clouds of all subjects.

4.3.6 Evaluating the Performance of Customization

Once a face model is generated, we need to find its deviation from the data cloud of the subject, which is the ground truth for our experiment. For this purpose we brought both the model and the data cloud to a common scale and aligned them around origin in 3 dimensional coordinates. We define error as the absolute value of distance for each wireframe vertex to the nearest data point in the data cloud. The mean error is calculated over all 612 nodes of the wireframe model. The same process is repeated to evaluate the performance of customization through Procrustes analysis. ASM operates on modes of variation of the landmark points. Therefore the mean error of ASM was calculated only over the landmark points.

The faces we are dealing with can be of different sizes. In order to bring all error measurements to a standard base we exploited *relative error* in our experiments. We benefited from the bounding box of the 3 dimensional data cloud in quantifying the relative error. For each subject the relative error is obtained by dividing the mean error with the diagonal length of the 3 dimensional bounding box belonging to that subject.

We also developed a model coloring strategy to illustrate the error variation on the surface of the model. We colored the negative and positive errors in shades of blue and red, respectively. A perfect match of the model with the data cloud is represented in green. The frontal and lateral profiles of a customized model are illustrated in Figure 4.9.

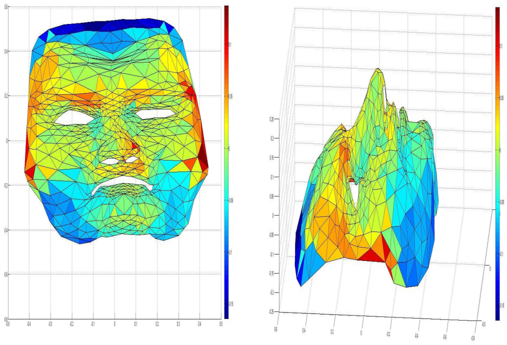


Figure 4.9: Illustration of error variation on a customized model

4.3.7 Identifying the Landmark Vertices

Marking a substantial number of facial landmarks is an error prone task. Therefore determining an ideal number of landmark traits that is sufficient in accurately defining a face is critical in our research. To reduce the number of landmarks we observed the variation of the magnitude of mean error with respect to varying number of landmarks.

We performed this experiment by gradually decreasing the number of landmarks from 42 to 10. In this experiment we employed 15 randomly selected subjects from the Bosphorus 3 dimensional face data set.

As expected the mean error of the model increases as the number of landmarks decreases (Figure 4.10). There is a trade-off between the effort required in locating facial landmarks and the accuracy. Taking this fact into consideration we chose to employ 32 landmarks in the customization process. All of our subsequent experiments are conducted using these landmarks.

Following a similar analysis, we chose number of neighbors k as 5. Choosing a very low value for k deteriorates the smoothness of the model. On the other hand when k is above 10 the landmarks that are in farther regions of the face start influencing the customization of a non-landmark vertex, reducing the modelling performance.

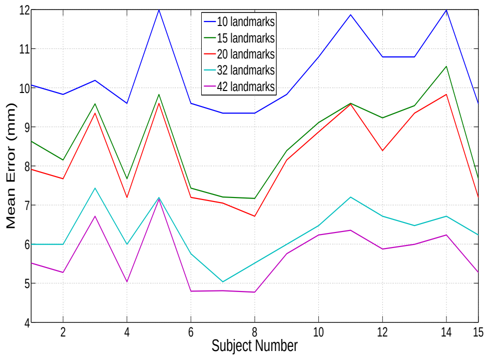


Figure 4.10: Mean error comparison for the proposed method with respect to varying number of landmarks

4.3.8 NNWA Customization Results

We applied our algorithm on 104 subjects in the Bosphorus face data set. Our results demonstrate low relative error values and variability indicating the robustness of the proposed technique (Figure 4.11). We carried out these experiments only on the frontal image of the subject's face. Therefore relative error for each subject was determined over a single face model.

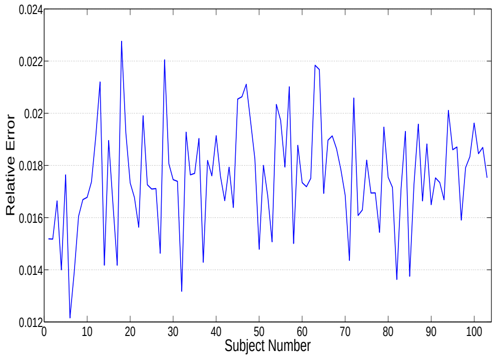


Figure 4.11: Relative error using 5-neighbor NNWA customization for 104 subjects in the Bosphorus data set

We also observed the variation of relative error on an individual subject. To evaluate this we applied the NNWA customization on a randomly selected subject in the Bosphorus data set. Figure 4.12 illustrates the nearest neighbor customization on subject number 15.

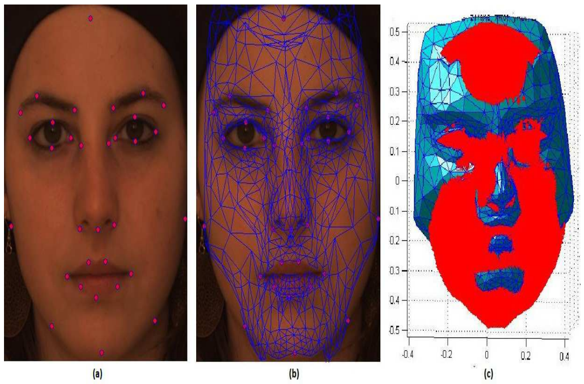


Figure 4.12: Facial landmarks on sample subject, generic wireframe model overlayed on the image, acquired 3 dimensional model and the data cloud (Subject Number 15)

The graph that demonstrates the relative error on each of the 612 vertices is presented in Figure 4.13. The majority of the vertices in the generic model consistently demonstrated low relative error magnitudes.

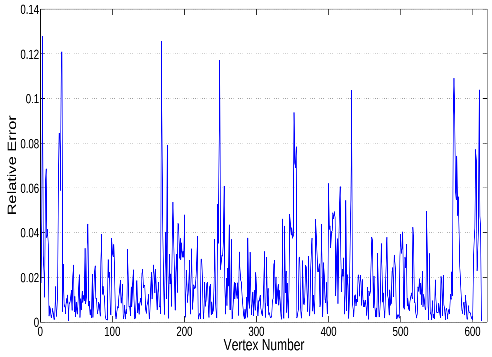


Figure 4.13: Relative error for the vertices of the wireframe model (Subject Number 15)

4.3.9 Procrustes Analysis Results

Procrustes analysis is an alignment technique for superimposing one or more shapes onto each other. This is performed through isotropic scaling, translation, and rotation. Procrustes analysis iteratively finds the best fit between two or more shapes outlined by the landmark points. It only allows rigid body transformations on the data sets and the transformations conserve the relative distance between feature points. Procrustes analysis has many variations. Of these different variations, General Procrustes analysis [125, 126], otherwise known as GPA is one of the more commonly exploited techniques in shape correspondence.

The alignment process of the General Procrustes analysis consists of six stages.

1. Normalize all shapes to unit size and translate their center of masses to origin.
2. Determine mean shape $\mathbf{m} = \frac{1}{n}\sum_i \mathbf{x_{i,j}}$ where i and j represent observations and cloud points.
3. Align each shape i with $\mathbf{m}$ via transformation T_i.
4. Re-calculate $\mathbf{m} = \frac{1}{n}\sum_i T_i\ (\mathbf{x_{i,j}})$.
5. Translate m to origin and normalize its size.
6. Go to step 3 until convergence.

Procrustes analysis is usually employed as the first step in 3 dimensional modelling. We also applied Procrustes analysis and evaluated the acquired relative error rates. Figure 4.14 presents a comparison of the relative error for 104 subjects using two methods; NNWA customization and Procrustes analysis. As expected, NNWA customization performs substantially better than Procrustes analysis.

4.3.10 Active Shape Model (ASM) Results

We implemented ASM to compare its results with the NNWA algorithm. ASM is originally proposed for 2 dimensional models. In our research we extended ASM to be used with a 3 dimensional generic face model. Our ASM implementation can be outlined with the following algorithm.

1. Align the 3 dimensional data clouds using Procrustes analysis (translation, rotation and isotropic scaling).

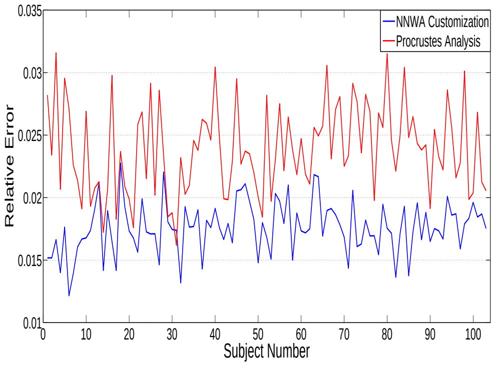


Figure 4.14: Relative error comparison for NNWA customization and Procrustes analysis

2. Apply principal component analysis (PCA) on the 3 dimensional data set to obtain the mean model $\mathbf{m}$ and eigenvectors $\mathbf{a}$.

3. Find the Jacobian of residual with respect to transformation parameters in 6 degrees of freedom.

4. Apply Gauss-Newton approximation to estimate the transformation parameters.

5. Apply the estimated transformation on $\mathbf{m} + \sum_{\mathbf{i}} \gamma_{\mathbf{i}} \mathbf{a}_{\mathbf{i}}$ where $\gamma_{\mathbf{i}}$ are the shape parameters.

6. Find the Jacobian of the residual with respect to shape parameters.

7. Apply Gauss-Newton approximation to estimate the shape parameters.

8. Go to step 3 until convergence.

We define the residual vector as the square of Euclidean distance between each facial landmark and the perspective projection of wireframe landmark. Figure 4.15 depicts the relative error comparison between NNWA and ASM algorithms. Although both ASM and NNWA methods attained very low error rates, we observe that ASM consistently outperforms NNWA customization. However this fact alone does not make ASM superior to the proposed method. ASM has important constraints as a statistical modelling technique and an iterative optimization algorithm.

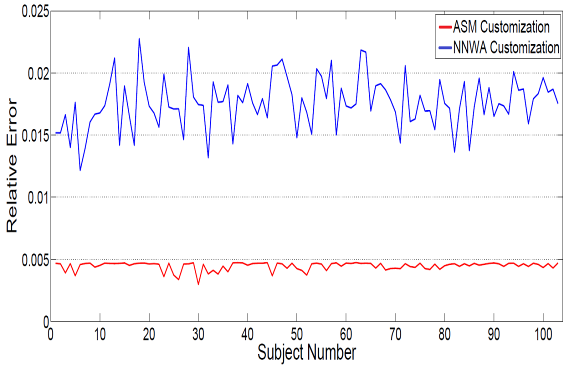


Figure 4.15: Relative error comparison for NNWA and ASM algorithms

ASM requires a large data set of 3 dimensional data clouds with data point correspondences for deriving the modes of variation (eigenvectors) for the data set. NNWA algorithm utilizes a generic wireframe model and directly operates on a facial image with marked landmarks. As an iterative error minimization approach ASM does not guarantee convergence to the global optimum. Moreover, the demonstrated relative error values for ASM are quantified only for 32 correspondence points, whereas NNWA algorithm provides customization for non-landmark vertices of the model as well.

Chapter 5

Tracking Rigid Body Motion

Given a model that has been customized for a subject, we can identify its vertices that are in the region of influence of any muscles using Eq. 3.3. We identify the projections of these vertices onto the image plane as *feature points*, as illustrated in Figure 5.1. The displacements of these features will be used both to estimate the rigid body motion of the subject's head and relative motion of skin points due to a facial expression. We track feature points on the image plane using the optical flow algorithm proposed by Lucas and Kanade [127].

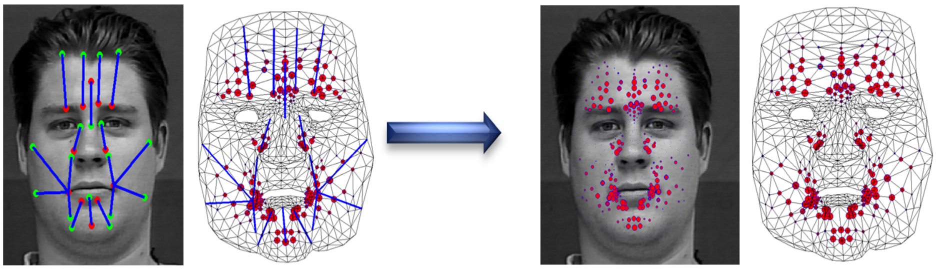


Figure 5.1: Identifying feature points to be tracked. Left: Influence regions of muscles. Right: Projection of vertices (feature points).

5.1 Optical Flow

Input for our system is image sequences that are captured over time. Image data is a function of space (x,y) and time (t) (Figure 5.2). Horn and Schunk [128] define optical flow as the apparent motion of brightness patterns in the image.

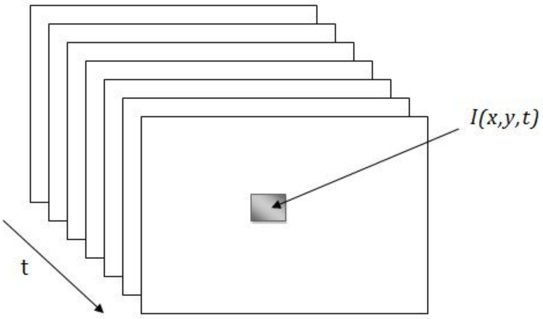


Figure 5.2: Image Data.

A pixel at location (x,y,t) with intensity I(x,y,t) move by Δx, Δy and Δt between consecutive frames. There are three main assumptions for estimating the apparent motion.

- **Brightness Constancy:** Projection of a tracking point looks same in every frame.
- **Spatial Coherence:** Tracking points move like their neighbors
- **Small Motion:** Tracking points do not move very far

We can formulate the motion with the brightness constancy constraint.

$$I(x,y,t) = I(x+\Delta x, y+\Delta y, t+\Delta t) \tag{5.1}$$

Using Taylor Series Expansion we can get

$$I(x+\Delta x, y+\Delta y, t+\Delta t) = I(x,y,t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t + H.O.T \tag{5.2}$$

In Eq. 5.2 the abbreviation H.O.T stands for the higher order terms. We neglect higher order terms and combine equations 5.1 and 5.2 as

$$I(x,y,t) = I(x,y,t) + \frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t \tag{5.3}$$

The term $I(x, y, t)$ appears both side of the equation, so they cancel each other producing the following equation

$$\frac{\partial I}{\partial x}\Delta x + \frac{\partial I}{\partial y}\Delta y + \frac{\partial I}{\partial t}\Delta t = 0 \tag{5.4}$$

We divide both of the terms with Δt

$$\frac{\partial I}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y}\frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t}\frac{\Delta t}{\Delta t} = 0 \tag{5.5}$$

and reorganize the Eq. 5.5

$$\frac{\partial I}{\partial x}V_x + \frac{\partial I}{\partial y}V_y + \frac{\partial I}{\partial t} = 0 \tag{5.6}$$

In Eq. 5.6 terms V_x and V_y stand for the x,y components of the velocity vector. Terms $\frac{\partial I}{\partial x}$, $\frac{\partial I}{\partial y}$ and $\frac{\partial I}{\partial t}$ represent the derivatives of the image at (x,y,t). For simplification we used I_x, I_y and I_t to represent the derivatives.

$$I_xV_x + I_yV_y = -I_t \quad or \quad \triangledown I^T.\vec{V} = -I_t \tag{5.7}$$

This is a single equation with two unknowns (V_x and V_y). In the literature this situation is referred as the *aperture problem*. Lucas and Kanade [127] utilized spatial coherence constraint for generating additional equations and solved the system with the least squares method.

According to spatial coherence constraint we collect all pixels within a window centered at p and generate new equations.

$$
\begin{aligned}
I_x(p_1)V_x + I_y(p_1)V_y &= -I_t(p_1) \\
I_x(p_2)V_x + I_y(p_2)V_y &= -I_t(p_2) \\
&\vdots \\
I_x(p_n)V_x + I_y(p_n)V_y &= -I_t(p_n)
\end{aligned}
\tag{5.8}
$$

In Eq. 5.8, p_i indicates the neighboring pixel and it is located inside the window. Here the size of the window is nxn. As a numeric explanation, if we use a 5x5 window, we can have 25 equations per pixel. We represent those equations in matrix form

$$
A = \begin{bmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ \vdots & \vdots \\ I_x(p_n) & I_y(p_n) \end{bmatrix} \quad v = \begin{bmatrix} V_x \\ V_y \end{bmatrix} \quad b = \begin{bmatrix} -I_t(p_1) \\ -I_t(p_2) \\ \vdots \\ -I_t(p_n) \end{bmatrix}
\tag{5.9}
$$

By inclusion of the spatial coherence constraint, we obtain more equations than unknowns. This system can be solved with convex optimization methods such as x, y, and least squares.

5.2 Lost Features Problem

Drifting of feature points is a major source of error both for estimating the rigid body motion of the head and deformations based on expressions. In an uncontrolled drift, the topology of feature point and its neighbors will be altered. Figure 5.3 depicts drifting of feature point $\mathbf{p}_0$ at time t_k. Note that any feature point is a projection of its corresponding wireframe vertex and without folding this topology change is not permissible.

To identify the drifting feature points, we record the directions of cross products on triangles formed by the feature point with its neighbors. The two vectors

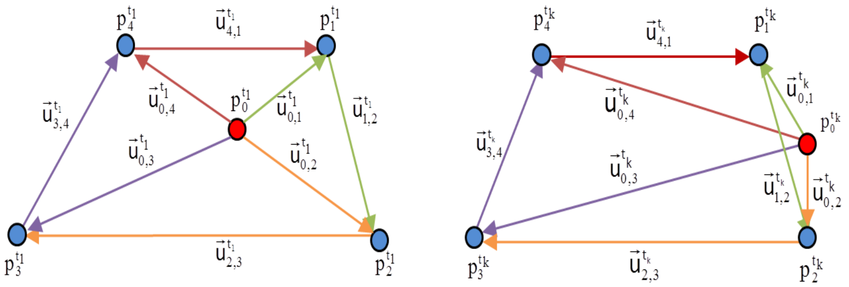


Figure 5.3: Drifting feature point on image plane.

to identify this point as a drifting feature are $\vec{\mathbf{u}}_{0,1}$ and $\vec{\mathbf{u}}_{1,2}$. We compute cross products in each frame, after tracking of feature points;

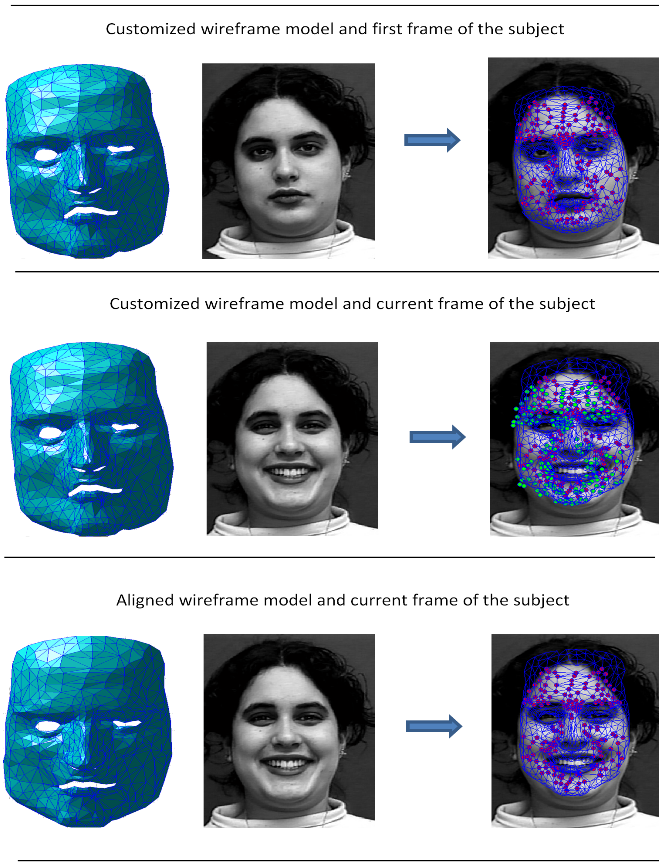
$$\vec{\mathbf{c}}_{1,2}^{t_k} = \vec{\mathbf{u}}_{0,1}^{t_k} \times \vec{\mathbf{u}}_{1,2}^{t_k} \tag{5.10}$$

The calculation of cross product is repeated for all faces the vertex resides on. We identify inversion of the direction of a cross product as a state (topology) change. All feature points that undergo a state change are marked as *untrusted*. The remaining feature points and their corresponding wireframe vertices will be used for estimating the orientation of the head.

5.3 Estimating head orientation

Precise alignment of the face model with the observed face image is mandatory for estimation of relative displacements of vertices. The orientation of the subject's head is determined by greedy search on trusted feature points and their corresponding vertices in 6 degrees of freedom, as depicted in Figure 5.4. A similar greedy search algorithm for finding the head orientation was implemented by Dornaika and Ahlberg [129].

The 3 dimensional rigid body motion of the subject's head is defined by rotations (θ) and translations (T) on x, y and z axis;



Customized wireframe model and first frame of the subject

Customized wireframe model and current frame of the subject

Aligned wireframe model and current frame of the subject

Figure 5.4: Alignment of the face model with the observed face image.

$$\mathbf{b} = [\theta_x \theta_y \theta_z T_x T_y T_z] \tag{5.11}$$

Our search algorithm iteratively seeks for a suitable transformation in each degree of freedom in positive and negative directions. An iteration ends with applying on the model the transformation that produces maximum reduction in error. We define error as the sum squared distance between the projected vertices and tracked feature points. Iterations will end if the error reduction is less than a predefined threshold.

Algorithm 5.3.1: RigidMotion(V, P)

```
//V : landmarks on the face model
//P : feature points on the image plane
b ← {0, 0, 0, 0, 0, 0}

while maxErrReduc > threshold
  do
    currErr ← sumsq(project(V), P)
    maxErrReduc ← 0

    for j ← degrees of freedom
      do
        for k ← {−stepSize, stepSize}
          do
            V ← transform(V, b_j, k)
            err ← sumsq(project(V), P)
            errReduc ← currErr − err
            if errReduc > maxErrReduc
              then
              maxErrReduc ← errReduc
              bestT ← j
              bestDir ← k

    if maxErrReduc > 0
      then b{bestT} ← b{bestT} + bestDir

return (b)
```

This algorithm successfully tracked the head motion in all image sequences in the CK database. It is unreliable however, when the motion of the subject is normal to the image plane. Note that in this case the same set of error minimizing projections can be obtained by displacement in z axis or rotations around the axes of the image plane (pitch and yaw).

Chapter 6

Estimating Deformations

Once the estimation of head orientation is complete, we have the face model aligned with the observed face on the image plane. Note that the projections of the wireframe vertices still would not precisely overlap with the image feature points. The causes of these deviations are (1) inaccuracies in the estimation of head orientation, (2) drifting feature points (untrusted features) and (3) deformations or relative motion of facial feature points due to facial expressions.

Our greedy search algorithm works successfully for small displacements and rotations of the head (Section 5.3). We also identified the drifting feature points and labelled them as untrusted (Section 5.2). The deviations between the *trusted* tracking points and the projections of corresponding vertices serve as indicators of facial expressions. We once again turn to ray tracing to extract the displacements of vertices due to facial expressions.

Figure 6.1 depicts a landmark vertex $\mathbf{v}_0$ and its neighbors on the wireframe model. Assuming that the surfaces are small enough so that they do not bulge or wrinkle, this vertex hypothetically moves on one of the faces it resides on. In this illustration, $\mathbf{v}_0$ moves on the surface defined by $\mathbf{v}_0$, $\mathbf{v}_1$ and $\mathbf{v}_2$.

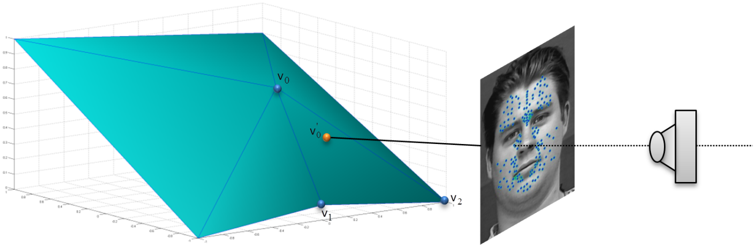


Figure 6.1: Estimating the new coordinates of vertices through ray tracing.

Note that unlike model customization, it is not possible to assume fixed depth for vertices in this stage. However, if we can identify the plane of motion for the vertex, we can estimate its new coordinates through a line–plane intersection. The plane of motion can be any of the faces the vertex resides on. We find the intersection of the ray with each of these faces.

Any vector lying in the plane must be perpendicular to the normal vector. Normal vector of the plane computed as

$$n = (v_0 - v_1) \times (v_0 - v_2) \tag{6.1}$$

And plane equation is defined

$$n \cdot (p - v) = 0 \tag{6.2}$$

where p(x, y, z) and v represent any two points that are on the plane, n is the normal vector of the plane. Then we rearrange the plane equation

$$a(x - x_v) + b(y - y_v) + c(z - z_v) = 0 \tag{6.3}$$

A line is specified by two points in the space. In our study the projection reference point $A(x_{prp}, y_{prp}, z_{prp})$ generates a line equation for each inverse projected facial landmark on the video frame $B(x, y, z) = f_p^{-1}(x_p, y_p, z_{vp})$.

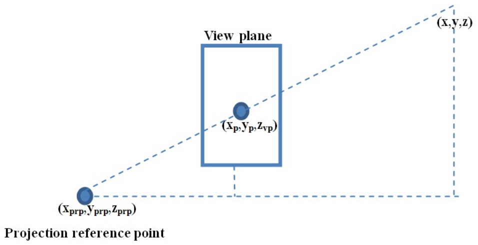


Figure 6.2: A line in the 3D space.

Given two points in the affine space

$$A = x_A \overrightarrow{i} + y_A \overrightarrow{j} + z_A \overrightarrow{k} \quad and \quad B = x_B \overrightarrow{i} + y_B \overrightarrow{j} + z_B \overrightarrow{k} \tag{6.4}$$

The vector pointing from A to B is calculated with

$$\overrightarrow{AB} = B - A = (x_B - x_A)\overrightarrow{i} + (y_B - y_A)\overrightarrow{j} + (z_B - z_A)\overrightarrow{k} \tag{6.5}$$

And the line through A and B is computed as

$$L(v_0') = A + v_0'(B - A) = A + v_0' u \tag{6.6}$$

where u presents the vector $\overrightarrow{AB}$. After generating line and plane equations, we computed their intersections.

$$a((x_A + v_0' x_u) - x_v) + b((y_A + v_0' y_u) - y_v) + c((z_A + v_0' z_u) - z_v) = 0 \tag{6.7}$$

We solved equation and obtained the value of variable $\mathbf{v}_0'$ which is the intersection point of the line and plane.

Line plane intersections are carried out for each triangular surface the vertex in consideration resides on. This produces a hypothesis for the new coordinates of the vertex for each plane. The intersection point $\mathbf{v}'_0$ may be found within or outside the boundaries of a triangular face as depicted in Figure 6.3.

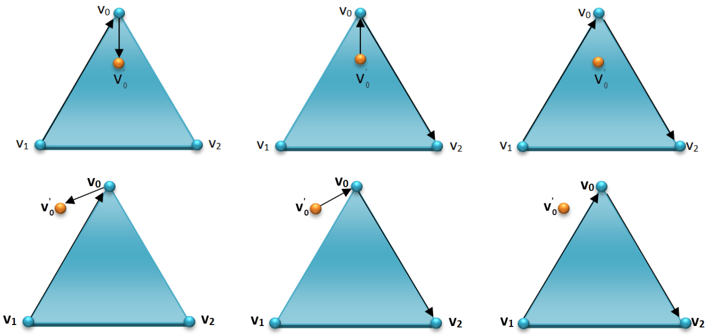


Figure 6.3: Identifying the plane of motion.

Note that the second row in Figure 6.3 implies that the plane of motion is not the selected plane. The intersection point is found within the bounded triangular plane in the first row, which is the plane of motion. To eliminate those intersection points that do not lie in the plane of motion, we determine three normal vectors for each face

$$\begin{aligned}
\vec{\mathbf{n}}_1 &= \overrightarrow{\mathbf{v}_1\mathbf{v}_0} \times \overrightarrow{\mathbf{v}_0\mathbf{v}'_{\mathbf{0}}} \\
\vec{\mathbf{n}}_2 &= \overrightarrow{\mathbf{v}'_{\mathbf{0}}\mathbf{v}_0} \times \overrightarrow{\mathbf{v}_0\mathbf{v}_2} \\
\vec{\mathbf{n}}_3 &\ \ = \overrightarrow{\mathbf{v}_1\mathbf{v}_0} \times \overrightarrow{\mathbf{v}_0\mathbf{v}_2}
\end{aligned} \tag{6.8}$$

where $\mathbf{v}_0$, $\mathbf{v}_1$ and $\mathbf{v}_2$ are the vertex and its neighbors on the *aligned* wireframe model. The intersection point $\mathbf{v}'_0$ is the back projection of the tracked feature point found through line plane intersection. Note that for $\mathbf{v}'_0$ to be in the region bounded by the face, normal vectors must point to the same direction;

$$\vec{\mathbf{n}}_1 \cdot \vec{\mathbf{n}}_3 > 1 - \epsilon \quad \textit{and} \quad \vec{\mathbf{n}}_2 \cdot \vec{\mathbf{n}}_3 > 1 - \epsilon \tag{6.9}$$

These two conditions enable us to identify the plane of motion and the new coordinates of the corresponding vertex for all *trusted* vertices. Note that these conditions do not restrict the vertex to move across the $\overrightarrow{\mathbf{v}_1\mathbf{v}_2}$ edge. In fact, this vertex may be moving in the same same direction with its neighbors and it may cross this edge, which is defined by vertices on the *aligned* model.

An *untrusted* vertex is considered to be drifting and lost in tracking, therefore (1) we update its coordinates with arithmetic average of its neighbors and (2) we project the vertex back to the image plane and relocate the drifting feature point at this location.

Chapter 7

Solving muscle forces

We solve the muscle forces using the anatomical muscle map (Section 3.5), estimated displacements of the wireframe vertices (Chapter 6) and the stiffness of the wireframe. The wireframe is modelled as a 3D surface that is composed of polygons. Each face on the wireframe model is defined by three vertices, representing a triangular plane in the 3D space. The edges between each neighboring vertices are modelled with springs as illustrated in Figure 7.1.

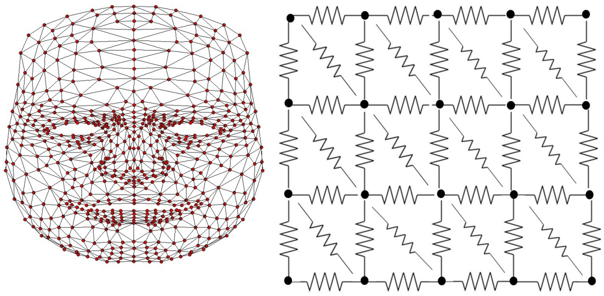


Figure 7.1: Representing the edges of the wireframe model with springs.

The aim of this study is analyzing muscle motions rather than the displacement of points. We used Hooke′s elasticity law for computing the total tensile forces on the grid vertices.

$$\mathbf{F} = -\mathbf{K}\mathbf{v} \tag{7.1}$$

The stiffness matrix $\mathbf{K}$ is an $n \times n$ matrix of effective stiffness values, where n is the number of vertices. The resultant force matrix $\mathbf{F}$ and vertex coordinates $\mathbf{v}$ consist of n rows and 3 columns, representing the x, y and z axes.

In multi body problems, object coordinates are used for determining the magnitude and direction of the force. For each vertex force equations are established for generating the stiffness matrix $\mathbf{K}$ of the whole system. Figure 7.2 shows a single spring identified by the vertices $\mathbf{v}_i$ and $\mathbf{v}_j$ and there are forces $\mathbf{F_i^{ij}}$ and $\mathbf{F_j^{ij}}$ respectively. At equilibrium $\mathbf{F_i^{ij}} + \mathbf{F_j^{ij}} = 0$ or $\mathbf{F_j^{ij}} = -\mathbf{F_i^{ij}}$. These forces calculated in the following equation.

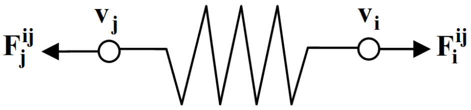


Figure 7.2: Single spring

$$\overrightarrow{\mathbf{F_i^{ij}}} = k_{ij}(l_{ij} - \|\mathbf{v_i} - \mathbf{v_j}\|)\frac{\mathbf{v_i} - \mathbf{v_j}}{\|\mathbf{v_i} - \mathbf{v_j}\|} \tag{7.2}$$

In Eq. 7.2, $\mathbf{F_i^{ij}}$ stands for the 3-dimensional force on vertex i in spring ij. k_{ij} is the stiffness of the spring attached to vertices i and j, and is taken constant in this study. l_{ij} is the rest length of this spring, $\mathbf{v}_i$ and $\mathbf{v}_j$ are the 3D coordinates of the vertices. Note that the first parenthesis in this equation represents the extension or contraction magnitude of the spring. The ratio forms the unit vector from $\mathbf{v}_j$ to $\mathbf{v}_i$, which represents the direction of the spring force.

Force can be calculated as a 3-dimensional vector. We can factor out the scalar terms in this equation to obtain;

$$\overrightarrow{\mathbf{F_i^{ij}}} = \alpha_{ij}(\mathbf{v_i} - \mathbf{v_j}) \quad \textit{where} \quad \alpha_{ij} = k_{ij}\frac{l_{ij} - \|\mathbf{v_i} - \mathbf{v_j}\|}{\|\mathbf{v_i} - \mathbf{v_j}\|} \tag{7.3}$$

The effective stiffness value, α_{ij} depends on the displacements of both vertices. Mutual effective stiffness values α_{ij} and α_{ji} are equal to each other. To represent the entire model with a linear set of equations, we collect the effective stiffness values in a stiffness matrix. If multiple vertices exert force on a single vertex, the effective stiffness values are summed.

An illustrative example is shown in Figure 7.3. Without loss of generality this grid can be considered as 3 dimensional, i.e. perpendicular forces can move the vertices in an out of the plane. We assume that the vertices that are on the boundaries of the grid are fixed.

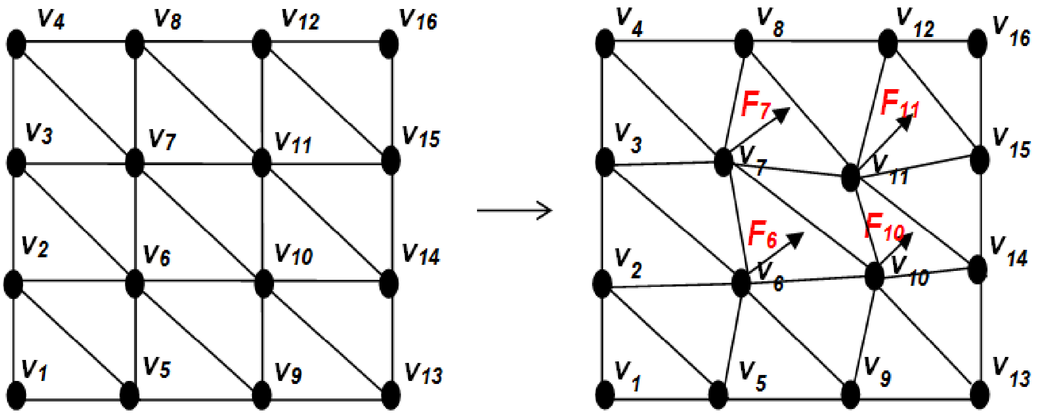


Figure 7.3: A simple wireframe and its deformation under the influence of external forces.

Both neighbors of vertex $\mathbf{v}_1$ are fixed. Since there can be no extension or contraction on the attached springs, their effective stiffness on $\mathbf{v}_1$ are zero. Consequently, the first row of the stiffness matrix $\mathbf{K}$ will be zero. The second vertex, $\mathbf{v}_2$ has only one non static neighbor, $\mathbf{v}_6$. The force exerted on $\mathbf{v}_2$ by motion of $\mathbf{v}_6$ is calculated with;

$$\overrightarrow{\mathbf{F}_{\mathbf{2}}^{\mathbf{26}}} = k_{26}(l_{26} - \|\mathbf{v_2} - \mathbf{v_6}\|)\frac{\mathbf{v_2} - \mathbf{v_6}}{\|\mathbf{v_2} - \mathbf{v_6}\|} \tag{7.4}$$

$$\overrightarrow{\mathbf{F_2^{26}}} = \alpha_{26}(\mathbf{v_2} - \mathbf{v_6}) \;\; where \;\; \alpha_{26} = k_{26}\frac{l_{26} - \|\mathbf{v_2} - \mathbf{v_6}\|}{\|\mathbf{v_2} - \mathbf{v_6}\|} \tag{7.5}$$

Consequently, the second row second column and second row sixth column entries of stiffness matrix $\mathbf{K}$ become α_{26} and $-\alpha_{26}$, respectively. Note that since vertex $\mathbf{v}_2$ is pinned this force will have to be balanced with an external stabilizing force, which will appear in the second row of force matrix $\mathbf{F}$ (Eq. 7.6).

$$\overrightarrow{\mathbf{F_2}} = \begin{bmatrix} 0 & \alpha_{26} & 0 & 0 & 0 & -\alpha_{26} & 0 & 0 & 0 & ... \end{bmatrix} \times \mathbf{v} \tag{7.6}$$

where the resultant force on second vertex is

$$\overrightarrow{\mathbf{F_2}} = \overrightarrow{\mathbf{F_2^{26}}} \tag{7.7}$$

When multiple vertices produce spring forces on a single vertex, we sum their effective stiffness values. As an example, both $\mathbf{v}_6$ and $\mathbf{v}_7$ may produce spring forces on $\mathbf{v}_3$, so the resultant force is computed as;

$$\begin{aligned}\overrightarrow{\mathbf{F_3}} &= \overrightarrow{\mathbf{F_3^{36}}} + \overrightarrow{\mathbf{F_3^{37}}} \\ &= \alpha_{36}(\mathbf{v}_3 - \mathbf{v}_6) + \alpha_{37}(\mathbf{v}_3 - \mathbf{v}_7) \\ &= (\alpha_{36} + \alpha_{37})\mathbf{v}_3 - \alpha_{36}\mathbf{v}_6 - \alpha_{37}\mathbf{v}_7\end{aligned}$$

Eq. 7.8 sets the third row third column of the stiffness matrix to $\alpha_{36} + \alpha_{37}$. The third row sixth and seventh columns of the stiffness matrix become $-\alpha_{36}$ and $-\alpha_{37}$, respectively (Eq. 7.8).

$$\overrightarrow{\mathbf{F_3}} = \begin{bmatrix} 0 & 0 & \alpha_{36} + \alpha_{37} & 0 & 0 & -\alpha_{36} & -\alpha_{37} & 0 & 0 & ... \end{bmatrix} \times \mathbf{v} \tag{7.8}$$

Since the relative coordinates are changed, force can be applied to a non-fixed vertex from its fixed neighbors. For instance, the total force on the sixth vertex will be computed with the following equation.

$$\begin{aligned}
\overrightarrow{\mathbf{F_6}} &= \overrightarrow{\mathbf{F_6^{26}}} + \overrightarrow{\mathbf{F_6^{36}}} + \overrightarrow{\mathbf{F_6^{56}}} + \overrightarrow{\mathbf{F_6^{67}}} + \overrightarrow{\mathbf{F_6^{69}}} + \overrightarrow{\mathbf{F_6^{6,10}}} \\
&= \alpha_{26}(\mathbf{x}_6 - \mathbf{x}_2) + \alpha_{36}(\mathbf{x}_6 - \mathbf{x}_3) + \alpha_{56}(\mathbf{x}_6 - \mathbf{x}_5) \\
&+ \alpha_{67}(\mathbf{x}_6 - \mathbf{x}_7) + \alpha_{69}(\mathbf{x}_6 - \mathbf{x}_9) + \alpha_{6,10}(\mathbf{x}_6 - \mathbf{x}_{10}) \\
&= (\alpha_{26} + \alpha_{36} + \alpha_{56} + \alpha_{67} + \alpha_{69} + \alpha_{6,10})\mathbf{x}_6 \\
&- \alpha_{26}\mathbf{x}_2 - \alpha_{36}\mathbf{x}_3 - \alpha_{56}\mathbf{x}_5 - \alpha_{67}\mathbf{x}_7 - \alpha_{69}\mathbf{x}_9 - \alpha_{6,10}\mathbf{x}_{10}
\end{aligned}$$

Eq. 7.9 will be the entry of the sixth row of the stiffness matrix.

$$\overrightarrow{\mathbf{F_6}} = \begin{bmatrix} 0 & -\alpha_{26} & -\alpha_{36} & 0 & -\alpha_{56} & (\alpha_{26} + \alpha_{36} + \alpha_{56} + \alpha_{67} + \alpha_{69} + \alpha_{610}) & ... \end{bmatrix} \times \mathbf{v} \tag{7.9}$$

We carry out the same analysis on each vertex to put together the stiffness matrix of the entire wireframe. The stiffness matrix for the system shown in Figure 7.3 consists of 16 rows and columns. Due to space restrictions, we only show the first six rows and seven columns of the stiffness matrix. Note that in this model, external forces on boundary vertices are the stabilizing forces that keep the model pinned in the coordinate system, whereas the external forces on vertices 6, 7, 10 and 11 represent muscles that act on these vertices.

$$\mathbf{K} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \\ 0 & \alpha_{26} & 0 & 0 & 0 & -\alpha_{26} & 0 & \cdots \\ 0 & 0 & \alpha_{36}+\alpha_{37} & 0 & 0 & -\alpha_{36} & -\alpha_{37} & \cdots \\ 0 & 0 & 0 & \alpha_{47} & 0 & 0 & -\alpha_{47} & \cdots \\ 0 & 0 & 0 & 0 & \alpha_{56} & -\alpha_{56} & 0 & \cdots \\ 0 & -\alpha_{26} & -\alpha_{36} & 0 & -\alpha_{56} & (\alpha_{26}+\alpha_{36}+\alpha_{56}+\alpha_{67}+\alpha_{69}+\alpha_{610}) & -\alpha_{67} & \cdots \\ & & & \vdots & & & & \end{bmatrix} \tag{7.10}$$

7.1 Least Squares Solution

We calculate the external forces on each vertex using Eq. 7.1. Recall that our muscle map **A** is a $3n \times m$ matrix in which each triplet of rows represents the effect of all muscles on a single vertex in x, y and z axes (Section 3.5). We therefore reorganize the $n \times 3$ force matrix **F** to generate a column vector $\vec{\mathbf{f}}_\mathbf{s}$ of x, y and z components of forces. Thus our anatomy based model for unknown muscle forces $\vec{\mathbf{f}}_\mathbf{m}$ becomes;

$$\mathbf{A}\vec{\mathbf{f}}_\mathbf{m} = \vec{\mathbf{f}}_\mathbf{s} \tag{7.11}$$

This is a linear, over determined system of equations with $3n$ equations and m unknowns. We use constrained least squares to solve this system of equations;

$$\vec{\mathbf{f}}_\mathbf{m} = (\mathbf{A}^T\mathbf{A})^{-1} \cdot \mathbf{A}^T \cdot \vec{\mathbf{f}}_\mathbf{s} \quad , \quad \vec{\mathbf{f}}_\mathbf{m} \geq 0 \tag{7.12}$$

The accuracy of least squares method depends on the condition number of the coefficient matrix. The condition number of our muscle map **A** differs for each subject due to customization of the wireframe model. On average, we found the condition number of the muscle map to be 4.50, which indicates a reliable system to be solved with the least squares method.

Chapter 8

Experiments and Results

In this study we propose a set of novel, anatomy–based features that represent the activation levels of facial muscles. We demonstrate the representation power of facial muscle forces on classification of seven basic facial expressions (anger, disgust, fear, happiness, sadness, surprise and neutral) that are frequently used in the literature. We will start this discussion with the facial expression database used in our experiments (Section 8.1). Next, we will introduce identified muscle forces in our experiments (Section 8.2). We will conclude this chapter with the results and discussion of the facial expression recognition experiments (Section 8.3).

8.1 Database Description

We used Cohn-Kanade (CK) database in our classification experiments [35] as it is the most frequently used database in the literature. This database contains 228 FACS-annotated image sequences of six emotions which are performed by 97 different subjects. Table 8.1 presents the distribution of observations to expression classes.

In CK each recording ends with the peak (apex) of the expression. Since the representative muscle forces are obtained from the last frame of the video session, we

Table 8.1: Distribution of the input data

	# Subjects (Sequences)
Anger	29
Disgust	34
Fear	17
Happy	61
Sad	16
Surprise	71
Total	228

collected the muscle forces that were identified in the last frame of the performed expression.

8.2 Identified Muscle Forces

Processing of each sequence starts with customization of the wireframe model to the subject in the first frame (Chapter 4). Vertices that are in the region of influence of at least one muscle are determined (Section 3.5), projected onto the image plane and the feature points are initialized. Tracking of feature points using optical flow (Section 5.1) provides us an estimate of head orientation (Section 5.3) and relative displacements of wireframe vertices (Chapter 6). These displacements will be used to calculate the external forces on each vertex, which in turn will be used for solving muscle forces (Chapter 7).

Figure 8.1 presents a visual demonstration of our results. The strength of activity for each muscle is coded with line thickness. We observe muscular activity in the forehead for anger and surprise expressions. Disgust is clearly separated from other expressions with muscle activity on both sides of the nose. Muscular activities in fear and happiness are very similar, as are the observed expressions. Sadness is distinguishable with muscular activity on the chin.

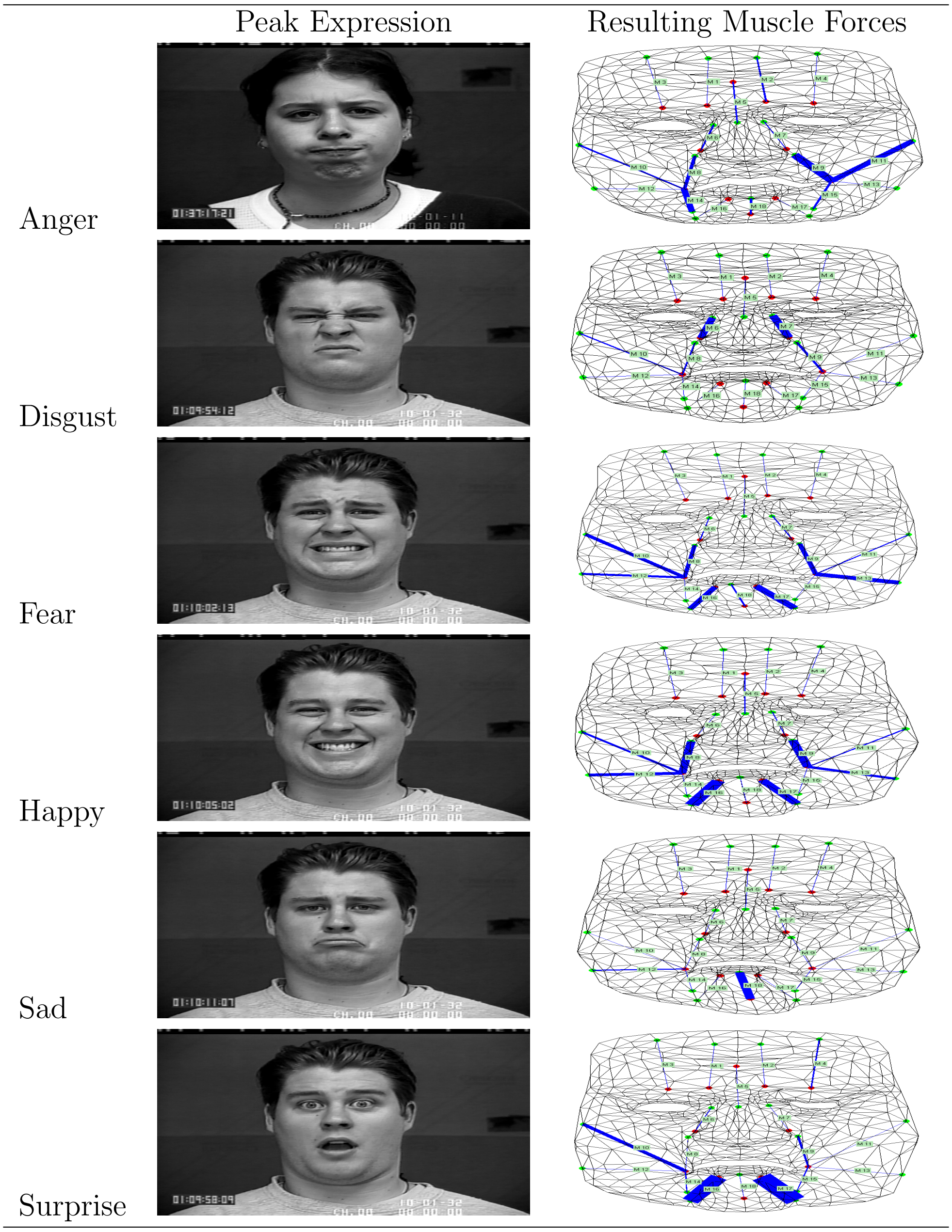


Figure 8.1: Expressions and extracted muscle forces.

8.3 Classification of Seven Basic Expressions

Using the activation levels of facial muscles, we planned to classify seven basic facial expressions. We utilized all available data in the CK database. This database contains 228 labelled image sequences of 6 discrete emotions. Inclusion of the neutral expression expands the size of the dataset to 253 observations. During the experiments we applied leave–one–out technique on all sequences of our database in a cross validation scheme of 253 rounds. Each cross validation round provides us the class label of the peak frame in one sequence. The results of the cross validation rounds are consolidated in a confusion matrix. Classification experiments are performed with 4 classifiers; Linear Discriminant Analysis (LDA), Naive Bayes (NB), k-Nearest Neighbor (kNN) and Support Vector Machine (SVM). Individual performances of these classifiers will be presented in the next sections.

8.3.1 Linear Discriminant Analysis (LDA)

In multi–class LDA distribution of observations is modelled with class means and covariance. LDA projects all the data into a lower dimensional space and maximizes the ratio of between–class variance to within–class variance. In other words, LDA maximizes distances between classes while minimizing the distances within class members. We choose LDA as our first classifier since it is a simple linear classifier, it prevents over tuning on training observations, and it is suitable when observations are not equally distributed to classes.

Table 8.2 presents the results obtained by the LDA classifier. In this confusion matrix rows indicate the actual classes and columns indicate the obtained classes through classification. The diagonal entries show correctly classified expressions and off-diagonal entries correspond to misclassification. The lowest classification performance was obtained for the sadness expression (44 %) where 31 % of examples were incorrectly classified as anger. Fear expression is frequently confused

Table 8.2: Classification results for LDA. A: Anger, D: Disgust, F: Fear, H: Happy, Sa: Sad, Su: Surprise, N: Neutral.

	A	**D**	**F**	**H**	**Sa**	**Su**	**N**	%
A	**18**	1	2	0	2	0	6	62
D	1	**30**	0	0	0	1	2	88
F	0	0	**8**	5	4	0	0	47
H	2	0	6	**53**	0	0	0	87
Sa	5	0	1	0	**7**	0	3	44
Su	1	0	0	0	1	**69**	0	97
N	0	0	0	0	0	0	**25**	100

with happiness and sadness. Similarly, 9.8 % of frames that belong to the happiness class were misclassified as fear. We obtained the best performance in the neutral class (100 %), where we assume that all muscles are idle. The average performance of LDA classification is found as **75 %**.

8.3.2 Naive Bayes (NB)

Naive Bayes classifier is based on Bayes′ rule with independence assumption. Eq. 8.1 presents the Bayes′ rule

$$P(C|F_1, ..., F_n) = \frac{P(C)P(F_1, ..., F_n|C)}{P(F_1, ..., F_n)} \tag{8.1}$$

where C stands for the class and F_1 through F_n stand for feature variables. In the formulation denominator is constant and the numerator is equivalent to the joint probability $P(F_1, ..., F_n|C)$.

Naive Bayes assumes that each feature F_i is conditionally independent of every other feature F_j given the class C. Under this assumption the model has the form

$$\begin{aligned} P(F_1, ..., F_n|C) = P(F_1|F_2, ..., F_n, C)P(F_2, ..., F_n|C) \\ P(F_1|C)P(F_2, ..., F_n|C) \\ P(F_1|C)P(F_2|C)...P(F_n|C) \end{aligned} \tag{8.2}$$

In learning phase using training data Naive Bayes generates conditional probability tables. In testing phase it assigns the test data to the most probable class. Naive Bayes is suitable for non–uniform distribution of samples among classifiers.

Table 8.3: Classification results for Naive Bayes. A: Anger, D: Disgust, F: Fear, H: Happy, Sa: Sad, Su: Surprise, N: Neutral.

	A	**D**	**F**	**H**	**Sa**	**Su**	**N**	%
A	**13**	3	2	1	6	0	4	45
D	0	**31**	0	0	0	1	2	91
F	2	0	**7**	5	3	0	0	41
H	0	0	0	**61**	0	0	0	100
Sa	1	0	0	0	**11**	0	4	69
Su	0	0	0	1	0	**70**	0	99
N	0	0	0	0	0	0	**25**	100

Table 8.3 presents classification results by the Naive Bayes classifier. We obtained the lowest classification accuracy in fear (41 %). It was confused with anger, happiness and sadness. Happy and neutral expressions reached higher accuracy rates (100 %). Observations that belong to the Anger class was confused with all other classes except surprise. The overall success rate of the Naive Bayes classifier is found as **77.9 %**.

8.3.3 k-Nearest Neighbor (kNN)

k-Nearest Neighbor is a non-parametric classifier. It assigns testing data to the class that is most common among its k nearest neighbors. Table 8.4 presents classification results by the kNN classifier. During the experiments k value is taken as 3. As in Naive Bayes, we obtained the lowest classification accuracy in fear (29 %). Significant percentages of fear examples were misclassified as happiness (35.3 %) and sadness (23.5 %). We again obtained the highest accuracy in the neutral expression (100 %). The overall success rate of the kNN classifier is found as **77.3 %**.

Table 8.4: Classification results for kNN. A: Anger, D: Disgust, F: Fear, H: Happy, Sa: Sad, Su: Surprise, N: Neutral.

	A	**D**	**F**	**H**	**Sa**	**Su**	**N**	%
A	**17**	2	1	0	8	1	0	59
D	1	**32**	0	0	0	1	0	94
F	1	0	**5**	6	4	1	0	29
H	0	0	2	**59**	0	0	0	97
Sa	3	0	1	0	**10**	0	2	63
Su	0	0	0	1	0	**70**	0	99
N	0	0	0	0	0	0	**25**	100

8.3.4 Multi–class Support Vector Machine (Multi–class SVM)

Support vector machine finds the optimal hyperplane that separates two classes with the maximum margin. SVM is a binary classifier, but adaptable to multi–class classification problem. We implemented multi–class SVM using one–vs–one (OVO–SVM) strategy [130], training a SVM classifier for each pairwise combinations of classes. For k classes, OVO-SVM approach generates $(k(k-1)/2)$ binary classifiers. Given a set of muscle forces, each binary SVM classifier returns a label of decided class. We feed muscle forces extracted from peak frame of an expression sequence independently to these classifiers. The class that was voted most frequently among these classifiers, in other words the mode of the returned class labels, is decided as the class of the frame.

Table 8.5 presents our results for the multi–class SVM classifier. As in LDA, lowest recognition rate is obtained in sadness expression (75 %). It is most of the times confused with anger and fear expressions. Significant percentage of fear examples were misclassified as happy (23.5 %), conversely 6.5 % of the happy examples were misclassified as fear. The overall classification performance of multi–class SVM is found as **87.1 %**.

Table 8.5: Classification results for multi–class SVM. A: Anger, D: Disgust, F: Fear, H: Happy, Sa: Sad, Su: Surprise, N: Neutral.

	A	**D**	**F**	**H**	**Sa**	**Su**	**N**	%
A	**23**	2	0	0	4	0	0	79
D	2	**31**	0	0	0	1	0	91
F	0	0	**13**	4	0	0	0	77
H	1	2	4	**54**	0	0	0	89
Sa	2	0	2	0	**12**	0	0	75
Su	0	0	1	0	0	**70**	0	99
N	0	0	0	0	0	0	**25**	100

8.4 Discussion

Experiments that are introduced in this section are performed to facilitate comparison with the state of the art studies. As aforementioned in Chapter 2, it is necessary to prepare same experimental set up with other studies for passing judgement on the most successful approach. In the rest of this section we will present a comparative evaluation of our approach with other state–of–the–art algorithms.

Table 8.6 presents facial expression recognition results of seven basic emotions (anger, disgust, fear, happiness, sadness, surprise and neutral). In both of these studies CK database is utilized and leave–one–out testing strategy is followed. These results indicate that we have comparable results with Kotsia et al. [95] and Bartlett et al. [14].

Kotsia et al. classified seven emotions with texture–based (74.3 %) and shape–based features (84.8 %). They improved their classification accuracy by combining features (92.3 %). Compared to our muscle–based features (geometry-based), the features used in this study are more complex (texture-based) whereas their overall performance is lower (74.3 %). The classification performance of muscle–based features is higher than both texture and shape–based features.

Bartlett et al. demonstrated recognition performances of different classification algorithms. As can be seen from the table, muscle–based features and gabor

Table 8.6: Comparison of recent studies in seven class recognition with leave–one–out strategy. Explanation of the abbreviation: ^{n}: neutral class.

Study	Features	Methodology	Success	Emotion	Database
Current study (to appear)	Muscle forces	SVM	87.1 %	7^n	CK
Bartlett et al. [14]	Gabor features	Adaboost SVM AdaSVM LDA_{PCA}	90.1 % 88.0 % 93.3 % 80.7 %	7^n	CK
Kotsia et al. [95]	Texture Shape Combined	DNMF SVM MRBF NN	74.3 % 84.8 % 92.3 %	7^n	CK

features showed similar classification performances by multi–class SVM classifier. However Bartlett et al. increased the classification accuracy to 93.3 % by selecting a subset of gabor features using AdaBoost and classifying them by SVM.

In the second experiment we changed our experimental set up. Instead of extracting muscle forces in peak frames only, we collected the muscle forces that were identified in the last 5 frames of the performed expression. We split the data using 10-fold cross validation method and repeated our experiments with kNN classifiers. We partitioned the data into 10 equal sized sets. 9 of them are used for training the model and the remaining set is used for estimating the performance of the trained model (testing). Experiments are repeated 10 times and each time a randomly selected set is held out for testing. Note that in this experiment, distinct observations from the same expression of a specific subject may appear both in training and testing data sets.

Table 8.7 demonstrates our results for the kNN classifier. We took k value as 3 for making an accurate comparison with the study of Sebe et al. [94]. According to the results, kNN classifier performed best on the neutral expression (100 %). Lowest recognition rate is obtained in anger expression (90 %). Few examples of the anger expression were confused with all other classes except fear. The overall classification performance in classification with kNN is found as **96 %**.

Table 8.7: kNN classification results on seven class recognition with random cross–validation strategy. A: Anger, D: Disgust, F: Fear, H: Happy, Sa: Sad, Su: Surprise, N: Neutral.

	A	**D**	**F**	**H**	**Sa**	**Su**	**N**	%
A	**130**	4	0	2	4	4	1	90
D	4	**164**	0	0	0	2	0	97
F	0	0	**80**	2	1	2	0	94
H	0	0	2	**302**	0	1	0	99
Sa	3	0	1	0	**76**	0	0	95
Su	3	0	1	1	1	**347**	2	98
N	0	0	0	0	0	0	**125**	100

Table 8.8 presents a comparison of our approach with the study of Sebe et al. In both studies CK database and 10-fold cross validation testing strategy is utilized. As can be seen from the table muscle–based features achieve higher accuracy rates than Sebe et al.

Table 8.8: Comparison of recent studies in seven class recognition with random cross–validation strategy. Explanation of the abbreviation: ^{n}: neutral class.

Study	**Features**	**Methodology**	**Success**	**Emotion**	**Database**
Current study (to appear)	Muscle forces	kNN	96.0 %	7^n	CK
Sebe et al. [94]	Bezier volume deformation	kNN	91.8 %	7^n	CK

In our third experiment we extended our data set with the CK+ [36] database. This database contains 327 FACS-annotated image sequences of seven emotions which are performed by 118 different subjects. We collected the muscle forces that were identified in the last frame of the performed expression. We applied leave–one–out testing strategy and classified seven basic expressions (anger, disgust, fear, happiness, sadness, surprise and neutral) with multi–class SVM classifier.

Table 8.9 demonstrates our results for multi–class SVM classifier. Compared to CK experiment, disgust and sadness results are higher. Lowest recognition rate is obtained in fear expression (68 %). It was confused with all other classes except

Table 8.9: Multi–class SVM classification results in seven class recognition problem on the CK+ database. A: Anger, D: Disgust, F: Fear, H: Happy, Sa: Sad, Su: Surprise, N: Neutral.

	A	**D**	**F**	**H**	**Sa**	**Su**	**N**	%
A	**35**	2	2	1	5	0	0	78
D	2	**56**	1	0	0	0	0	95
F	2	1	**17**	2	3	0	0	68
H	2	1	4	**61**	0	1	0	88
Sa	3	0	2	0	**23**	0	0	82
Su	1	0	3	0	0	**79**	0	95
N	0	0	0	0	0	0	**25**	100

surprise and neutral. Significant percentage of anger examples were misclassified as sadness expression (11.1 %), conversely 10.7 % of the sadness examples were misclassified as anger. The overall classification performance is found as **86.6 %**.

In the literature limited number of studies utilize CK+ database. One such study is reported by Lucey et al. [36], in which they classified seven basic emotions (anger, disgust, fear, happiness, sadness, surprise and contempt). For comparing the performances of muscle based features we included the Contempt expression in our dataset and applied leave–one–out cross validation approach.

Table 8.10: Multi–class SVM classification results in seven class recognition problem on the CK+ database. A: Anger, D: Disgust, F: Fear, H: Happy, Sa: Sad, Su: Surprise, C: Contempt.

	A	**D**	**F**	**H**	**Sa**	**Su**	**C**	%
A	**32**	1	3	0	5	0	4	71
D	2	**54**	1	0	0	0	2	92
F	3	1	**15**	2	3	0	1	60
H	2	1	4	**59**	0	1	2	86
Sa	3	0	2	0	**23**	0	0	82
Su	1	0	3	0	0	**78**	1	94
C	5	2	0	2	1	0	**8**	44

Table 8.10 presents our results on CK+ database and including contempt using the multi–class SVM classifier. The multi–class SVM classifier performed best on the surprise expression (94 %). Lowest recognition rate is obtained in contempt

expression (44 %). Specifically, 27.7 % of the contempt examples were misclassified as anger, and 8.8 % of anger examples were misclassified as contempt. The overall classification performance of multi–class SVM is found as **75.5 %**.

Table 8.11 presents a comparative evaluation of our approach with the study of Lucey et al. They classified seven emotions with shape–based (50 %) and appearance–based features (66.7 %). They improved their classification accuracy by combining features (83.3 %). Considering the individual performances of features, muscle–based features achieve better performances than shape–based and appearance–based features alone. However in the case of combined features they obtained significantly better results than us.

Table 8.11: Comparison of recent studies in seven class recognition problem on the CK+ database.

Study	**Features**	**Methodology**	**Success**	**Emotion**	**Database**
Current study (to appear)	Muscle forces	SVM	75.5 %	7	CK+
Lucey et al. [36]	Shape	SVM	50.0 %	7	CK+
	Appearance	SVM	66.7 %		
	Combined	SVM	83.3 %		

Researchers who follow leave–one–out strategy did not report significantly higher success rates than ours on the CK+ dataset. The biggest challenges on this dataset are to discriminate between Anger vs. Sad, Anger vs. Contempt and Happy vs. Fear classes. We can deduce the reasons for these confusions by observing the misclassified examples. Figure 8.2 presents samples of fear and happiness expressions in the CK+ database. In this image, the frames were annotated as Fear, Fear, Happy and Fear, in the same order. Note that the first three frames are hard to distinguish even for a human observer. They are very similar in appearance, especially around the mouth corners. The fear expression of the third subject is significantly different than the others, which is easily distinguishable from her happiness expression.

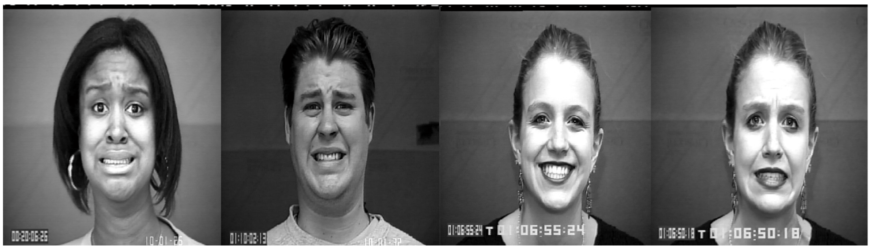


Figure 8.2: Similarity between fear and happiness in the CK+ database. Frames: Fear–Fear–Happy–Fear.

Figure 8.3 presents examples of anger and sadness expressions from the CK+ database. The first two images are annotated as Anger and the last one is annotated as Sad by FACS coders. Note that these expressions are also challenging examples for human observers.

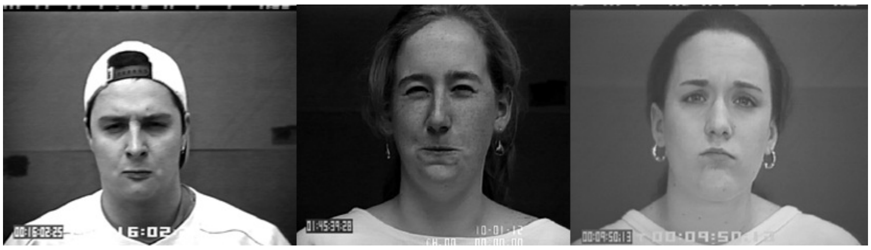


Figure 8.3: Similarity between anger and sadness in the CK+ database. Frames: Anger–Anger–Sad.

Classification performance on seven basic facial expressions (anger, disgust, fear, happiness, sadness, surprise and neutral) with muscle forces as features has been in the range **75-87 %**. This performance is close to the human ceiling of recognition [86, 131] and comparable to the results of the state–of–the–art algorithms that use geometric, appearance or FACS based features in classification.

A recent study by Goeleven et al. [132] provides a good benchmark for validating the performance of automated expression recognition algorithms. This study reports the recognition accuracy of human subjects on basic expressions. They used Karolinska Directed Emotional Faces database, a very similar database to CK+ that includes all expressions other than Contempt. The recognition rates of humans and our algorithm are presented in Table 8.12. These figures indicate that use of muscle based features with SVM classifier is at least as successful as humans in facial expression recognition.

Table 8.12: Performance comparison for human observers [132] and the proposed method.

	A	**D**	**F**	**H**	**Sa**	**Su**	**N**
Human	79 %	72 %	43 %	**93 %**	**77 %**	77 %	63 %
Muscle–based features with SVM	79 %	**91 %**	**77 %**	89 %	75 %	**99 %**	**100 %**

As a final remark our results indicate that features based on facial anatomy and muscle activation levels match up with state–of–the–art algorithmic approaches and human observers.

Conclusion

Action Units of FACS and geometric features derived from them have been used frequently in facial expression recognition research. Many FACS AUs correspond to a compound effect of multiple facial muscles. This characteristic of FACS AUs makes identification and scoring of a facial action an intricate task both for human experts and computers. Specifically, (1) it is hard to identify subtle facial activities, (2) AUs restrict the analysis to psychologically known mechanisms of emotions, (3) it is hard to identify the individual action units in compound expressions and (4) it is hard to identify feature points when interpersonal variations are present.

In this thesis we propose new features that are based on muscle forces composing all facial expressions under the constraints of facial anatomy. We aim to determine the muscle activations through a mapping from displacement of facial features to activation levels of facial muscles. Our aim in this study is to be able to automatically recognize six universal facial expressions with same or better accuracy of best practices in the literature. Our contribution is a set of new features that are based on the facial anatomy.

Our proposed feature extraction system consists of; (1) semi–automatic customization of the face model to the subject, (2) identification and tracking of facial features that reside in the region of influence of a muscle, (3) estimation of head orientation and alignment of the face model with the observed face, (4) estimation of relative displacements of vertices that produce facial expressions, and (5) solving vertex displacements to obtain muscle forces.

The proposed feature extraction system obtains its strength from customization of a generic anatomical model to subject's face and tracking of multiple points on muscular regions of influence. The generic face model embeds prior knowledge in the anatomy of the human face and customization enables us to precisely locate muscles on a given face.

In this study, we model human faces with a generic wireframe (HIGEM), which consists of 612 vertices and 1,128 faces. We define 18 major muscles (features) based on the anatomy of the human face. We model muscles as linear fibers and compute muscles regions of influence contraction of a muscle affects all vertices of the wireframe that are within its region of influence.

In customization stage we focus on registering a 3D generic wireframe model with an input face image. In this stage our input is a set of facial landmarks in the 2D video frame. The problem of model customization is to find a mapping from 2D inputs to 3D space. We estimate the 3D coordinates of the facial landmarks on the camera view plane using ray tracing method.

We propose a semi-automatic wireframe fitting algorithm for customizing a high-polygon wireframe model. In the first step we manually mark 32 facial landmarks on both of the face image and wireframe model. We apply an initial alignment to the wireframe model. Next, we utilize ray tracing method for estimating the 3D coordinates of the facial landmarks. Finally, we calculate the new positions of the vertices, that are not manually labelled as landmark, with the proposed *nearest neighbors weighted average customization* algorithm. We also provide a comparative analysis of NNWA algorithm with Procrustes analysis and Active Shape Model. We demonstrate that the performance of the proposed algorithm is comparable with state of the art algorithms in face modelling such as ASM.

Feature points that are tracked on video frame are used to determine 3D rigid body motion of the head and deformations on wireframe model. We choose vertices that exist in the muscle influence regions as feature points. However in tracking phase feature points may drift away from their correct positions. This drifting problem will affect the accuracy of the alignment and deformation calculations. For identifying and solving the drifting problem we propose a cross product based method and examined state (topology) changes of each feature point. All feature points that undergo a state change are marked as *untrusted*. The remaining feature points and their corresponding wireframe vertices are utilized for estimating the orientation of the head. We implement a greedy search algorithm to find a solution for 3D transformation of the model to match with the observation.

The deviations between the *trusted* tracking points and the projections of corresponding vertices in the aligned face model serve as indicators of facial expressions. We apply ray tracing to extract the displacements of vertices due to facial expressions. In the customization stage we assumed that wireframe vertices have fixed depths. However, it is not possible to utilize this assumption in facial deformation estimation stage. We identify plane of motion for each of the *trusted* vertex and estimate its new coordinates through line–plane intersection. We also recompute coordinates of the *untrusted* vertices by taking the average of its neighbors.

We model human face as a 3D surface that is composed of polygons. Each face on the wireframe model is defined by three vertices. The edges between neighboring vertices are modelled with springs. External forces (muscle forces) on the vertices are computed with Hooke′s law. We developed the stiffness matrix of the entire face model and solved muscle activations using constrained least squares method.

We demonstrate the representative power of the proposed features on four classifiers; LDA, NB, kNN and SVM. The best performance on the classification problem of seven expressions including neutral was 87.1 %, obtained by use of SVM. The results we attained in this study are close to the human recognition ceiling of 87-91.7 % and comparable with the state–of–the–art algorithms in the literature. Up to 10 % increase in performance with the SVM classifier indicates a complex decision boundary between expression classes as Sebe et al. [94] noted. In this dissertation we proposed and experimentally showed that muscle activations based features are good discriminators for facial expression recognition.

References

[1] A. Mehrabian, "Communication without words," *Psychol. Today*, vol. 2, no. 9, pp. 52–55, 1968.

[2] H. Gu and Q. Ji, "Information extraction from image sequences of real-world facial expressions," *Machine Vision and Applications*, vol. 16, no. 2, pp. 105+, 2005.

[3] E. Vural, M. S. Bartlett, G. Littlewort, M. Çetin, A. Erçil, and J. R. Movellan, "Discrimination of moderate and acute drowsiness based on spontaneous facial expressions," in *20th International Conference on Pattern Recognition, ICPR 2010, Istanbul, Turkey, 23-26 August 2010.* IEEE, 2010, pp. 3874–3877.

[4] G. de Boulogne and R. Cuthbertson, *The Mechanism of Human Facial Expression*, ser. Studies in Emotion and Social Interaction. Cambridge University Press, 1990.

[5] C. Darwin, *The Expression of the Emotions in Man and Animals*, anniversary ed., P. Ekman, Ed. Harper Perennial, 1872/2009.

[6] P. Ekman and W. Friesen, *Facial Action Coding System: A Technique for the Measurement of Facial Movement.* Palo Alto: Consulting Psychologists Press, 1978.

[7] B. Braathen, M. S. Bartlett, G. Littlewort, E. Smith, and J. R. Movellan, "An approach to automatic recognition of spontaneous facial actions," in *Advances in Neural Information Processing Systems, Number 15.* MIT Press, 2002.

[8] M. Pantic and M. Bartlett, "Machine analysis of facial expressions," in *Face Recognition*, K. Delac and M. Grgic, Eds. Vienna, Austria: I-Tech Education and Publishing, July 2007, pp. 377–416.

[9] J. A. Coan and J. J. Allen, *Handbook of Emotion Elicitation and Assessment*. Oxford University Press, 2007.

[10] I. Kotsia and I. Pitas, "Facial expression recognition in image sequences using geometric deformation features and support vector machines," *Image Processing, IEEE Transactions on*, vol. 16, no. 1, pp. 172–187, 2007.

[11] M. Pantic and L. J. M. Rothkrantz, "Expert system for automatic analysis of facial expressions," 2000.

[12] J. Lien, T. Kanade, J. Cohn, and C.-C. Li, "Automated facial expression recognition based on facs action units," in *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, 1998, pp. 390–395.

[13] Y. li Tian, T. Kanade, and J. F. Cohn, "Recognizing action units for facial expression analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 2, pp. 97–115, 2001.

[14] M. Bartlett, G. Littlewort, C. Lainscsek, I. Fasel, and J. Movellan, "Machine learning methods for fully automatic recognition of facial expressions and facial actions," in *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, vol. 1, 2004, pp. 592–597 vol.1.

[15] I. A. Essa and A. Pentland, "Coding, analysis, interpretation, and recognition of facial expressions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 757–763, 1997.

[16] S. Kimura and M. Yachida, "Facial expression recognition and its degree estimation," in *1997 Conference on Computer Vision and Pattern Recognition (CVPR 97), June 17-19, 1997, San Juan, Puerto Rico.* IEEE Computer Society, 1997, pp. 295–300.

[17] J. J.-J. Lien, T. Kanade, J. F. Cohn, and C.-C. Li, "Subtly different facial expression recognition and expression intensity estimation," in *1998 Conference on Computer Vision and Pattern Recognition (CVPR 98), June 23-25, 1998, Santa Barbara, CA, USA.* IEEE Computer Society, 1998, pp. 853–859.

[18] Y. li Tian, T. Kanade, and J. F. Cohn, "Eye-state action unit detection by gabor wavelets," in *Advances in Multimodal Interfaces - ICMI 2000, Third International Conference, Beijing, China, October 14-16, 2000, Proceedings*, ser. Lecture Notes in Computer Science, vol. 1948. Springer, 2000, pp. 143–150.

[19] M. Pantic and I. Patras, "Dynamics of facial expression: Recognition of facial actions and their temporal segments from face profile image sequences," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 36, no. 2, pp. 433–449, 2006.

[20] M. F. Valstar, M. Pantic, Z. Ambadar, and J. F. Cohn, "Spontaneous vs. posed facial behavior: automatic analysis of brow actions," in *Proceedings of the 8th International Conference on Multimodal Interfaces, ICMI 2006, Banff, Alberta, Canada, November 2-4, 2006.* ACM, 2006, pp. 162–170.

[21] E. Goldfinger, *Human Antomy for Artists: The Elements of Form.* Oxford University Press, 1991.

[22] A. K. Jain and S. Z. Li, *Handbook of Face Recognition.* Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2005.

[23] S. Park and D. Kim, "Subtle facial expression recognition using motion magnification," *Pattern Recognition Letters*, vol. 30, no. 7, pp. 708–716, 2009.

[24] V. Bruce and A. Young, "Understanding face recognition." *British journal of psychology (London, England : 1953)*, vol. 77 (Pt 3), pp. 305–327, Aug. 1986.

[25] M. W. Eysenck, *Principles of Cognitive Psychology.* Hove, UK: Psychology Press, 2001.

[26] H. Mcgurck and J. W. Macdonald, "Hearing lips and seeing voices," *Nature*, vol. 264, pp. 746–748, 1976.

[27] A. J. Calder, A. M. Burton, P. Miller, A. W. Young, and S. Akamatsu, "A principal component analysis of facial expressions," *Vision Research*, vol. 41, pp. 1179–1208, 2001.

[28] K. Humphreys, G. Avidan, and M. Behrmann, "A detailed investigation of facial expression processing in congenital prosopagnosia as compared to acquired prosopagnosia," *Experimental Brain Research*, vol. 176, no. 2, pp. 356–373, 2007.

[29] B. Fasel and J. Luettin, "Automatic Facial Expression Analysis: A Survey," *Pattern Recognition*, vol. 36, no. 1, pp. 259–275, 2003.

[30] D. S. Messinger, A. Fogel, K. L. Dickson *et al.*, "What's in a smile?" *Developmental Psychology*, vol. 35, no. 3, pp. 701–708, 1999.

[31] M. Pantic and I. Patras, "Detecting facial actions and their temporal segments in nearly frontal-view face image sequences," in *Systems, Man and Cybernetics, 2005 IEEE International Conference on*, vol. 4, 2005, pp. 3358–3363 Vol. 4.

[32] M. Bartlett, P. Viola, T. Sejnowski, J. Larsen, J. Hager, and P. Ekman, "Classifying facial action," in *Advances in Neural Information Processing Systems*, vol. 8, 1996, pp. 823–829.

[33] M. Pantic, L. Rothkrantz, and H. Koppelaar, "Automation of non-verbal communication of facial expressions," in *Proc. Conf. Euromedia*, 1998, pp. 86–93.

[34] B. Joosten, "Facial Expression Recognition (Towards digital support for behavioral scientists)," Master's thesis, Tilburg University, 2011.

[35] T. Kanade, J. Cohn, and Y.-L. Tian, "Comprehensive database for facial expression analysis," in *Proceedings of the 4th IEEE International Conference on Automatic Face and Gesture Recognition (FG'00)*, March 2000, pp. 46 – 53.

[36] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression," in *Computer Vision and Pattern Recognition*, 2010.

[37] "PICS database," `http://pics.psych.stir.ac.uk/`.

[38] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with gabor wavelets," 1998, pp. 200–205.

[39] A. Martínez and R. Benavente, "The AR face database," Computer Vision Center, Tech. Rep. 24, Jun 1998.

[40] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, pp. 1615–1618, 2003.

[41] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-PIE," *Image Vision Comput.*, vol. 28, no. 5, pp. 807–813, May 2010.

[42] M. Pantic, M. Valstar, R. Rademaker, and L. Maat, "Web-based database for facial expression analysis," in *Proc. IEEE Int'l Conf. Multimedia and Expo*, 2005, pp. 317–321.

[43] M. F. Valstar and M. Pantic, "Induced Disgust, Happiness and Surprise: an Addition to the MMI Facial Expression Database," in *Proceedings of Int'l Conf. Language Resources and Evaluation, Workshop on EMOTION*, Malta, May 2010, pp. 65–70.

[44] M. S. Bartlett, G. Littlewort, M. G. Frank, C. Lainscsek, I. R. Fasel, and J. R. Movellan, "Automatic recognition of facial actions in spontaneous expressions," *Journal of Multimedia*, vol. 1, no. 6, pp. 22–35, 2006.

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

[46] C. Beumier and M. Acheroy, "Face verification from 3d and grey level clues," *Pattern Recognition Letters*, vol. 22, no. 12, pp. 1321–1329, 2001.

[47] A. Savran, N. Alyuz, H. Dibeklioglu, O. Celiktutan, B. Gokberk, B. Sankur, and L. Akarun, "Bosphorus database for 3D face analysis," in *Proceedings of the First COST 2101 Workshop on Biometrics and Identity Management (BIOD)*, Denmark, May 2008.

[48] L. Yin, X. Wei, Y. Sun, J. Wang, and M. J. Rosato, "A 3d facial expression database for facial behavior research," in *Proc. IEEE Int'l Conf. Face and Gesture Recognition*, 2006, pp. 211–216.

[49] L. Yin, X. Chen, Y. Sun, T. Worm, and M. Reale, "A high-resolution 3d dynamic facial expression database," in *Automatic Face Gesture Recognition, 2008. FG '08. 8th IEEE International Conference on*, 2008, pp. 1–6.

[50] S. Li, Z. Lei, and M. Ao, "The HFB face database for heterogeneous face biometrics research," in *Computer Vision and Pattern Recognition Workshops, 2009. CVPR Workshops 2009. IEEE Computer Society Conference on*, 2009, pp. 1–8.

[51] W. Gao, B. Cao, S. Shan, X. Chen, D. Zhou, X. Zhang, and D. Zhao, "The cas-peal large-scale chinese face database and baseline evaluations," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 38, no. 1, pp. 149–161, 2008.

[52] P. Phillips, K. Bowyer, T. Scruggs, E. Ortiz, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, "Overview of the face recognition grand challenge," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, 2005, pp. 947–954 vol. 1.

[53] A. B. Moreno and A. Sánchez, "GavabDB: a 3D Face Database," in *Workshop on Biometrics on the Internet*, Vigo, 2004, pp. 77–85.

[54] N. F. Troje and H. H. Bülthoff, "Face recognition under varying poses: The role of texture and shape," *Vision Research*, vol. 36, pp. 1761–1771, 1996.

[55] T. Faltemier, E. Ortiz, and K. Bowyer, "Using a multi-instance enrollment representation to improve 3d face recognition," in *Biometrics: Theory, Applications, and Systems, 2007. BTAS 2007. First IEEE International Conference on*, 2007, pp. 1–6.

[56] S. Zafeiriou, M. Hansen, G. Atkinson, V. Argyriou, M. Petrou, M. Smith, and L. Smith, "The photoface database," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*, 2011, pp. 132–139.

[57] S. Gupta, K. Castleman, M. Markey, and A. Bovik, "Texas 3d face recognition database," in *Image Analysis Interpretation (SSIAI), 2010 IEEE Southwest Symposium on*, 2010, pp. 97–100.

[58] K. Messer, J. Matas, J. Kittler, J. Lüttin, and G. Maitre, "Xm2vtsdb: The extended m2vts database," in *Second International Conference on Audio and Video-based Biometric Person Authentication*, 1999, pp. 72–77.

[59] T. Heseltine, N. Pears, and J. Austin, "Three-dimensional face recognition using combinations of surface feature map subspace components," *Image and Vision Computing*, vol. 26, no. 3, pp. 382 – 396, 2008.

[60] H. A. Rowley, S. Member, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions On Pattern Analysis and Machine intelligence*, vol. 20, pp. 23–38, 1998.

[61] H. Schneiderman and T. Kanade, “A statistical model for 3d object detection applied to faces and cars,” in *IEEE Conference on Computer Vision and Pattern Recognition*, June 2000, pp. 1746–1759.

[62] M. hsuan Yang, D. Roth, and N. Ahuja, “A snow-based face detector,” in *Advances in Neural Information Processing Systems 12*. MIT Press, 2000, pp. 855–861.

[63] S. Romdhani, P. Torr, B. Scholkopf, and A. Blake, “Computationally efficient face detection,” in *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 2, 2001, pp. 695–700 vol.2.

[64] P. Viola and M. Jones, “Robust real-time face detection,” *International Journal of Computer Vision*, vol. 57, pp. 137–154, 2004.

[65] M. F. Valstar and M. Pantic, “Combined support vector machines and hidden markov models for modeling facial action temporal dynamics,” in *Human-Computer Interaction, IEEE International Workshop, HCI 2007, Rio de Janeiro, Brazil, October 20, 2007, Proceedings*, ser. Lecture Notes in Computer Science, M. S. Lew, N. Sebe, T. S. Huang, and E. M. Bakker, Eds., vol. 4796. Springer, 2007, pp. 118–127.

[66] A. A. Seyedarabi, H. and S. Khanmohammadi, “Analysis and synthesis of facial expressions by feature-points tracking and deformable model.” *Journal of Iranian Association of Electrical and Electronic Engieers (IAEEE)*, vol. 4, no. 1, 2007.

[67] J.-Y. Bouguet, “Pyramidal implementation of the lucas kanade feature tracker,” *Intel Corporation, Microprocessor Research Labs*, 2000.

[68] K. Lu and X. Zhang, “Facial expression recognition from image sequences based on feature points and canonical correlations,” in *Artificial Intelligence and Computational Intelligence (AICI), 2010 International Conference on*, vol. 1, 2010, pp. 219–223.

[69] M. F. Valstar and M. Pantic, "Fully automatic recognition of the temporal phases of facial actions," *IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics*, vol. 42, no. 1, pp. 28–43, February 2012.

[70] M. F. Valstar, M. Pantic, and I. Patras, "Motion history for facial action detection in video," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics: The Hague, Netherlands, 10-13 October 2004*, 2004, pp. 635–640.

[71] G. Guo and C. R. Dyer, "Learning from examples in the small sample case: face expression recognition," *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, vol. 35, no. 3, pp. 477–488, 2005.

[72] M. S. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. R. Fasel, and J. R. Movellan, "Recognizing facial expression: Machine learning and application to spontaneous behavior," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA*, 2005, pp. 568–573.

[73] M. S. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan, "Fully automatic facial action recognition in spontaneous behavior," in *in 7th International Conference on Automatic Face and Gesture Recognition*, 2006, pp. 223–230.

[74] J. Whitehill and C. W. Omlin, "Haar features for facs au recognition," in *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, ser. FGR '06, 2006, pp. 97–101.

[75] B. Jiang, M. F. Valstar, and M. Pantic, "Action unit detection using sparse appearance descriptors in space-time video volumes," in *Ninth IEEE International Conference on Automatic Face and Gesture Recognition (FG 2011), Santa Barbara, CA, USA, 21-25 March 2011.* IEEE, 2011, pp. 314–321.

[76] Z. Zhang, M. J. Lyons, M. Schuster, and S. Akamatsu, "Comparison between geometry-based and gabor-wavelets-based facial expression recognition using multi-layer perceptron," in *FG*. IEEE Computer Society, 1998, pp. 454–461.

[77] J. Chen, D. Chen, Y. Gong, M. Yu, K. Zhang, and L. Wang, "Facial expression recognition using geometric and appearance features," in *Proceedings of the 4th International Conference on Internet Multimedia Computing and Service*, ser. ICIMCS '12, 2012, pp. 29–33.

[78] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models-their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.

[79] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," in *Computer Vision - ECCV 98, 5th European Conference on Computer Vision, Freiburg, Germany, June 2-6, 1998, Proceedings, Volume II*, ser. Lecture Notes in Computer Science, vol. 1407, 1998, pp. 484–498.

[80] L. Gang, L. Xiao-hua, Z. Ji-Liu, and G. Xiao-gang, "Geometric feature based facial expression recognition using multiclass support vector machines," in *Granular Computing, 2009, GRC '09. IEEE International Conference on*, 2009, pp. 318–321.

[81] S. Lucey, I. Matthews, C. Hu, Z. Ambadar, F. D. la Torre, and J. F. Cohn, "AAM derived face representations for robust facial action recognition," in *Seventh IEEE International Conference on Automatic Face and Gesture Recognition (FG 2006), 10-12 April 2006, Southampton, UK*, 2006, pp. 155–162.

[82] M. Pantic and L. Rothkrantz, "Facial action recognition for facial expression analysis from static face images," *IEEE Transactions on Systems, Man and Cybernetics - Part B*, vol. 34, no. 3, pp. 1449–1461, 2004.

[83] G. Donato, M. S. Bartlett, J. C. Hager, P. Ekman, and T. J. Sejnowski, "Classifying facial actions," *IEEE Trans. Pattern Anal and Machine Intell*, pp. 974–989, 1999.

[84] M. J. Lyons, J. Budynek, and S. Akamatsu, "Automatic classification of single facial images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 12, pp. 1357–1362, 1999.

[85] Y. Zhang and Q. Ji, "Facial expression understanding in image sequences using dynamic and active visual information fusion," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, 2003, pp. 1297–1304 vol.2.

[86] P. Ekman and W. V. Friesen, *Pictures of Facial Affect.* Consulting Psychologists Press, 1976.

[87] W. A. Fellenz, J. G. Taylor, N. Tsapatsoulis, and S. Kollias, "Comparing template-based, feature-based and supervised classification of facial expressions from static images," in *Proceedings of Circuits, Systems, Communications and Computers (CSCC'99)*, 1999, pp. 5331–5336.

[88] G. Littlewort, I. Fasel, M. S. Bartlett, and J. R. Movellan, "Fully automatic coding of basic expressions from video," Tech. rep.(2002) U of Calif., S.Diego, INC MPLab, Tech. Rep., 2002.

[89] Z. Wen and T. Huang, "Capturing subtle facial motions in 3d face tracking," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, 2003, pp. 1343–1350 vol.2.

[90] M. G. Frank and P. Ekman, "The ability to detect deceit generalizes across different types of high-stake lies," *Journal of Personality and Social Psychology*, vol. 72, pp. 1429–1439, 1997.

[91] M. Bartlett, B. Braathen, G. Littlewort-Ford, J. Hershey, I. Fasel, T. Marks, E. Smith, T. Sejnowski, and J. R. Movellan, “Automatic analysis of spontaneous facial behavior: A final project report,” Machine Perception Lab, Institute for Neural Computation, University of California, San Diego., Tech. Rep., 2001.

[92] J. F. Cohn, T. Kanade, T. Moriyama, Z. Ambadar, J. Xiao, J. Gao, and H. Imamura, “A comparative study of alternative facs coding algorithms,” Robotics Institute, Carnegie Mellon University, Pittsburgh, Tech. Rep., 2001.

[93] T. Moriyama, T. Kanade, J. F. Cohn, J. Xiao, Z. Ambadar, J. Gao, and H. Imamura, “Automatic recognition of eye blinking in spontaneously occurring behavior,” in *Proceedings of the 16th International Conference on Pattern Recognition (ICPR ’2002*, 2002, pp. 78–81.

[94] N. Sebe, M. S. Lew, Y. Sun, I. Cohen, T. Gevers, and T. S. Huang, “Authentic facial expression analysis,” *Image Vision Comput.*, vol. 25, no. 12, pp. 1856–1863, 2007.

[95] I. Kotsia, S. Zafeiriou, and I. Pitas, “Texture and shape information fusion for facial expression and facial action unit recognition,” *Pattern Recognition*, vol. 41, no. 3, pp. 833–851, 2008.

[96] S. Park, H.-S. Lee, J. Shin, and D. Kim, “The postech subtle facial expression database 2007 (sfed07),” in *Proc, of the 8th POSTECH-KYUTECH Joint Workshop On Neuroinformatics*, 2008, pp. 653–54.

[97] J. Sung and D. Kim, “Real-time facial expression recognition using staam and layered gda classifier,” *Image Vision Comput.*, vol. 27, no. 9, pp. 1313–1325, 2009.

[98] M. DeMello, *Faces Around the World: A Cultural Encyclopedia of the Human Face.* ABC-CLIO, 2012.

[99] M. Rydfalk, “Candide, a parameterized face,” Dept. of Electrical Engineering, Linköping University, Tech. Rep., 1987.

[100] B. Welsh, “Model-based coding of images,” Ph.D. dissertation, British Telecom Research Lab, 1991.

[101] J. Ahlberg, “Candide-3 - an updated parameterised face,” Dept. of Electrical Engineering, Linköping University, Tech. Rep., 2001.

[102] F. Parke, “Parameterized models for facial animation,” *Computer Graphics and Applications, IEEE*, vol. 2, no. 9, pp. 61–68, 1982.

[103] K. Waters, “A muscle model for animation three-dimensional facial expression,” in *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, ser. SIGGRAPH ’87, 1987, pp. 17–24.

[104] D. Terzopoulos and K. Waters, “Physically-based facial modelling, analysis, and animation,” *The Journal of Visualization and Computer Animation*, pp. 73–80, 1990.

[105] D. Terzopoulos and K. Waters, “Analysis and synthesis of facial image sequences using physical and anatomical models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 569–579, 1993.

[106] F. Erol, “Modeling and animating personalized faces,” Master’s thesis, Bilkent University, 2002.

[107] I. Essa and A. Pentland, “A vision system for observing and extracting facial action parameters.” Massachusetts Institute of Technology, Perceptual Computing Section, Tech. Rep., 1994.

[108] S. M. Platt and N. I. Badler, “Animating facial expressions,” in *Proceedings of the 8th annual conference on Computer graphics and interactive techniques*, ser. SIGGRAPH ’81, 1981, pp. 245–252.

[109] K.Waters and D. Terzopoulos, “Modeling and animating faces using scanned data,” *The Journal of Visualization and Computer Animation*, pp. 123–128, 1991.

[110] S. Pieper, J. Rosen, and D. Zeltzer, “Interactive graphics for plastic surgery: a task-level analysis and implementation,” in *Proceedings of the 1992 symposium on Interactive 3D graphics*, 1992, pp. 127–134.

[111] G. Breton, C. Bouville, and D. Pelé, “Faceengine a 3d facial animation engine for real time applications,” in *Proceedings of the sixth international conference on 3D Web technology*, ser. Web3D ’01, 2001, pp. 15–22.

[112] Y. Zhang, E. C. Prakash, and E. Sung, “A new physical model with multilayer architecture for facial expression animation using dynamic adaptive mesh,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 10, no. 3, pp. 339–352, 2004.

[113] E. S. Y. Zhang, E.C. Prakash, “Face alive,” *Journal of Visual Languages and Computing*, pp. 125–160, 2004.

[114] M. Eskil, “High polygon generic wireframe model – HIGEM,” http://pi.isikun.edu.tr/, Jan. 2013.

[115] B. Moghaddam and A. Pentland, “Face recognition using view-based and modular eigenspaces,” in *In Automatic Systems for the Identification and Inspection of Humans, SPIE*, 1994, pp. 12–21.

[116] L. Lu, Z. Zhang, H.-Y. Shum, Z. Liu, and H. Chen, “Model and Exemplar-based Robust Head Pose Tracking Under Occlusion and Varying Expression,” in *Computer Vision and Pattern Recognition*, 2001, pp. 1–8.

[117] J. Ahlberg, “An active model for facial feature tracking,” *EURASIP Journal on Applied Signal processing*, vol. 2002, pp. 566–571, 2001.

[118] F. Dornaika and J. Ahlberg, "Face model adaptation for tracking and active appearance model training," in *British Machine Vision Conference*, 2003, pp. 57.1–57.10.

[119] F. Dornaika and J. Ahlberg, "Fast and reliable active appearance model search for 3-d face tracking," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 34, no. 4, pp. 1838–1853, 2004.

[120] S. Krinidis and I. Pitas, "Facial expression synthesis through facial expressions statistical analysis." in *European Signal Processing Conference (EU-SIPCO06)*, 2006.

[121] I. Kotsia, I. Buciu, and I. Pitas, "An analysis of facial expression recognition under partial facial image occlusion," *Image and Vision Computing*, vol. 26, no. 7, pp. 1052 – 1067, 2008.

[122] N. Vretos, N. Nikolaidis, and I. Pitas, "A model-based facial expression recognition algorithm using principal components analysis," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, 2009, pp. 3301–3304.

[123] S. Krinidis and I. Pitas, "Statistical analysis of human facial expressions," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 1, no. 3, 2010.

[124] Y. Luximon, R. Ball, and L. Justice, "The 3D chinese head and face modeling," *Comput. Aided Des.*, vol. 44, no. 1, pp. 40–47, Jan. 2012.

[125] J. Gower, "Generalized procrustes analysis," *Psychometrika*, vol. 40, pp. 1–10, 1966.

[126] J. Gower and G. B. Dijksterhuis, *Procrustes problems.* Oxford University Press, 2004.

[127] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th International Joint

Conference on Artificial Intelligence (IJCAI 81), Vancouver, BC, Canada, August 1981, 1981, pp. 674–679.

[128] B. Horn and B. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, no. 1-3, pp. 185–203, 1981.

[129] F. Dornaika and J. Ahlberg, "Fitting 3d face models for tracking and active appearance model training," *Image Vision Comput.*, vol. 24, no. 9, pp. 1010–1024, 2006.

[130] T. Guneş and E. Polat, "Yüz İfade Analizinde Öznitelik Seçimi Ve Çoklu SVM Sınıflandırıcılarına Etkisi," *Journal of the Faculty of Engineering and Architecture of Gazi University*, vol. 24, pp. 7–14, 2009.

[131] J. N. Bassili, "Facial motion in the perception of faces and of emotional expression." *J. Exp. Psychol. Human*, vol. 4, no. 3, pp. 373–379, Aug. 1978.

[132] E. Goeleven, R. De Raedt, L. Leyman, and B. Verschuere, "The karolinska directed emotional faces: A validation study," *Cognition Emotion*, vol. 22, pp. 1094–1118, 2008.

Curriculum Vitae