EIT FAKULTÄT FÜR ELEKTROTECHNIK UND INFORMATIONSTECHNIK

AUTOMATIC FACIAL ANALYSIS METHODS: FACIAL POINT LOCALIZATION, HEAD POSE ESTIMATION, AND FACIAL EXPRESSION RECOGNITION

Dissertation

zur Erlangung des akademischen Grades

Doktoringenieur

(Dr.-Ing.)

von **M.Sc. Anwar Maresh Qahtan Saeed**

geb. am 10. Juni 1980 in Taiz, Jemen

genehmigt durch die Fakultät für Elektrotechnik und Informationstechnik

der Otto-von-Guericke-Universität Magdeburg

Gutachter:

apl. Prof. Dr.-Ing. habil. Ayoub Al-Hamadi

Prof. Dr. rer. nat. Andreas Wendemuth

Prof. Dr. Bogdan Matuszewski

Promotionskolloquium am: 26. März 2018

With love and gratitude, this work dedicated to my family who are my motivation and life.

Anwar

Abstract

Facial analysis via camera systems gains an increasing attention due to the non-intrusive nature of the cameras. Accordingly, it has been employed in various applications ranging from entertainment as video games, medical purpose as pain assessment, and security as surveillance. This dissertation addresses three tasks concerning the facial analysis: facial point localization, head pose estimation, and facial expression recognition. The proposed methods here are frame-based and fully automatic, which starts by locating the face within the processed frame.

Neural networks in a cascade framework are exploited here to locate 49 facial points within a detected face patch. The localization process takes place over five neural networks; four refinement networks follow a guided initialization in the first network. A feature selection is performed before each neural network, boosting the algorithm generalization capability. This framework locates the facial points with an average error for each point ranging between 0.72% and 1.57% of the face width. Further conducted evaluations and comparisons prove the competitiveness of the proposed approach in terms of accuracy and efficiency besides its better generalization capability.

Additionally, I propose a framework to estimate the head pose of a face depicted in DRGB frame; it is configurable to work on only RGB frame as well. The pose estimation is boosted by the deployment of the depth data in extracting additional depth-based features or in performing depth-based face cropping. With the latter method, I achieved an accurate pose estimation with average error rates of

$4.19°$, $3.84°$, and $4.13°$ for pitch, yaw, and roll rotation angles, respectively, based on cross-validation conducted on a public database. This approach generalizes almost perfectly to another public database, where a pose estimation with average rates of $4.23°$, $4.64°$, and $4.33°$ for pitch, yaw, and roll rotation angles, respectively, was achieved based on cross-database evaluation. These results are more accurate in comparison to results stemmed from corresponding state-of-the-art approaches.

Moreover to recognize the facial expression in single frames, I propose three approaches: a geometry, an appearance based besides, and a hybrid of them both. I utilize an earlier locating of 49 facial points to recognize the facial expression in person dependent and independent scenarios. The displacement of these points to their location in person-specific or general neutral model is considered as a major cue for the expression recognition; the displacement is always evaluated with respect to the face configuration (measured from non-movable facial points). Personalized features lead always to a better recognition rate by at least 3%. Recognizing the expression via a geometry-based approach of only 8 facial points is also proposed. Using it, I achieved an average recognition rate of 87.48%, lower by only 2.24% in comparison to the results using 49 points. Three appearance-based feature types and 4 different classifiers were investigated within the appearance-based framework for the facial expression recognition. With histogram of orientation gradients and Support Vector Machine classifier, I achieved the best average recognition rates: 87.26% and 83.71% for the person independent 6-class and 7-class cases, respectively. The latter recognition rate was improved to 89.14% by using a proposed framework to joint facial expression recognition and facial point localization. This framework exploits both geometry- and appearance- based methods for the expression recognition, and both cascade-regression and local-based methods for the facial point localization. The accuracy of the points localization was enhanced as well in comparison to the isolated methods. The proposed methods here outperform state-of-the-art approaches that utilize a similar evaluation protocol. The geometry-based methods generalize across databases better than the appearance-based methods, as empirically proven.

Zusammenfassung

Gesichtsanalyse mit Kamerasystemen wird wegen der nicht-intrusiven Eigenschaft von Kameras einer steigenden Aufmerksamkeit zuteil. Daher wurden bereits Vielfältige Anwendungen adressiert - von Videospielen, über Medizin (Schmerzerkennung) und Sicherheit (Überwachung). Diese Dissertation adressiert drei Aufgabenbereiche der Gesichtsanalyse: Gesichtslandmarkenerkennung, Kopfposeschätzung, und Mimikschätzung. Die hier vorgeschlagenen Methoden sind bildbasiert und vollautomatisch - sie starten mit der Gesichtserkennung im zu analysierenden Bild.

Neuronale Netze in einer Kaskade werden genutzt, um 49 Gesichtslandmarken innerhalb eines erkannten Gesichts zu lokalisieren. Die Lokalisation findet in über 5 Kaskaden statt; vier Verbesserungskaskaden folgen einer geführten Initialisierung in der ersten Kaskade. Eine Merkmalsselektion ist durchgeführt in jeder Kaskade, um die Generalisierungsfähigkeit zu verbessern. Dieses Framework lokalisiert die Landmarken mit einem mittleren Fehler für jeden Punkt zwischen $0,72\%$ und $1,57\%$ der Gesichtsbreite im Bild. Weitere Evaluierungen und Vergleiche beweisen die Wettbewerbsfähigkeit des vorgeschlagenen Verfahrens bezüglich Genauigkeit und Effizienz neben ihrer besseren Generalisierbarkeit.

Zusätzlich schlage ich ein Framework zur Kopfposeschätzung in RGBD-Bildern

vor; es ist auch zur Nutzung von RGB-Bildern ohne Tiefeninformationen konfigurierbar. Die Poseschätzung wird durch die Nutzung von Tiefeninformationen verbessert durch zusätzliche tiefenwertbasierte Merkmale oder durch tiefenwertbasierte Gesichtserkennung. Mit letzterer Methode erreichte ich eine akkurate Poseschätzung mit einer durchschnittlichen Fehlerrate von $4,19°$, $3,84°$ und $4,13°$ für die Nick-Gier-Roll-Winkel durch Kreuzvalidierung auf einem öffentlich zugänglichen Datensatz. Dieses Verfahren generalisiert fast perfekt auf einen anderen öffentlichen Datensatz, wobei dort mittlere Fehlerraten von $4,23°$, $4,64°$ und $4,33°$ für Nick-Gier-Roll-Winkel durch Kreuz-Datenbank-Evaluation erreicht wurden. Diese Ergebnisse sind genauer im Vergleich zu Ergebnissen aus Stand-der-Technik-Verfahren.

Außerdem schlage ich drei Verfahren zur bildbasierten mimischen Expressionsanalyse vor: ein geometrisches, ein holistisches und ein Hybrid aus beiden Verfahren. Dabei verwende ich die 49 zuvor lokalisierten Gesichtslandmarken, um die mimische Expression in personenabhängigen und personenunabhängigen Szenarien zu erkennen. Die Verschiebung dieser Punkte zu ihrer Position im personenspezifischen oder neutralen Modell geben wichtige Hinweise für die Expressionsanalyse; die Verschiebung ist stets evaluiert im Bezug zur Gesichtskonfiguration (gemessen an nicht-bewegbaren Gesichtslandmarken). Personalisierte Merkmale führen immer zu einer Verbesserung in der Erkennung von mindestens 3%. Auch die Erkennung der Expression mittels geometrie-basierter Verfahren mit nur 8 Gesichtslandmarken ist vorgeschlagen. Damit habe ich eine mittlere Erkennungsrate von $87,48\%$ erreicht, nur $2,24\%$ weniger als mit 49 Punkten. Drei holistische Merkmalstypen und 4 verschiedene Klassifikatoren wurden untersucht im holistischen Framework zur mimischen Expressionserkennung. Mit Histogrammen orientierter Gradienten (Histograms of oriented Gradients - HOG) und einer Support Vektor Maschine (SVM) erreichte ich die besten mittleren Erkennungsraten: $87,26\%$ für den personenunabhängigen 6-Klassen Fall und $83,71\%$ für den personenunabhängigen 7-Klassen Fall. Die Erkennungsrate des letzteren Falls wurde zu $89,14\%$ verbessert durch eine gemeinsame mimische Expressionserkennung und Gesichtslandmarkenlokalisierung. Dieses Framework untersucht geometrische-

und holistische Methoden zur Gesichtslandmarkenlokalisierung. Auch die Genauigkeit der Lokalisierung der Gesichtslandmarken wurde verbessert im Vergleich zu den isolierten Methoden. Die vorgeschlagene Methode übertrifft Standder-Technik-Verfahren, die ein ähnliches Evaluationsprotokoll verwenden. Empirisch bewiesen generalisieren die geometrie-basierten Methoden besser für die verwendeten Datensätze als die holistischen Verfahren.

Table of Contents

List of Figures

List of Tables

List of Algorithms

CHAPTER 1

Introduction

The person's affective state affects his face, voice, gait, behavior of communications, body expression, heart rate, skin conductivity, blood pressure, etc.. Therefore to infer the human emotional state, a multi-modal approach, performing a fusion of all previously mentioned signals, is required. This task is still a big challenging either in the way of estimating the different signals or in the way of combining them. Nowadays, the research focuses on developing affect aware machines, precisely named affective Computing. These machines would not only recognize the human emotional state but also express it and response to it. As the emotion affects directly the person intelligence in terms of perception, rational thinking, planning, and decision making, we can confidently say recognizing the emotion is not an extra functionality or a supplementary tool that could be added to any system.

In the recent years, several approaches have been proposed to estimate the human affective state from single or combined modalities [132]. For example, Soleymani et al. [150] propose an approach to continuously detect the human emotion via electroencephalogram (EEG) signals and facial expressions. Jenke et al. [77] investigated deeper in perceiving the emotion from only the EEG signals. Han et al. [65] propose a framework to recognize the emotional arousal, indicator of emotion intensity, via audio-visual features extracted from video content and human brain's functional activity measured using functional magnetic resonance imaging

(fMRI). Hammal et al. [64] state that head movements can enhance our understanding of emotion communication. To recognize the emotional states, Nardelli et al. [120] employ a nonlinear analysis of Heart rate variability (HRV) that is derived from the electrocardiogram (ECG). Interestingly, Griffin et al. [57] studied the perception of Laughter from whole-body motion, where they propose an automatic approach for the recognition in continuous and categorical perspectives. Overcoming the traditional contact-based sensors for the stress detection, Chen et al. [29] use a hyperspectral imaging technique to detect the psychological stress. Gruebler and Suzuki [59] designed a wearable device to recognize the positive expressions by analyzing the facial electromyography(EMG) signals, which are read by placing electrodes directly over the facial muscles. Wen et al. [164] state that the correlations of physiological signals such as Fingertip blood oxygen saturation (OXY), galvanic skin response (GSR), and heart rate (HR) are reliable cues as well to recognize human emotions like amusement, anger, grief, and fear. A survey of automatic recognition and generation of the affective expression through body movements was provided by Karg et al. [83]. Yang et al. [173] conducted clinical experiments to investigate the relation between vocal prosody and change in depression severity over time. They found that vocal prosody is a powerful measure of change in the depression severity and therefore, could be used to asses the therapy sessions. It is proven by Giakoumis et al. [55] that the biosignals: GSR and ECG carry useful cues about the boredom. For the recognition, they built an automatic approach that utilizes moment-based features extracted from the biosignals. Jarlier et al. [76] show a great capability of the thermal images to be used in the facial analysis towards robust emotion recognition. By analyzing the movements of the head and hands, Lu et al. [102] are able to detect the human deception. Werner et al. [166] propose an approach to automatically detect the human pain. Their approach is useful for the cases where the patient cannot utter as the algorithm decision is built upon analyzing the head pose and the facial expression. An entire project [3] is dedicated to build a companion system that is capable of adapting itself to an individual based upon an estimation of his current emotion state.

1.1 Facial Analysis

As clearly mentioned, the face is one of the main information sources utilized to infer the human emotional state, not only from the facial expression but also from

the head movements. Earlier the facial muscle contractions were detected through contact-based sensors mounted at the target muscle. Additionally, the facial analysis process was not fully automatic, as human intervention is required to locate the face or to initialize the facial point locations. Due to the importance of this modality, much attention has been paid to this topic. Nowadays, using a non-intrusive sensor as a camera, all the facial processing can be performed automatically, starting from face detection, through localization of facial landmarks and ending with inference of subjects metal state. This makes the facial data are more favorable for the emotion recognition than the bio-signals. Consequently, more applications deploying the knowledge of human mental state have been developed. Those applications cover a wide range of disciplines from entertainment to complex medical systems. In the Human-Human Interaction (HHI), judging the behavior based on physiological signals is inconvenient when intrusive sensors are required to read those signals. The verbal interactive signals (transcript, voice tone) are potential channels providing indications about the human behavior. As there are different words that could be used to express the same thing, relying on these channels looks difficult [131, 7]. Besides that the authors in [7] state that predicting the human behavior from nonlinguistic messages is more reliable. They categorize the visual channel (facial expression and body gesture) as the most important modalities used in human judgment of other behaviors. According to the reported studies in [7, 131], inferring the human behavior from facial expression and body gesture (54%) is more accurate than using only the facial expression (40%), which is slightly better than using only speech (36%) and much more accurate than using only transcripts (29%), body (28%), or tone of the voice (26%).

Duchenne de Boulogne, the French neurologist, was the first scientist who investigated the effect of specific emotions on the face muscles [43]. He believed that the face is like a board where each human inner state is effecting the face in a specific way. He stimulated the facial muscles using electrical probes before capturing the resulting expression, where the photography had been employed for the first time for this purpose, as shown in Figure 1.1. Darwin [41] had taken this research step forward. He asked his friends to asses several photos depicting facial expressions, taken from Duchenne experiments, which opens the door to use photographs in inferring the facial expression. Darwin found that laughter

Figure 1.1: Samples of Duchenne experiment. The facial muscles were stimulated by electrical probes to generate specific facial expressions [43].

is primarily the expression of happiness, which could be clearly seen while children play or people meet old friends. As a sign of laughter, the mouth is opened, where its corners move backwards and a little upwards as well. Simultaneously, the upper lip is, to some extent, raised. According to Duchenne study [43], the great zygomatic muscles are responsible for the mouth movements (draw the corners backwards and forwards). Darwin considered the upper and lower orbicular muscles of the eyes besides the muscles running to the upper lip are at the same time more or less contracted, which affects the laugh intensity as well; the facial muscles are shown in Figure 1.2. Validating the theory of relating the expression to the facial muscles, Darwin showed two photographs, one depicts a natural smile and the other unnatural caused by activating the great zygomatic muscles (see Figure 1.3), to twenty-four persons. The natural one was recognized by all, while only three persons did not perceive the smile expression from the unnatural one, which can be attributed to the missing contraction of the orbicular muscles.

Later on, Ekman and Friesen had taken a pioneer step in the facial analysis field by standardizing it through the development of a Facial Action Code System (FACS) [47]. They broke the facial expression down into smaller action units (AU). Each AU codes a small visible change in facial muscles, as shown in Table 1.1.

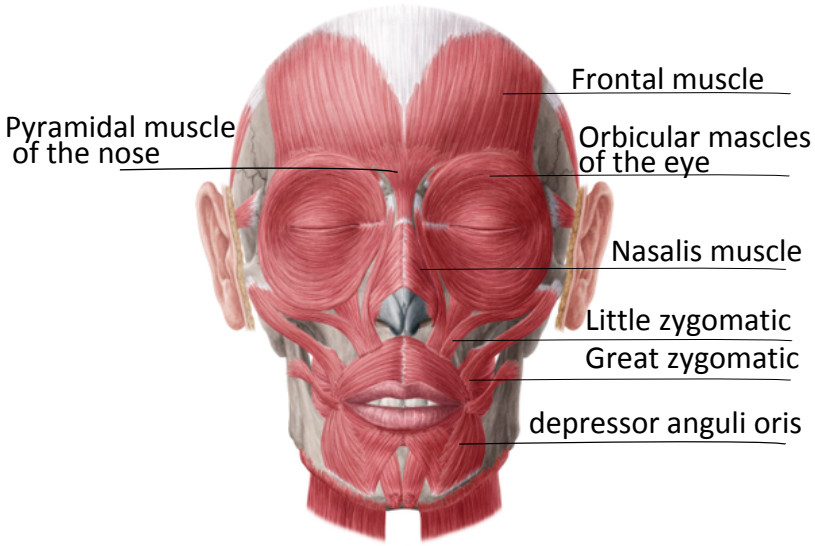


Figure 1.2: The facial muscles. Source [1]

Table 1.1: Action units (AU) in the Facial Action Coding System.

AU no.	FACS Description	Muscular Basis
1	Inner Brow Raiser	Frontalis, Pars Medialis
2	Outer Brow Raiser	Frontalis, Pars Lateralis
4	Brow Lowerer	Depressor Glabellae; Depressor Supercilli; Corrugator
5	Upper Lid Raiser	Levator Palebrae Superioris
6	Cheek Raiser	Orbicularis Oculi, Pars Orbitalis
7	Lid Tightener	Orbicularis Oculi, Pars Palebralis
8	Lips Toward Each Other	Orbicularis Oris
9	Nose Wrinkler	Levator Labii Superioris, Alaeque Nasi
10	Upper Lip Raiser	Levator Labii Superioris, Caput Infraorbitalis
11	Nasolabial Furrow Deepener	Zygomatic Minor
12	Lip Corner Puller	Zygomatic Major
13	Cheek puffer	Caninus
14	Dimpler	Buccinnator

Continued on next page

Table 1.1 – continued from previous page

AU no.	FACS Description	Muscular Basis
15	Lip Corner Depressor	Triangularis
16	Lower Lip Depressor	Depressor Labii
17	Chin Raiser	Mentalis
18	Lip Puckerer	Incisivii Labii Superioris; Incisivii Labii Inferioris
20	Lip Stretcher	Risorius
22	Lip Funneler	Orbicularis Oris
23	Lip Tightner	Orbicularis Oris
24	Lip Pressor	Orbicularis Oris
25	Lips Part	Depressor Labii, or Relaxation of Mentalis or Orbicularis Oris
26	Jaw Drop	Masetter; Temporal and Internal Pterygoid
27	Mouth Stretch	Ptergoids; Digastric
28	Lip suck	Orbicularis Oris
38	Nostril Dilator	Nasalis, Pars Alaris
39	Nostril Compressor	Nasalis, Pars Transversa and Depressor Septi Nasi
41	Lid Droop	Relaxation of Levator Palpebrae Superioris
42	Slit	Orbicularis Oculi
43	Eyes Closed	Relaxation of Levator Palpebrae Superioris
44	Squint	Orbicularis Oculi, Pars Palpebralis
45	Blink	Relaxation of Levator Palpebrae and Contraction o Orbicularis oculi, Pars Palpebralis
46	Wink	Orbicularis Oculi

Consequently, each facial expression is composed of several AUs simultaneously occurring with different intensities.

The perception of the facial expression can take two forms. The first type is messege

(a) (b) (c)

Figure 1.3: Natural smile vs. unnatural smile, three photographs used by Darwin [41] to validate the relation between muscles and facial expressions. (a) neutral state (b) natural smile (c) unnatural smile caused by the galvanization of the great zygomatic muscles.

judgment like, where I assumed the context is priorly known and the facial expression presents the human emotion [48]. The second type is sign judgment, where the context is unknown and the emotion inference would be fused along with several other modalities [32]. The categorical judgment of the facial expression was preferred at a point where it yields considerably higher agreement across the observers. The most common categories of the facial expression are happiness, sadness, anger, surprise, disgust, and fear. These expressions are described as the basic expressions as they are cross-cultural recognizable. To this end, Ekman and Frisen [50] dedicated an experiment, where they showed expressive photographs to observers from five different cultures (Japan, Brazil, Chile, Argentina, and U.S.). They were asked to choose one emotion category out of six; the results affirm the cross-cultural property of the six expressions. Similar results were obtained by repeating the experiment in two-preliterate cultures (Borneo and New Gunea).

Describing the emotion in a categorical-based way confines the wide nature of the emotion. Therefore, a main objective of the research community was to find a suitable continues dimension to describe the emotions. Russell and Mehrabian [139] stated that three dimensions (pleasure-displeasure, arousal-nonarousal, and dominance-submissiveness) are necessary and sufficient to describe a large variety of the human emotional states. In particular, they describe 151 emotional states using the three-dimensional space, where each state is characterized by its mean and standard deviation with respect to each axis. Circumplex model of affect, proposed

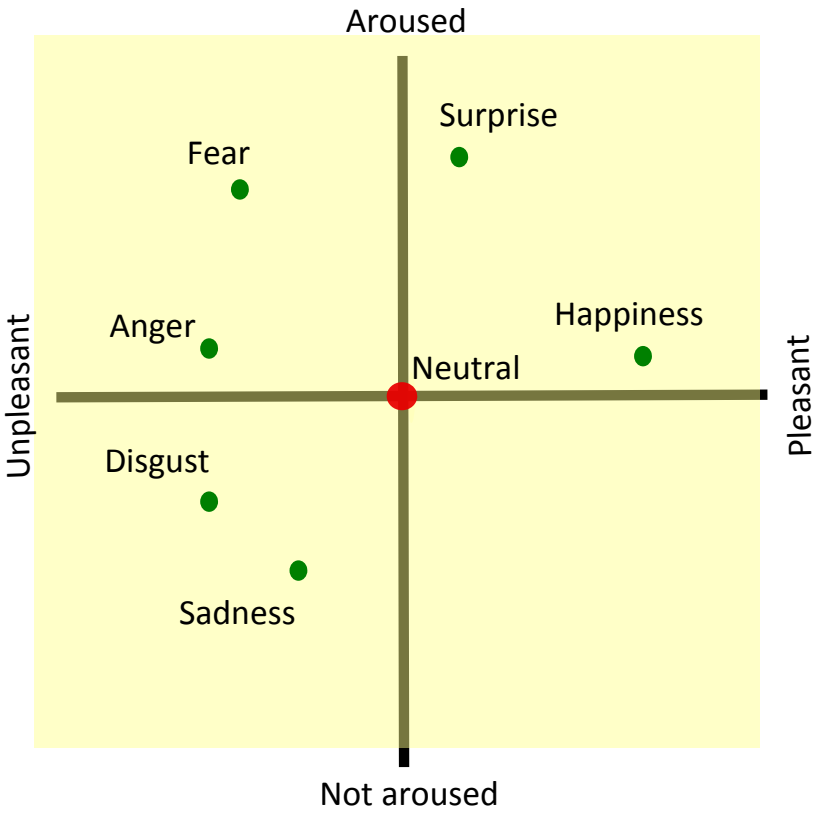


Figure 1.4: The basic facial expressions mapped on the Circumplex model of affect

by Russel [137], was built to describe the emotional states in only two-dimensional space (Arousal-Valence). Interestingly, it is found that the basic facial expressions are located on a circle in this model [138, 22], as shown in Figure 1.4. Niese et al. [123] employ this model to recognize the facial expression and measure its intensity as well.

1.1.1 Discussion

As each person is able to suppress or fake his facial expression, many debates were raised around the study of the facial expressions. Do the facial expressions carry a truth sign about the emotion? Is it enough to judge person emotion based only one modality? Are the acted expressions similar to spontaneous expressions? Could the exaggerated expressions exist in the real life or only a lower intensity of it? Some facial expressions may carry even contradictory information, e.g. laugh (or smile) can be a sign for either delight or frustration [70]. Compound facial expressions of an emotion are also discussed in [45], where a compound of two emotions can be recognized and distinguished from them separately. All these inquires are

out of the scope of this thesis. Here, I consider building an automatic approach to recognize the facial expression. This approach is trained using expressive facial images. Hence, throughout this work, facial expression recognition appears interchangeable with the term emotion recognition. Additionally, two pre-processing approaches for the facial analyses are developed. Taken into consideration that the proposed approaches are beneficiaries for the study of the aforementioned inquires.

1.2 Problem Statement

To successfully infer the human facial signs, I have to propose a comprehensive solution to handle the underlying challenges. The typical processing chain for an appearance-based facial analysis approach starts with image acquisition, then preprocessing, feature extraction, post processing, and ends with feature classification. The resolution of the acquired image affects the performance of the remaining stages; however, the proposed approach supposes to work with reasonable image resolution. Additionally, the face should be correctly and consistently located within the processed image. Due to the face scanning parameters, e.g. scale step and spatial search step, the face detectors are exposed to produce non-consistent cropping for the detected faces especially when the images depict faces of different scale. Such behavior would ruin any further training or testing on the top of the located face patches. Therefore, it is necessary to perform a post processing stage to minimize the variation in the perspective of the returned windows by the face detector. The geometric-based methods represent a second way for the facial analysis. These methods rely mainly on the position of facial landmarks. This highlights the importance of building a robust approach for the facial point detection, which should be invariant to illumination conditions, reasonable viewing angle, human ethnic groups, skin tones, facial expressions, and many other factors. With respect to the head pose, the facial analysis is usually performed in a sequential way, in which several models doing the same facial analysis task, but each for a discrete group of poses. To this end, building a robust head pose estimator is necessary. Such estimator should be invariant to illumination conditions, human ethics, skin tones, and face scales as well.

1.3 Motivation and Application

Facial analysis has been an active research topic for more than two decades due to its increasing importance in various disciplines ranging from entertainment (video games) to medical applications and affective computing. In this work, I propose an approach to automatically recognize the facial expressions. Additionally, I develop methods to perform two main pre-processing tasks: head pose estimation, and facial point detection. Those methods could benefit many computer vision systems besides the facial expression recognition. In what follows, I provide a brief overview of potential applications that benefit from further improvements in the field of facial point detection, head pose estimation, and facial expression recognition.

1.3.1 Facial Point Detection

Facial point detection is a crucial pre-processing in many computer vision systems that involve facial analysis. Consequently, developing an accurate, automatic, robust approach for it has been paid more research attention in recent years. A variety of facial signals can be perceived from the points' relative location, movement, or surrounding texture. Explicitly, the facial basic expressions are recognized through the facial points' relative location [178, 141, 54], movement [153, 144], surrounding texture [178, 93]. Additionally, the facial points are used for the face registration task that precedes the facial analysis [98, 109]. Baltrusaitis et al. [10] employ 22 facial points to detect 12 AUs, in his way to infer the human mental state. Human pain intensity is estimated by monitoring the texture alterations of face regions defined by eight facial points [166]. Lip reading is built directly on top of a facial point detector [142, 156], precisely the mouth points. Monitoring the eyes' facial points over a sequence of images is helpful to infer information about driver fatigue [78]. Simply, the facial point detector is a powerful tool for the facial analysis.

1.3.2 Head Pose Estimation

Head pose estimation is a crucial pre-processing step for several computer vision systems, e.g. it is important to qualify systems of face and facial expression recognition to be pose invariant and accordingly more robust. One the other hand, it is

the core task for many other computer vision systems, e.g. head gesture recognition, gaze recognition, driver monitoring, etc.. One challenge for the facial analysis systems is to cope with uncooperative persons, whose faces are in arbitrary in-depth rotations. This variation caused by pose is larger than variation between persons, thus impairing further facial analysis tasks, e.g. the face recognition and the facial expression recognition as well. Zhang and Gao [180] conducted a survey of approaches recognize the face across poses in which the pose is estimated as a pre-processing step in many of them. Niese et al. [122] estimate the head pose before employing it to project extracted features, distances between the facial points and optical flows, onto a frontal face and then perform a pose-invariant facial expression recognition. Considering different poses, Moore and Bowden [114] exploit a texture-based approach to perform a multi-view facial expression recognition. They dedicate a separate classifier for each pre-estimated pose. By learning the mapping between facial points in each pair of discrete non-frontal pose and its corresponding frontal pose, Rudovic et al. [135] propose a Coupled Scaled Gaussian Process Regression (CSGPR) model for head pose normalization in his way to develop a pose-invariant approach for the facial expression recognition. In a similar way, a robust estimation of the head pose leads to pose-invariant face recognition [180]. For head gesture recognition, a continuous estimation of the head pose over an image sequence is required. Morency and Darrell [115] use the nod of a person's head as user interface commands, precisely for dialog box confirmation and document browsing. Head gestures are also considered as a language in Human-Robot conversations, in which human can instruct the robot or pass it feedbacks [148]. The human mental state can be inferred through the fusion of several modalities, one of them is the head gesture [81, 10]. The head pose is a valuable cue for inferring the gaze direction [26]; for this purpose, a new database is devoted [111]. Additionally, head pose carries rich information about the visual focus of attention, which is employed in different applications such as human behavior analysis in multi-person scenarios [9], and driver assistance systems [119, 80].

1.3.3 Facial Expression Recognition

The importance of knowing a human mental state appears in different disciplines. For example, Human Computer Interaction (HCI) is required to be improved to be as good as Human Human Interaction (HHI). Hence, recognizing the human emotions by machines is considered an important step forward. For developing a companion-based assistant system, facial expression is considered as a complementary aspect to hand gestures and other modalities [165]. Human pain and its intensity can now be inferred from the facial expression [166]. Discovering an existence of deception is a common facial analysis as well [7]. Feedbacks for different services can be automatically taken through reading the human facial expression. As a case in point, the one-to-one tutoring outperforms conventional group methods of instruction. Consequently, adapting one-to-one tutoring to student performance through a cognitive process (non verbal behavior recognition) is crucial [97]. The feedback via facial expression is also exploited in the design of games [88, 66, 23]. The methods developed here are applicable for building a face recognition system as well, which is employed in several surveillance and security applications [72, 33, 163]. An intelligent vehicle system can detect drowsy drivers via analyzing the facial appearance [160].

1.4 Goals and Contributions of Thesis

The main objective of this work is to advance the frame-based facial analysis research by developing robust approaches for facial point localization, head pose estimation, and facial expression recognition. Those approaches shall work automatically starting from locating the face in the processed image. Additionally, they are required to be effective, efficient, invariant to resolution, skin tone, age, and some other factors. The contributions of this dissertation are summarized as follows.

- Considering the **facial point localization**, I propose an approach to locate 49 facial points via neural networks in a cascade-regression framework. This approach is superior to state-of-the-art approaches and two commercial software packages. Moreover, it is one of the most efficient methods for point localization.

- Considering the **human head pose estimation**, I propose a framework to

estimate the head pose based on RGBD images. The framework starts by performing a cropping refinement task on the detected face patches. To this end, three methods were proposed; two are RGBD based, while the last one was RGB based. The last method qualifies our approach to be applicable without depth information on the conventional 2D cameras. I adapt three appearance-based feature types to encode the varying facial appearance across poses, where a fair comparison among them in terms of accuracy and computation times is provided. Additionally, I introduce depth-based features; by employing them I achieve a competitive accuracy at lower computation time. Several conducted evaluations state that the proposed approach provides a competitive estimation accuracy and has a better generalization capability in comparison to the state-of-the-art methods.

- Considering the **facial expression recognition**, I propose a geometry- and an appearance- based methods to infer a facial expression from a single face patch. For the geometry-based approach, I employ the 49 facial points, detected by method developed in this dissertation. Additionally, I introduce a geometry-based method relying only on 8 facial points, where the drop in the average recognition rate does not exceed 3%. For the appearance-based method, I investigate across appearance-based feature descriptors and classifiers to arrive at the optimal method. The configurations of all methods were empirically set. Finally, I propose a framework to joint facial expression recognition and facial point localization. This framework tackles the two tasks on a frame basis as well. It makes use of both geometry and appearance- based methods for the expression recognition, and of both cascade-regression and local-based methods for the point localization. With this framework, both recognition rate and localization accuracy are boosted.

1.5 Overview of the Manuscript

This manuscript is organized in eight major chapters, including this introductory chapter. The contents of the remaining seven chapters are summarized as follows. **Chapter 2** provides a brief survey of the most related research regarding the three addressed tasks: facial point localization, head pose estimation, and facial expression.

Chapter 3 describes the necessary fundamentals of the employed machine learning approaches. Additionally, the exploited appearance-based features are explained. Throughout this dissertation, I employ the Viola and Jones (VJ) face detector in different approaches; a description of this detector, highlighting its advantages and disadvantages, is also given in this chapter.
Chapter 4 details datasets that are exploited throughout this dissertation. The datasets are grouped under three categories according to their intended use here, not their specifications.
Chapter 5 concerns the detail description of the proposed approach for the facial point localization. The corresponding experiments and comparisons are also presented in this chapter.
Chapter 6 concerns the detail description of the proposed approach for the human head pose estimation. In this chapter, I introduce depth-based descriptors to encode the face geometry, which are employed later for the pose estimation. Three proposed methods to guarantee a consistent face cropping are described within this chapter as well. The corresponding experiments and comparisons are also presented in this chapter.
Chapter 7 concerns the detail description of the proposed approach for facial expression recognition. Two geometry-based approaches, one of 49 facial points and another of just 8 facial points, are explained. An appearance-base approach is discussed and evaluated for different feature descriptors and classifiers. Additionally in this chapter, a proposed framework for joint facial expression and facial point localization is explained in detail. Various experiments, conducted to assess the performance of the proposed methods, are presented.
Chapter 8 concludes the contributions, investigations, and experiments that have been discussed in the thesis. Additionally in this chapter, I briefly suggest some possible directions for further research regarding the same area, either as an extension, improvement, or employing.
Appendix A details the results stemmed from evaluating the proposed methods for facial expression recognition on the BU-4DFE database.
Appendix B details the results stemmed from cross-database evaluations of the proposed geometry- and appearance- based approaches for facial expression recognition.

CHAPTER 2

State-of-the-Art

Throughout the last three decades, several approaches have been proposed to tackle the three facial analysis tasks, which are addressed in this dissertation as well. Most of them differ from our methods as they relay on: human intervention (not fully automatic), temporal information (not frame-based), or different data sources (e.g. thermal imaging). In what follows, I give a brief overview of recent studies concerning the three tasks.

2.1 Facial Point Localization

Facial point localization is the backbone for many facial-based applications. In the literature, several approaches have been developed to accomplish this localization; however, there is still a space for improvement in the regards of efficiency; accuracy; invariance to head pose, illumination, and expression. The general outline of any point detector is as follows. First, the face is located within the processed image. Second, potential locations of the facial points are evaluated. Third, a plausible facial configuration is investigated. In the first step, the face detector should guarantee a consistent face cropping, or registration process follows, as many approaches relay on the face detector output to set the size of local patches accordingly [14]. In a similar way based upon the scale of the detector output: Martinez et al. [108] locate 20 facial points in near-frontal face view, Baltrusaitis et al. [12] locate 68 facial points, Tzimiropoulos and Pantic [155] locate 68 facial points as

well. The detected face may be further registered across a small range of scales (around the output of face detector) to stabilize the detector output or the point localization may be performed across this small range of poses [152]. For the second step, different methodologies were proposed. Some approaches perform the time-consuming grid-search [14]. In [12], they measure the local response of each point using linear Support Vector Machine (SVM). More efficient methods were proposed as well: Martinez et al. [108] develop a guided search method, Smith et al. [149] evaluate only points that match some exemplars. In the third step, to guarantee an admissible face configuration from those locally evaluated points, many approaches exploit the prior global shape information through Point Distribution Model (PDM) [113], Constrained Local Models (CLM) [36, 12], graph models [157], pictorial structures [51], consensus of exemplars [14], Active Shape Model (ASM) [92], aggregated evidences with probabilistic model [108]. Tzimiropoulos and Pantic [155] jointly optimize the part-based model along with the global shape model via Gauss-Newton optimization, which results in a joint update at each iteration of the model parts location and appearance.

Instead of evaluating all potential locations for each facial point and then refining the selection through a global shape model, cascade regression methods, which are significantly more efficient, can be exploited. An initial shape (usually the mean shape of the training set) is set within the box enclosing the detected face. A regression-based method then maps texture features extracted from patches around the initial points' location to their displacement from the ground truth position. Next, succeeding iterations (neural networks) map features extracted with respect to the new points' location, which was estimated from the previous iteration, to their displacement from the ground truth. Acceptable results are usually obtained after three to seven iterations. The cascade regression does not employ an explicit general shape model; however, the shape constraints implicitly hold as the point displacement lies on the manifold of the training set. Xiong and De la Torre [170] provide a supervised decent method (SDM) to learn the feature to point location mapping. Yan et al. [171] perform the cascade regression across the face scales and shifts to overcome the inconsistent face cropping issue, at the cost of more processing time. The facial point localization can be optimized simultaneously with other tasks, e.g. Zhu and Ramman [185] introduce a unified model for face detection, pose estimation, and landmark localization (68 points in frontal cases) in single images. Their model is based on a mixture of trees with a shared

pool of parts, where every facial landmark is modeled as a part; global mixtures are used to capture topological changes due to the viewpoint. They search across scales to score local part before evaluating several configurations of parts across views as well.
In this dissertation, I automatically locate 49 facial points via neural networks in a cascade regression fashion. Moreover, I propose a framework for joint facial expression recognition and point localization, which is configured for the case of seven expressions and eight facial points.

2.2 Head Pose Estimation

A head pose estimator is beneficial for many computer vision systems, either as a pre-processing step qualifying them to be pose-invariant, or as a core task when head gestures are recognized. Consequently, in the literature one can find several approaches proposed to tackle the pose estimation. In what follows, I categories these approaches according to the following criteria: temporal dependency, estimation continuity, and data source.

2.2.1 Temporal Dependency

With respect to the temporal dependency criterion, I divide the developed approaches into frame- and video- based categories. Frame-based approaches refer to those which consider only the current frame data for their instant decision, in other words, without employing any temporal information as in the video-based methods. Exploiting the face appearance in single frames, the head pose is inferred using texture-based descriptors: HoG by Murphy-Chutorian et al. [119] and by Zhu and Ramanan [185], LGBP by Ma et al. [105], Gabor filters by Wu et al. [167]. Gurbuz et al. [61] had developed a model free method for the head pose estimation using stereo-vision. They employ the reconstructed face plane along with the eye location to estimate each instance pose. In contrast to frame-based methods, several approaches exploit the temporal information either to enhance the pose estimation accuracy or to estimate a wider range of head poses. Requiring no training phase, Jimenez et al. [79] employ a stereo camera to infer the current human head pose. In their proposed approach, a 3D face model is created from 2D points superposed over the frontal face image using a stereo correspondence. Then, RANSAC and POSIT algorithms are used to track the 2D points and

accordingly deduce the human pose at each frame, assuming the tracking starts from a frontal pose of zero rotation angles. Otherwise, the pose angles of the first frame will appear as a constant offset error. Tu et al. [154] approach tracks the head pose in low resolution videos with the help of a particle filtering framework, where the appearance variations are online modeled by an incremental weighted PCA subspace with a forgetting mechanism. Similar to many other approaches, they assume the face tracking begins from a face of zero rotation angles. Employing multiple cameras to enhance the head pose tracking is an option in several approaches. Ruddarraju et al. [134] extend an eye-tracking method from a single camera system to a multiple camera system, where the head pose is estimated by triangulating multiple facial features obtained from the eye tracker.

2.2.2 Data Source

The majority of the aforementioned approaches estimate the head pose in gray or RGB 2D images. Other approaches enhance their face tracker via 3D information stemmed either from a stereo/multi-camera or cameras accompanied by a depth sensor, namely Depth-sensing (RGBD) cameras. In contrast to the color image texture, the depth data are less sensitive to the illumination variations. Consequently, nowadays several approaches exploit the offered depth information by those sensors to boost their pose estimation performance. Based only on the depth data, Niese et al. [125] create a person-specific head model that consists of 3D point vertices and surface normals. Then, they use the Iterative Closest Point (ICP) algorithm to fit this head model to a current head pose of the considered person, assuming that the face is located in the upper part of the point cloud, and the pose tracking starts from smaller angle values. Fanelli et al. [52] estimate the head pose from only the depth data as well but on a frame basis. To this end, they use discriminative random regression forests, in which each node splitting is supposed to reduce the entropy of the class label distribution and the variance of the head position and orientation. The employed random forests are supposed to detect the face patch as well. Exploiting both sources (D+RGB) from a Kinect sensor, Yang et al. [172] use three steps to arrive at an estimate of the head pose. First, they detect a coarse location of the face. Then, based on the coarse detection, they perform a refining search on the image coordinates and scales to find the accurate head location. Finally, they estimate the head pose with the help of a feed-forward

Multi-Layer Perceptron (MLP) network. Mukherjee and Robertson [117] employ both sources (D+RGB) to infer the head pose via deep neural network. A thermal image is a functional source as well, Buddharaju et al. [21] propose an approach to estimate the pose of a head depicted in it.

2.2.3 Estimation Continuity (Pose Domain)

With respect to the domain of the pose estimates, I divide the state-of-the-art approaches into two groups. A group returns discrete pose estimates, while the other returns continuous estimates. The first category assigns the detected face to one of many discrete poses, usually the pose ranges are quantized by 15°. Approaches belonging to this category are not qualified for head gesture recognition as their resolution constrained by a fixed quantization error. Classification-based methods are mainly employed to classify each head image into one discrete pose, as used by Ma et al. [105], Zhu and Ramanan [185], and Dahmane et al. [38]. On the other side, regression-based methods are mainly employed to provide a continuous estimate of the head pose, as used by Murphy-Chutorian et al. [119], Yang et al. [172], and Fanelli et al. [52]. The approaches of [79, 125] fit a general/personalized head model to the considered person head data to return a continuous estimate of his head pose.

A summary of the aforementioned approaches, in which each approach is described in terms of the three criteria: temporal-dependency; data source; and the pose estimate continuity, is given in Table 2.1. Accordingly, our proposed method is classified as a frame-based approach that provides continuous head pose estimates based on D+RGB data sources.

2.3 Facial Expression Recognition

Ekman and Friesen [47] broke the facial deformations down into smaller AUs, where each AU codes small visible change in facial muscles. Accordingly, each facial expression is defined to be composed of several AUs simultaneously occurring with different intensities. Consequently, expression recognition can be performed on the basis of an AUs recognizer. Another functional option, one can use directly

Table 2.1: A summary of the state-of-the-art approaches for head pose estimation. Each approach is described in terms of three criteria, its temporal-dependency (**Te-De**), data source (**Da-So**), and the estimate continuity (**Es-Co**).

Approach	**Te-De**	**Da-So**	**Es-Co**
Murphy-Chutorian et al. [119]	Frame-based	RGB	Continuous
Gurbuz et al. [61]	Frame-based	Stereo camera	Continuous
Jimenez et al. [79]	Video-based	Stereo camera	Continuous
Tu et al. [154]	Video-based	RGB	Continuous
Ruddarraju et al. [134]	Video-based	Multiple camera	Continuous
Niese et al. [125]	Video-based	Depth	Continuous
Fanelli et al. [52]	Frame-based	Depth	Continuous
Yang et al. [172]	Frame-based	Depth + RGB	Continuous
Mukherjee and Robertson [117]	Frame-based	Depth + RGB	Continuous
Buddharaju et al. [21]	Frame-based	Thermal Image	Discrete
Ma et al. [105]	Frame-based	RGB	Discrete
Dahmane et al. [38]	Frame-based	RGB	Discrete
Zhu and Ramanan [185]	Frame-based	RGB	Discrete
Our proposed approach	Frame-based	Depth + RGB	Continuous

geometry and appearance features for expression recognition; those features implicitly encode the aforementioned AUs.

With respect to prior knowledge of person-specific neutral state, I sort the expression recognition methods into two groups. Approaches of the first group consider this knowledge essential for their methods. They deduce the facial expression by comparing features from the investigated face image with those of the same face at the neutral expression [104, 122]. These approaches do have limitations such as the human intervention to define the person-specific neutral state. Several methods were proposed to automatically estimate this state, e.g. the average over many frames for each person is assumed to be the person-specific neutral expression; however, this method is an error prone and cannot provide hand-annotation accuracy. As an example of this category, Lucey et al. [104] manually labeled 68 facial points in keyframes within each image sequence, then used a gradient descent Active Appearance Model (AAM) to fit these points in the remaining frames. They infer the human facial expression from the displacement of those points via a multi-class SVM. Another example in 3D, Niese et al. [122] utilize dynamic and

geometric features extracted across video clip from facial points and specific regions associated with the 3-D face model of each subject. These points are initially annotated or detected on the neutral state image and tracked over the remaining sequence. Spatio-temporal features of image sequence are utilized as well for the expression recognition, e.g. Valstar et al. [158] exploit the motion history inside the face image to infer the facial expression via Sparse Network of Winnows (SNoW) and a standard k Nearest Neighbour (kNN) classifier. Zhu et al. [186] use Hidden Markov Model (HMM) along with moment invariants to do facial expression recognition. By modeling the temporal behavior of the facial expressions via dynamic based network, Zhang et al. [181] identify the expression from spatio-temporal information. They employ IR illuminations and Kalman filtering to assist the facial point detection and tracking. Baltrusaitis et al. [10] suggest a dynamic system with three levels of inference on progressively longer time scale to understand the human mental states from facial expressions and upper-body gestures, where they employ both Dynamic Bayesian Network (DBN) and HMM. Lorincz et al. [100] exploit time-series kernels to analyze the spatio-temporal process of the facial points, where the point movements in 3D space are classified with kernels derived from time-warping similarity measures. Many other approaches exploit the dynamics of the facial points within an image sequence to recognize the depicted facial expression, assuming the dynamics start from the neutral state [162, 183, 96, 124, 130]. Texture dynamics are used as well for the expression recognition as could be seen in [86, 182, 168].

By contrast, the approaches of the second category do not relay on prior knowledge of person-specific neutral state. They usually generate a feature vector of larger size, which leads to increase the classifier test and train time. Most these approaches operate on a frame basis. For example, Littlewort et al. [98] convolved registered detected face image with a filter bank of 72 Gabor filters of eight orientations and nine spatial frequencies, where each filter output value is considered as a feature. All those features form the input into an individual SVM classifier for smaller facial action units (AU). Finally, they built a Multivariate Logistic Regression classifier (MLR) on top of the output of the AU classifiers to recognize the human facial expressions. With the similar idea but different texture descriptor, Shan et al. [145] employ the Local Binary Patterns (LBP) for the facial expression recognition. Several modified versions of LBP were also proposed, e.g. Local Normal Binary Patterns (LNBP) [143], Local Phase Quantisers (LPQ) [179], Local Sign

Directional Pattern (LSDP) [25]. Histograms of Oriented Gradients (HoG), texture-based features, were also exploited for the expression recognition [24, 82]. Panning et al. [129] utilize the Haar-like features. A concatenation of geometry- and texture-based features was also proposed by Datta et al. [42].

In this dissertation, I propose methods to recognize the facial expressions automatically on a frame basis. These methods are geometry and appearance based. Additionally, I propose a framework for joint facial expression recognition and point localization.

CHAPTER 3

Fundamentals

The main objective of this chapter is to describe necessary fundamentals of approaches and methods exploited throughout this dissertation. The chapter is divided into three sections. The first section is dedicated to provide a brief overview of four machine learning approaches, which were exploited for the classification and regression purpose within the dissertation. The second section explores three appearance-based features, explaining their computation and presenting examples of their successful employment. In the third section, the Viola and Jones (VJ) face detector is elaborated since it has been employed in most of the developed methods here.

3.1 Machine Learning

Machine learning, as the name suggests, is the process to grant the machine the ability to take a decision based on observation events, as it learned that in earlier stage with similar situations. The study of the machine learning is related to several areas such as pattern recognition, computational learning theory, artificial intelligence. The machine learning here stands for the algorithms that are capable of performing the aforementioned tasks, learning and testing (giving decision). Throughout this work, I employ several machine learning algorithms either to assign an observation to a predefined class in a classification task, or to predict a value based on the observation in a regression task. In particular, I employ

classifier-based algorithms to recognize the facial expressions. On the other hand, regression-based methods are utilized to predict the human-head pose and to locate the facial point within the face. In what follows, I provide a brief overview of machine learning algorithms that are exploited throughout this dissertation.

3.1.1 Artificial Neural Network (ANN)

Initiated by MacCulloch and Pitts [110] perspective of building logical machine in analogy to the biological neural network, the artificial neural network had been receiving a lot of sophisticated contributions. The major step forwards was the commencement of developing self organized learning method [69] and supervised learning [133]. The first model of the perceptron, simplest form of a neural network, was proposed by Rosenblatt [133], as shown in Figure 3.1. The figured perceptron is a linearly combination of input data in addition to an external bias. The resulting sum is applied to a hard limiter to produce an output of ± 1.

$$y = \text{sign}\left(\sum_{\text{input}} w_i x_i + b\right) \tag{3.1}$$

The above model (Eq. (3.1)) represents a single-layer neural network that is only capable of classifying linearly separable patterns. Complicated tasks with non-linearly separable patterns require a multi-layer network, in which hidden layers perform nonlinear transformation on the input data into new space of only salient inputs [68]. In this dissertation, I employ a feed-forward artificial neural network model, known as a multi-layer perceptron (MLP). MLP comprises input layer, output layer, and one or more hidden layers. Each layer contains several units. The

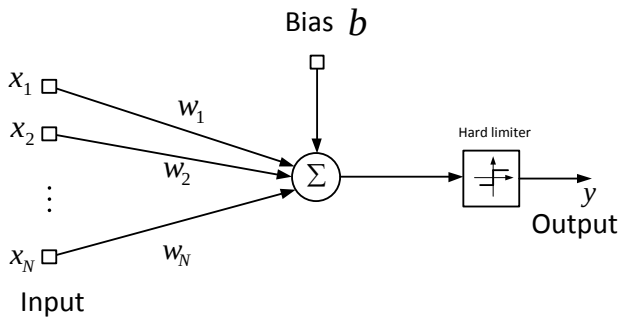


Figure 3.1: The Signal-flow graph of the perceptron.

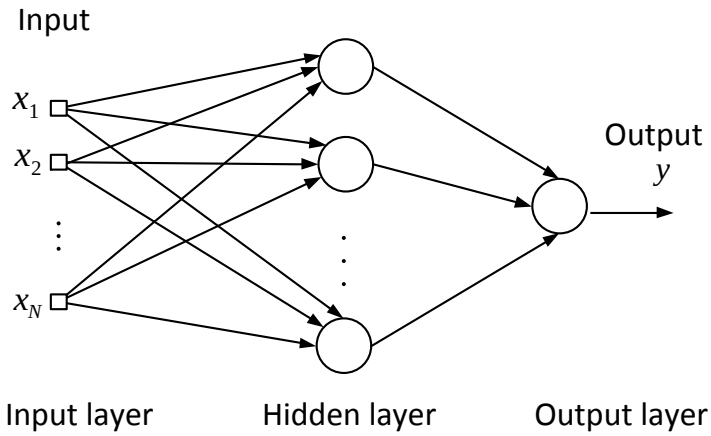


Figure 3.2: Architectural graph of a multilayer perceptron with one hidden layers.

input layer units are equal to the input dimension and the output units to the required output dimension, while the number of the hidden layer units varies with respect to the task. In a fully connected MLP, each perceptron (node) is connected to all nodes of the previous layer, with one way forward connection as shown in Figure 3.2. At each unit, the inputs are weighted and then summed, plus a weighted bias. The resulting sum is then applied to an activation function to produce the unit output. Several functions, preferably differentiable, can be used here such as

- Identity function: $h(x) = x$,
- Symmetrical sigmoid: $h(x) = \beta(1 - e^{-\alpha x})/(1 + e^{\alpha x})$,
- Gaussian function: $h(x) = \beta e^{-\alpha x^2}$.

For MLP with one hidden layer, the input-output relation can be formulated as follows.

$$y = h^{(2)}\left(\sum_{j=1}^{M} w_j^{(2)} h^{(1)}\left(\sum_{m=1}^{N} w_{jm}^{(1)} x_m + w_{j0}^{(1)}\right) + w_0^{(2)}\right). \tag{3.2}$$

The weight values $(w_{\cdot}^{(2)}, w_{\cdot\cdot}^{(1)})$ need to be adjusted with an aim of minimizing resulting error that is defined as follows.

$$E(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{N}(y_i - t_i)^2, \tag{3.3}$$

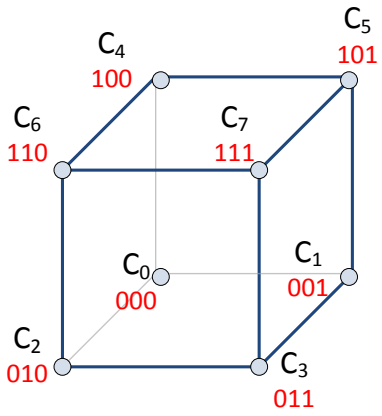
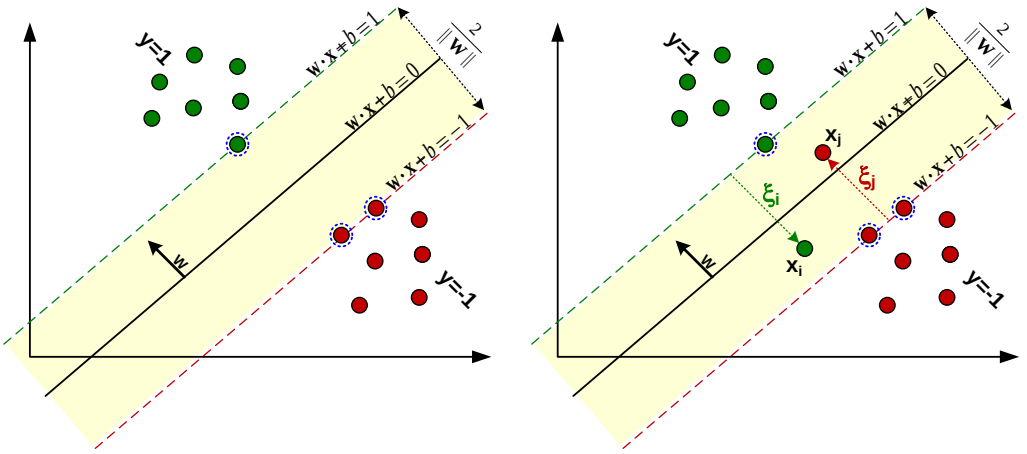


Figure 3.3: The constellation of three MLP outputs employed to encode eight classes.

N denotes the number of the training samples; t is the target value. In this dissertation, I employ the Back Propagation (BP) algorithm [136, 90] to obtain the optimal weight values. With a proper scaling function, a single output MLP can perform either regression or binary classification task. For a multi-class classification, a multi output MLP should be employed, where d outputs are capable to handle up to 2^d classes. Figure 3.3 presents a constellation with three outputs encoding eight classes, where in testing a decision is taken in favor of the nearest class. Another solution to handle the multi-class classification is to build one-vs-all classifiers and decide in favor of the maximum prediction.

3.1.2 Support Vectors Machines (SVMs)

Similarly, SVM was firstly introduced as a binary classifier, but recently several variants of SVMs based on the same principles have been proposed to extend SVM into regression and multi-class problems [4, 28]. SVMs are based on the principle of structural risk minimization (SRM), which works by maximizing the margin between the decision hyperplane and the closest training examples. Consequently, SVMs have better generalization capability in comparison to ANN. In the case of binary linearly separable classification problem, SVM estimates the optimal decision boundary by maximizing the minimal distance from the decision boundary to the labeled data. The identification of the decision boundary split the label space into two sides, hence, any new sample can be easily classified to the side that belongs to. Formally, let $\mathcal{D} = \{\mathbf{x}_i \in \mathbb{R}^d | \; i = 1, \ldots, n\}$ be our training set, and



(a) (b)

Figure 3.4: SVMs classification: (a) A binary SVM with the corresponding optimal hyperplane, support vectors are those on the margin, (b) SVM with soft margin decision boundary.

$y_i \in \{+1, -1\}$ be the class label of $\mathbf{x}_i$, thus two parallel separating hyperplanes can be described as follows:

$$y_i = \begin{cases} +1, & \mathbf{w} \cdot \mathbf{x}_i + b \geq 1 \\ -1, & \mathbf{w} \cdot \mathbf{x}_i + b \leq -1 \end{cases} \tag{3.4}$$

where "$\cdot$" denotes the dot product operator, $\mathbf{w}$ is a perpendicular vector to the two hyperplanes and b is the bias, as shown in Figure 3.4 (a). Therefore, the separating decision boundary (i.e. the optimal hyperplane) that maximizes the margin between the two classes is created by solving the following constrained optimization problem:

$$\text{Minimize} : \frac{1}{2}\|\mathbf{w}\|^2, \quad \text{subject to} \quad y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 \ \forall i \tag{3.5}$$

By Lagrange duality, after some lengthy but straightforward calculations, the dual problem of the primal problem in Eq. (3.5) is given as:

$$\begin{aligned} \text{Maximize} : \quad & \mathcal{W}(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \tfrac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \\ \text{subject to} \quad & \alpha_i \geq 0, \quad \sum_{i=1}^{n} \alpha_i y_i = 0 \quad \forall i. \end{aligned} \tag{3.6}$$

where $\alpha_i \geq 0$ are the lagrangian multipliers. Since Eq. (3.6) describes a quadratic programming (QP) problem, and a global maximum always exists for α_i, $\mathbf{w}$ can be deduced as:

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i \tag{3.7}$$

An interesting characteristic of this solution of the dual problem in Eq. (3.7) is that many values of α_i are zeros. The feature vectors $\mathbf{x}_i$ corresponding to $\alpha_i > 0$ are termed *support vectors* that lay on the hyperplanes, hence the decision boundary can be adequately determined by them alone. Formally, let $t_j (j = 1, \ldots, \mathbb{k})$ be the indices of $\mathbb{k}$ support vectors, then Eq. (3.7) can be rewritten as follows.

$$\mathbf{w} = \sum_{j=1}^{\mathbb{k}} \alpha_{t_j} y_{t_j} \mathbf{x}_{t_j} \tag{3.8}$$

For testing a feature vector $\boldsymbol{z}$ of an unknown letter class, this function is first evaluated: $f(\boldsymbol{z}) = \mathbf{w} \cdot \boldsymbol{z} + b = \sum_{j=1}^{\mathbb{k}} \alpha_{t_j} y_{t_j} (\mathbf{x}_{t_j} \cdot \boldsymbol{z}) + b$. It is then decided that $\boldsymbol{z}$ belongs to the first letter class if $f(\boldsymbol{z}) > 0$ or to the second letter class otherwise. In order to deal with nonlinearly classification problem, the authors in [34] show that this type of challenge can be efficiently approached by allowing some examples to violate the margin constraints (see Figure 3.4 (b)). These potential violations can be formulated using some positive slack variables ξ_i and a penalty parameter $C \geq 0$ that penalizes the margin violations. The slack variables that approximate the number of misclassified examples basically depend on the output of the discriminant functional $\mathbf{w} \cdot \mathbf{x} + b$. Formally, the optimization problem, in this case, can be written as:

$$\begin{aligned} &\text{Minimize :} && \tfrac{1}{2}\|\mathbf{w}\|^2 + C \sum_{i=1}^{n} \xi_i \\ &\text{subject to} && y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0 \quad \forall i. \end{aligned} \tag{3.9}$$

After computations similar to those performed for the linearly separable case, the dual constrained optimization problem is formulated as

$$\begin{aligned} &\text{Maximize :} && \mathcal{W}(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \tfrac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \\ &\text{subject to} && 0 \leq \alpha_i \leq C \quad \forall i, \quad \sum_{i=1}^{n} \alpha_i y_i = 0. \end{aligned} \tag{3.10}$$

The dual optimization problem in (3.10) is very similar to that of the linear separable case, but here there is an upper bound C on the coefficients α_i. Likewise, by using the same formula in (3.8), the weight vector $\mathbf{w}$ can be recovered. The solution algorithm attempts to keep $\boldsymbol{\xi}$ null, while maximizing the margin. It does not minimize the number of misclassifications, but minimizes the sum of distances from the margin hyperplanes. When C increases, the number of error decreases and the number of support vectors drops; further as C tends to ∞, the number of errors tends to 0.

3.1.2.1 Extension to non-linear Decision Boundary

So far, this brief introduction has considered SVMs with a linear decision boundary only. To generalize the SVMs from linear classification to nonlinear classification, the method makes use of a mapping function ϕ that transforms data points $\mathbf{x}_i$ from the input space $\mathbf{X}$ into a high dimensional feature space $\mathbf{F}$. By employing a proper transformation, a nonlinear operation in the input space can be transformed into a linear operation in the feature space. Thus, it makes the classification problem easier and turns the original nonlinearly separable problem into linearly separable. In practice, the feature space has a higher dimensionality than the input space, hence, the computation in the feature space is more costly. To avoid expensive computation in the feature space, the so-called kernel trick is introduced [147], the algorithm is similar to the linear case, except that dot product is replaced with a non-linear kernel function. Recalling the expression for the SVM optimization problem given by Eq. (3.6), the data points only appear as an inner product. Hence, the kernel function is defined such that it calculates the inner product in the feature space, as follows

$$K(\mathbf{x}_i, \mathbf{x}_j) = \boldsymbol{\phi}(\mathbf{x}_i) \cdot \boldsymbol{\phi}(\mathbf{x}_j). \tag{3.11}$$

Now, by substituting every occurrence of the inner product in Eq. (3.6) with the kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$, the dual problem is rewritten as

$$\begin{aligned} &\text{Maximize :} \quad \mathcal{W}(\boldsymbol{\alpha}) = \sum_{i=1}^{n} \alpha_i - \tfrac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \\ &\text{subject to} \qquad 0 \leq \alpha_i \leq C \quad \forall i, \quad \sum_{i=1}^{n} \alpha_i y_i = 0. \end{aligned} \tag{3.12}$$

In practice, there are several commonly used kernel functions, such as

- Polynomial kernel of degree d:

$$K(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \cdot \mathbf{y} + 1)^d,$$

- Radial Basis Function (RBF) kernel (Gaussian kernel) with width σ:

$$K(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|^2 / 2\sigma^2),$$

- Sigmoidal kernel with parameters κ and θ:

$$K(\mathbf{x}, \mathbf{y}) = \tanh(\kappa \mathbf{x} \cdot \mathbf{y} + \theta).$$

The aforementioned methods were optimized to solve two-class problem. Hence, a comprehensive technique is required to perform a multi-class classification task, where each observation is assigned into one of k classes. To this end, several methods were built on top of two-class classifiers, either by utilizing the one-vs-all or one-vs-one configurations [71]. Additionally, the multi-class probability can be estimated based on the underlying two-class predictions [169]. In this work, I exploited the implementation of [28], where the multi-class tasks are solved based on several one-against-one classifiers. Specifically, $k(k-1)/2$ classifiers are constructed, each is trained only using two-class data. Let r_{ij} denote the probability of class label $y = i$ for a given observation $\mathbf{x}$, which is estimated using the pairwise classifier $C_{ij} = C_{ji}$ as described in [95].

$$r_{ij} = P(y = i | C_{ij}, \mathbf{x}), \tag{3.13}$$

The main aim here is to estimate the posterior probability for each class, $p_i = P(y = i|\mathbf{x}),\ i = 1, \ldots, k$. With respect to relations like $p_i/(p_i + p_j) \approx r_{ij}$ and $\sum_{j:j\neq i} r_{ji}p_i \approx \sum_{j:j\neq i} r_{ij}p_j$, estimating the posterior probability $\mathbf{p}$ is formulated as the following optimization problem.

$$\begin{aligned} \min_{\mathbf{p}} \quad & \sum_{i=1}^{k} \sum_{j:j\neq i} \left(r_{ji}p_i - r_{ij}p_j\right)^2 \\ \text{subject to} \quad & \sum_{i=1}^{k} p_i = 1, \quad p_i \geq 0,\ i = 1, \ldots, k. \end{aligned} \tag{3.14}$$

Eq. 3.14 has a unique solution that can be obtained using a simple linear system [169]. Then, the classification rule is defined as follows

$$C_{\mathbf{SVM}}(\mathbf{x}) = \arg\max_i (p_i(\mathbf{x})). \tag{3.15}$$

3.1.2.2 Support Vector Regression (SVR)

In SVR [4], the input-output relation can be given as follows

$$f(\mathbf{x}) = \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}) + b. \tag{3.16}$$

A piecewise linear function is used as an error function such that

$$E_r(y - f(\mathbf{x})) = \begin{cases} 0 & \text{for} \quad |y - f(\mathbf{x})| \leq \varepsilon, \\ |y - f(\mathbf{x})| - \varepsilon & \text{otherwise.} \end{cases} \tag{3.17}$$

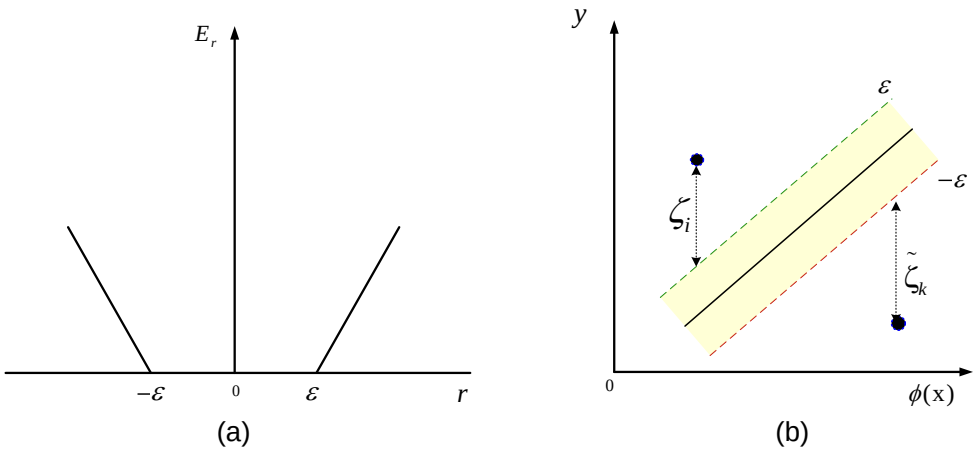


Figure 3.5: (a) The SVM error function, where r is the residual ($r = y - f(\mathbf{x})$). (b) ε-insensitive zone.

As shown in Figure 3.5, the ideal estimation is realized when the absolute residual is within ε (ε insensitive zone), namely

$$|y - f(\mathbf{x})| \leq \varepsilon. \tag{3.18}$$

For feasible solutions, non-negative slack variables $(\zeta, \tilde{\zeta})$ are introduced here as well for the training samples that are outside the ε-tube of radius.

$$\zeta_i = \begin{cases} 0 & \text{for} \quad y - f(\mathbf{x}) - \varepsilon \leq 0, \\ y - f(\mathbf{x}) - \varepsilon & \text{otherwise.} \end{cases} \tag{3.19}$$

$$\tilde{\zeta}_i = \begin{cases} 0 & \text{for} \quad y - f(\mathbf{x}) + \varepsilon \geq 0, \\ -(y - f(\mathbf{x})) - \varepsilon & \text{otherwise.} \end{cases} \tag{3.20}$$

Minimizing $||\mathbf{w}||$ leads to maximizing the margin, the margin here means the farthest distance from the hyperplane to the training samples that are inside the ε-tube. As the margin increases the generalization probability is increasing. Finally, the SVM regression problem is formulated as follows.

$$\begin{aligned} \text{Minimize} \quad & \tfrac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{N}(\zeta_i + \tilde{\zeta}_i) \\ \text{subject to} \quad & y_i - \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i) - b \leq \varepsilon + \zeta_i \\ & \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}_i) + b - y_i \leq \varepsilon + \tilde{\zeta}_i \\ & \zeta_i \geq 0, \quad \tilde{\zeta}_i \geq 0, \quad \forall i. \end{aligned} \tag{3.21}$$

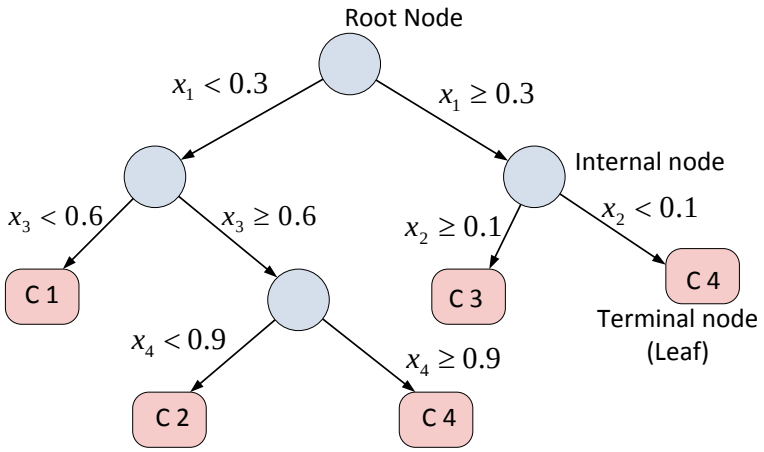


Figure 3.6: Decision tree.

Eq.(3.21) dual problem is then given as follows.

$$\begin{aligned} \text{Maximize :} \quad & \mathcal{W}(\boldsymbol{\alpha}, \tilde{\boldsymbol{\alpha}}) = -\varepsilon \sum_{i=1}^{n}(\alpha_i + \tilde{\alpha}_i) + \sum_{i=1}^{n} y_i(\alpha_i - \tilde{\alpha}_i) \\ & \qquad -\tfrac{1}{2} \sum_{i,j=1}^{n} (\alpha_i - \tilde{\alpha}_i)(\alpha_j - \tilde{\alpha}_j) K(\mathbf{x}_i, \mathbf{x}_j) \\ \text{subject to} \quad & \sum_{i=1}^{n}(\alpha_i - \tilde{\alpha}_i) = 0, \\ & 0 \leq \alpha_i \leq C, \quad 0 \leq \tilde{\alpha}_i \leq C \quad \forall i. \end{aligned} \tag{3.22}$$

Finally, Eq. (3.16) would be written as follows

$$f(\mathbf{x}) = \sum_{i=1}^{n}(\alpha_i - \tilde{\alpha}_i) K(\mathbf{x}_i, \mathbf{x}) + b. \tag{3.23}$$

3.1.3 Random Forest (RF)

Random forest represents a significant enhanced version of the binary decision tree that was invented by Breiman et al. [20]. In his proposed enhancement to produce the RF, Breiman [19] employed three main techniques: bagging, random features selection for splitting, and testing with out-of-bag (OBB) data. Various types of nodes are connected to build up a singles decision tree. Root node is the head node with no incoming edges. Internal nodes come with one incoming edge and outgoing edges as well. Finally, the terminal nodes (leaves) are with only incoming edges, as shown in Figure 3.6. Except the leaves, each node splits the feature space into two or more sub-spaces. This split is done with respect to small set of the

features at a time. It aims to minimize the node impurity $I(N)$ that is defined as follows.

- Regression Impurity.

$$I(N) = \sum_{i=1}^{N} (y_i - t_i)^2, \tag{3.24}$$

  y_i is the node value, t_i is the target value corresponding to the sample i.

- Classification Impurity.
  The node impurity can be measured using one of the three following methods.

  - Entropy Impurity

$$I(N) = \sum_{j}^{N_c} P(c_j) \log P(c_j) \tag{3.25}$$

  - Gini Impurity

$$I(N) = \sum_{j \neq i}^{N_c} P(c_j) P(c_i) \tag{3.26}$$

  - Misclassification Impurity

$$I(N) = 1 - \max_{j} P(c_j) \tag{3.27}$$

The tree will grow till you reach pure node (all the samples share the same class or target value), or you meet a termination criterion such as a maximum deep, minimum number of samples in a node, etc.. At the end, each leaf is assigned to a class that corresponds to the majority of its underlying training samples, in the classification case. In the regression case, the leaf value is set to mean of the target values of its underlying training samples. For testing instance, you navigate it from the root down to the leaf, where the leaf assigned class or value represents the tree prediction value.
In Random trees, Breiman proposes to combine many trees' predictions to counterbalance the high variance of the single tree prediction, as bagging method suggests [18]. In the classification case, each tree votes for a class, and the class with maximum votes is considered the final output of the RT. Averaging the leave value across all trees is the final output in the regression case. Consequently, the prediction accuracy is significantly improved as the number of trees increases. The

training samples of each tree are independently sampled from the training data. This guarantees building identical independent distributed trees (i.i.d) leading to a maximum reduction in the prediction variance through averaging the output of all trees. An average of N i.i.d random variables, each with variance σ^2 is a variable with variance $\frac{\sigma^2}{N}$. While its variance in present of pairwise correlation ρ is given as

$$\rho\sigma^2 + \frac{1-\rho}{N}\sigma^2, \tag{3.28}$$

which is obviously way lower than the former case [67]. To strength the independence among the trees, the features are randomly selected at each node split. Usually, the node splits are performed with respect to the square root of the total number of available features. Assuring the generalization capability of RT, each split is designed with a subset of the data and evaluated with the rest (OOB data). Algorithm 3 summarizes the RF procedure.

Algorithm 1: Random forest for classification and regression.

Data: Training Samples, N_t number of trees, nf number of features to randomly select at each node split, test sample $\mathbf{x}$.

Result: $f_{\mathbf{RF}}(\mathbf{x})$, or $C_{\mathbf{RF}}(\mathbf{x})$

Training
for $i \longleftarrow 1$ **to** N_t **do**
(1) Generate a bootstrap sample from the training samples.
(2) Grow a decesion tree T_i using the generated bootstrap sample, where each node split is built upon nf randomly selected features, and evaluated using OOB data.

Testing
let $T_i(\mathbf{x})$, $C_i(\mathbf{x})$ denote the output of the i tree in the cases of regression and classification, respectively. $In(.)$ is an indicator function.

$$\textbf{Regression}: \quad f_{\mathbf{RF}}(\mathbf{x}) = \frac{1}{N_t}\sum_{i=1}^{N_t} T_i(\mathbf{x})$$

$$\textbf{classification}: \quad C_{\mathbf{RF}}(\mathbf{x}) = \underset{j=1,..,c}{\arg\max} \sum_{i=1}^{N_t} In(C_i(\mathbf{x}) = j)$$

3.1.4 k-Nearest-Neighbor (kNN)

k-Nearest-Neighbor (kNN) is one of the simplest classification methods, where a test sample is classified based on the k closest training samples in the feature

space [35]. This classification method does not depend on underlying joint distribution of the training samples and their labels, which makes it more sensitive to the outliers and redundant data. The nearest k samples are measured with a proper distance metric, usually the Euclidean distance metric. Of course, the features should share a similar scale or be standardized prior the classification. Let $C(a_1), \ldots, C(a_k)$ be the classes of the closest k samples, then the kNN classification can be formulated as follows.

$$C_{k\mathbf{NN}}(\mathbf{x}) = \underset{j=1,\ldots,c}{\arg\max} \sum_{i=1}^{k} In(C(a_i) = j) \tag{3.29}$$

Several modifications had been added to improve the kNN classification accuracy. Dudani [46] proposes to weight each nearest sample with a value w_{ai}, reflecting its closeness (higher value for closer sample), therefore Eq. (3.29) would be rewritten as follows.

$$C_{k\mathbf{NN}}(\mathbf{x}) = \underset{j=1,\ldots,c}{\arg\max} \sum_{i=1}^{k} w_{ai} In(C(a_i) = j) \tag{3.30}$$

A proper value for the number of neighbors k is a trade-off between reducing the sensitivity to noise with large k and preventing the dominance of the neighborhood with small k. As always, the cross-validation method is the optimal choice to adjust it.

3.2 Appearance-based Features

When the human face is captured using a fix-mounted camera, its appearance varies significantly across the head poses and facial deformations. These variations give the appearance a vital rule in any facial analysis method. In what follows, I provide a brief overview of three feature descriptors that have been used successfully to encode the facial appearance.

3.2.1 Gabor Filter-based (GAB) Features

This type of feature has functional similarity to certain cells in the human primary visual cortex, additionally it has a spatial frequency localization property (See Figure 3.7). Its kernel is defined as a Gaussian kernel modulated by a sinusoidal wave as follows.

$$g(x, y; \lambda, \theta, \sigma_x, \sigma_y) = \exp\left\{ -\frac{1}{2}\left(\frac{\acute{x}^2}{\sigma_x^2} + \frac{\acute{y}^2}{\sigma_y^2}\right)\right\} \times \exp\left\{ i\left(2\pi\frac{\acute{x}}{\lambda}\right)\right\}, \tag{3.31}$$

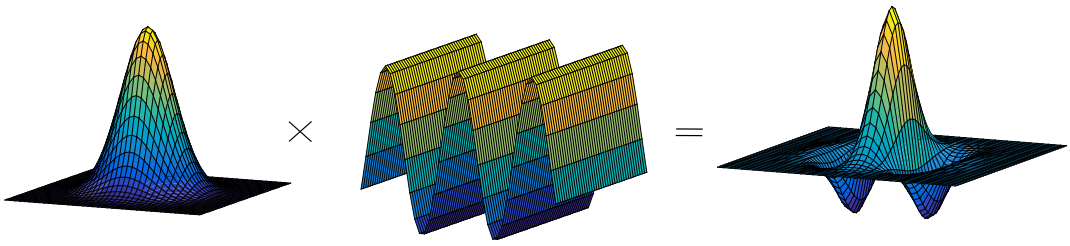


Figure 3.7: Gabor filter: a Gaussian kernel modulated by sinusoidal wave.

where λ is the frequency (in pixel) and θ is the orientation of the sinusoidal function. σ_x and σ_y are the standard deviations along $x-$ and $y-$axis, and $\acute{x} = x\cos\theta + y\sin\theta$, $\acute{y} = -x\sin\theta + y\cos\theta$. Obviously, the real and/or imaginary components of the filter can be derived from Eq. (3.31) and used alone or together. With the tunable orientation response, radial frequency bandwidth besides the joint resolution in space and frequency, GAB features are capable to encode the texture boundaries, discontinuities in phase, and characteristic [15]. To produce GAB features, a Gabor filter bank consisting of various scales, frequencies, standard deviation values, and rotations is convolved with the desired image patch (in gray scale). These convolutions yield a huge number of features. In this work, I scale the resulting patch from each convolution into a fixed size. Then I divide it into smaller cells, where I extract the median value of each cell. Next, I normalize these values to generate the kernal feature vector. Finally, I concatenate the vectors from all kernels to produce the GAB feature vector. The size of those cells along with Gabor filter parameters $(\lambda, \sigma_x, \sigma_y, \theta)$ should be adapted to suit the employing (developing) application. The main method to achieve that is to conduct a cross-validation experiment. GAB has been successfully employed in several applications, e.g. texture segmentation [127, 75], image matching [106], object recognition [177], facial expression recognition [98].

3.2.2 Local Binary Pattern (LBP) Features

The Local binary pattern (LBP) was originally introduced by Ojala et al. [126]. LBPs have been proven to be successful for texture encoding in many computer vision applications, e.g. facial expression recognition [114], face recognition [27],

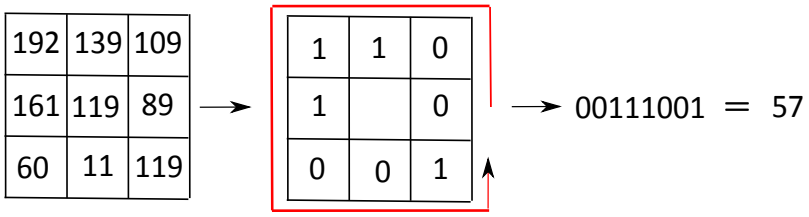


Figure 3.8: LBP operator. Each pixel is thresholded with neighborhood pixel values. The binary results make up the final response.

human detection [116, 161], object detection [121]. When you apply the LBP operator to an image patch, each pixel is labeled by thresholding its value with neighborhood pixel values and then the results are combined and read as a binary number. Let f_c and f_p denote the pixel values at center and neighbored pixels respectively, where $p = 0, ..., 7$. Then, each binary value $B(p)$ of the LBP_v is calculated as follows.

$$B(p) = \begin{cases} 1 & \text{if} \quad f_c \geq f_p \\ 0 & \text{otherwise} \end{cases}. \tag{3.32}$$

Next, a binomial factor 2^p is assigned for each $B(p)$ to get LBP_v as follows.

$$LBP_v = \sum_{p=0}^{7} B(p) \times 2^p. \tag{3.33}$$

After applying the LBP operator to each pixel, I divide the resulting patch into cells, where the LBPs are accumulated in a histogram for each cell. Finally, the concatenation of these neighboring histograms forms the final descriptor. Besides its computational simplicity and illumination invariance, LBP encodes different texture primitives, such as spot, edge, and corner. As the cell size decreases the spatial information dominance over texture understanding. Therefore, it is important to set the cell size via a cross-validation experiment. Figure 3.8 depicts an example of calculating the LBP_v.

3.2.3 Histograms of Oriented Gradients (HOG) Features

HOG feature descriptor was originally introduced by Dalal and Triggs [40], as a variant of Scale Invariant Feature Transform (SIFT) [101], who employed it for

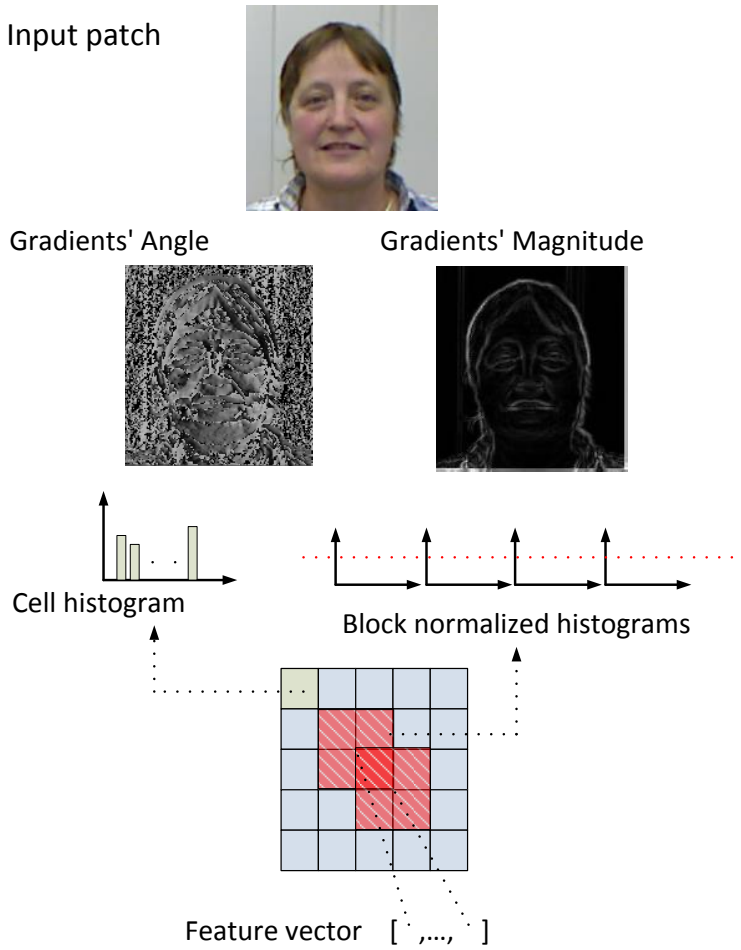


Figure 3.9: HoG features extraction.

pedestrian detection. In this descriptor, the gradient orientations are counted, and the spatial information are kept through dividing the entire patch into smaller regions and then concatenating the region results. Based on that, HOG is capable to encode the underlaying patch appearance. To encode an image patch, first it is divided into small spatial regions called cells. For each pixel in the cell, two gradients are computed using horizontal and vertical Sobel kernels and accordingly, the orientation and magnitude are calculated. Then, for each cell, a 1D histogram of orientation is formed, where each pixel vote is weighted by its magnitude. One can say, each bin in the histogram represents the occurrences of gradients that have orientations within a certain angular range. Trilinear interpolation may be employed here to reduce the aliasing effect [39, 128], where each pixel contributes on 8 bins in 4 neighbored cells. Sliding window, containing more than one cell and named

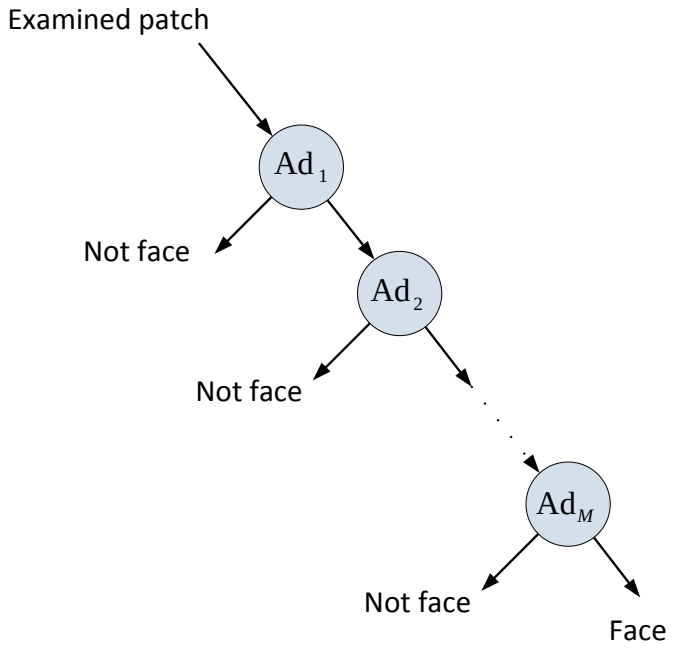


Figure 3.10: Rejection cascade employed in the VJ face detector, each node is an AdaBoost classifier whose week classifiers are decision trees.

block, scans the image with spacing stride measured in cells to extract the final descriptor. At each step, the histograms of the contained cells are normalized to achieve better invariance to illumination and shadowing before contributing to the final descriptor. Several parameters affect the HoG performance such as the size of each of the following: cell; block; and spacing stride, besides the number of the histogram bins. Those parameters are usually estimated with the help of cross-validation experiments. Figure 3.9 sketches the procedure of the HoG features extraction. Besides the pedestrian detection, HoG was successfully employed in several applications, e.g. object detection [151], face recognition [5], face detection [85], head pose detection [172], object registration [31].

3.3 Face Detection

Locating the face inside the processed image is the first step in any automatic facial analysis approach. To achieve that, a discriminative classifier is learned with the help of a database of faces and non-faces patches. The performance of such classifier is affected by the variation of several challenging factors such as pose,

expression, illumination, or races, across the training faces. Throughout this thesis, I employ VJ approach [159] for the face detection, mainly the frontal and profile models available in OpenCV Library [16], in which also the diagonal haar-like features were exploited [94]. These models were trained across expressions, admissible range of poses, illuminations, ethnics, skin tones, and small occlusion cases. In what follows a brief overview of VJ approach is given, including its advantages and shortcoming. VJ face detector is built on top of a rejection cascade of nodes, where each node is an AdaBoost group of decision tree (one level deep) classifiers, as shown in Figure 3.10. Let C_i denote the binary decision of tree i, then the adaboost classifier of n can be formulated as follows.

$$\mathbf{Ad} = \text{sign}(w_1 C_1 + w_2 C_2 + \cdots + w_n C_n), \tag{3.34}$$

Boosting technique is used to calculate the weight values $(w_1, \ldots, w_n)$ as summarized in Algorithm 2. The AdaBoost nodes are arranged to achieve a maximum speed with reasonable performance. Earlier nodes have a lower number of features as they are evaluated the most since each node terminates the patch testing when its output is false. Additionally, they have a higher detection rate which of course at the cost of higher false positives; however, by the end of the rejection cascade a higher detection rate (98%) is achieved with lower false-positive rate (0.0001%) [17].
The VJ detector exploits the Haar-like features, defined as a threshold applied to sum and difference of intensity values of adjacent image regions. Samples of Haar-like feature types are shown in Figure 3.11. The computation of the features is sped by the use of an integral image.

To locate a face inside an image, a sliding window shifts pixel-by-pixel scanning the whole image across various scales for potential faces. Those window scales are parametrized by: minimum search size; maximum search size; and step scale factor, where each face outside the selected scales will be ignored. Then, the positive overlapping windows that passed the minimum neighboring threshold are merged through averaging to produce the final detection results. The search is sped up by scaling the features instead of scaling the processed patch itself. One main drawback of VJ face detector is its inconsistent face cropping and consequently ruining any further automatic analysis. This issue appears when you scan an image containing faces of different scale with fix parameters, or scanning the same image but with different parameters as shown in Figure 3.12. To cope with

Algorithm 2: The boosting algorithm AdaBoost.

Data: N training Samples $(\mathbf{x}, y)$ with binary labels $y_i \in \{-1, +1\}$, $C(\mathbf{x})$ is a binary decision tree based on input vector $\mathbf{x}$ that is one level deep.

Result: $\mathbf{Ad}(\mathbf{x})$

Training

Initialization

$D_1(i) = \frac{1}{N}, \quad i, \ldots, N$ **for** $t \longleftarrow 1$ **to** T **do**

$$C_t(\mathbf{x}) = \underset{C_j \in H}{\operatorname{argmin}} E_j, \quad E_j = \sum_{i=1}^{N} [D_t(i) \times In(y_i \neq C_j(\mathbf{x}_i))]$$

$$E_t = \sum_{i=1}^{N} [D_t(i) \times In(y_i \neq C_t(\mathbf{x}_i))]$$

$$w_t = \frac{1}{2} \log [(1 - E_t)/E_t]$$

$D_{t+1}(i) = [D_{t+1}(i) \times \exp(-w_t y_i C_t(\mathbf{x}_i))] / Z_t$, Z_t is a normalization factor.

Testing

$$\mathbf{Ad}(\mathbf{x}) = \text{sign}\left(\sum_{t=1}^{T} w_t C_t(\mathbf{x})\right)$$

this issue, I propose several post-processing methods, explained in later chapters. False-positive detection can be further rejected by checking the existence of major facial components (Eyes, nose, mouth), or the existence of skin color within the detected patch. Tracking methods are potential solution to mitigate mis-detections while processing an image sequence. Due to the employing of the aforementioned efficient methods, integral image, and rejection cascade, VJ face detector can work in real time. Sharma et al. [146] showed the feasibility of building VJ face detector in real time. Moreover, it has been proven in our lab that it works at 45fps on NVIDIA GeForce GTX 780 by scanning an image of 640×480 pixels with all potential scales.

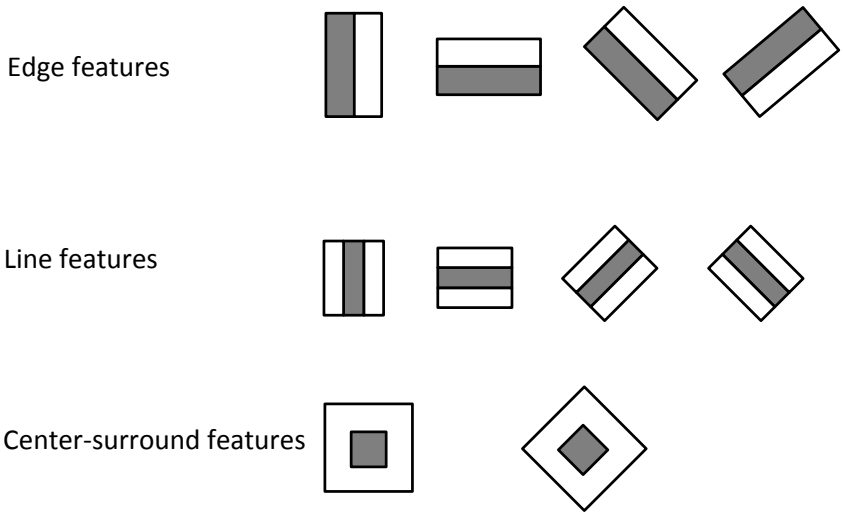
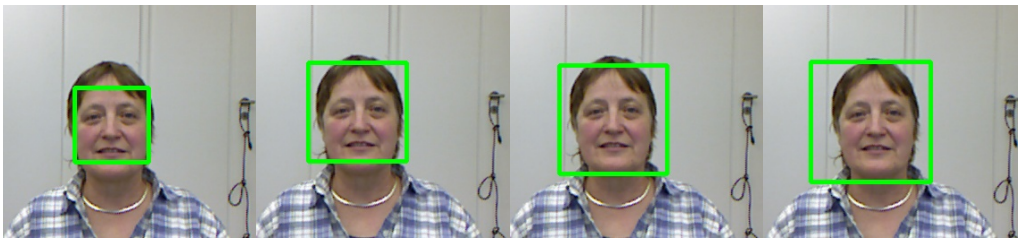


Figure 3.11: Samples of Haar-like features, add intensity values of the light region and then subtract the value of dark region

.



(a) [78 × 78] (b) [111 × 111] (c) [122 × 122] (d) [135 × 135]

Figure 3.12: Applying VJ face detector to an image each time with different searching parameters (e.g. scale step factor) leads always to a different cropping. The size of the returned box is shown beneath each sub-image in pixels. The image was taken from BIWI database.

CHAPTER 4

Databases

Throughout this dissertation, I exploited several databases either to train my proposed methods or to evaluate them in within/cross database scenarios. This section provides a brief description of those databases. I grouped them into three categories according to their employment here, not their eligibility. All of them are publicly available, which ensures the reproducibility of the achieved results.

4.1 Facial Point Databases

To conduct a sophisticated evaluation of our facial point detector, I exploited several databases that vary in illumination, pose, expression, etc.. Below follows a brief overview of those databases.

4.1.1 CMU Multi-PIE

This database was produced by Gross et al. [58], aiming to advance the research in face recognition across poses and illumination conditions; however, it has shown a great benefit for evaluating methods of facial point detection (as used here) and of facial expression recognition. It composes 337 subjects, 264 male and 102 female. 60% of them were European-Americans, 35% Asian, 3% African-American and 2% others, with an average age of 27 years. Each was photographed with 15 views at once under 19 illumination conditions. The data were gathered in four recording sessions, each dedicated for different expressions. A uniform static background is

used in all sessions. The recording was performed using 13 cameras located at the head level with various yaw angles spaced by $15°$, and using two cameras capturing the face from surveillance views. During the recording, each subject was asked to display one of the following facial expressions: smile, surprise, squint, disgust, scream. In total, the database contains 755,370 images from the 337 different subjects. The image resolution is 480×640 pixels, where the mean inter-pupil distance is 78 pixels in the frontal view.

4.1.2 MUCT

This database was produced by Milborrow et al. [112], aiming to advance the research in facial point detection across illumination, age, and ethnicity. It consists of 3755 images of human faces, each manually annotated with 76 facial points. 276 subjects, with equal numbers of males and females and a cross section of ages and races, were photographed with 5 views at once, with a uniform static background. Three cameras were located at head level simulating three different yaw poses, the other two at a higher and lower level of the face simulating 2 pitch angles. 10 different illumination conditions were applied, where each subject was captured in only up to 3 conditions. Some subjects appear with makeup, glasses, and headdresses. They were not asked to display any facial expression; however, a natural smile was presented in some frames. The image resolution is 480×640 pixels, where the mean inter-pupil distance is 88 pixels.

4.1.3 Helen

This database was produced by Le et al. [89], aiming to advance the research in facial point detection on high-resolution images across illumination conditions, poses, and expressions. The images were gathered from Flickr, implying more diversity and consequently, more challenge. The images were collected by making searching on Flickr with different keywords such as family, outdoor, boy, wedding. The search was carried out in different languages to avoid a cultural bias. Only images of faces greater than 500 pixels in width were incorporated. The faces may appear with a proportional amount of background in some samples or being very close so contacting with image edges. In total, the database contains 2330 samples categorized into training part of 2000 samples and testing part of 330 samples.

4.1.4 Head Pose Image

This database was produced by Gourier et al. [56], aiming to advance the research in the head pose estimation across skin tones. It comprises 2790 monocular face images of 15 persons with discrete variations of yaw and pitch angles from $-90°$ to $-90°$, spaced by $15°$. To record the images at each ground truth pose, they put markers inside the room where each marker corresponds to a 2D pose (yaw, pitch), then each person is asked to pose toward each marker for the capturing process. The images are of 384×288 pixels, where the mean inter-pupil distance is 65 pixels. The subjects appear in neutral expression in all frames. I manually annotated part of this database and exploited it for the facial point detection; however, it is originally produced in favor of pose estimation.

4.1.5 AFW

The Annotated Faces in the Wild (AFW) database was produced by Zhu and Ramanan [185], aiming to advance the research in facial point detection across illumination conditions, poses, and expressions. The images were collected from Flickr. It comprises 205 images depicting 468 faces of different scales, viewpoints, glasses, expressions, and skin tones.

4.1.6 LFPW

Labeled Face Parts in the Wild (LFPW) dataset was produced by Belhumeur et al. [14], aiming to advance the research in facial point detection on high-resolution images across illumination conditions, poses, and expressions. The images were collected from the web via straightforward text queries on sites such as Google, Flickr, and Yahoo. In this work, I utilized the part of LFPW available on the i.bug website [2]. In total, it contains 1035 images categorized into training part of 811 images and testing part of 224 images.

4.2 Head Pose Databases

One goal of this dissertation is to develop a head pose estimator that provides a continuous measure of the pose angles on a frame basis. To achieve accurate estimation and consistent face localization, I exploited the depth information. Based

on that, I evaluated our methods on RGBD databases that are accompanied by continues pose annotations for each frame.

4.2.1 BIWI

This database was produced by Fanelli et al. [52]. Each frame instance is represented by two images (RGB and depth), simultaneously stemmed from a Kinect sensor. It comprises 24 sequences of 20 different people (14 men and 6 women, 4 wearing glasses), recorded while sitting about 1 meter away from the sensor. Each subject was asked to rotate his head spanning all possible ranges of the three rotation angles (pitch, yaw, roll). To obtain the pose ground truth for each frame, they track the rotation of each head sequence using a personalized template along with the ICP method. In total, the database contains about 15K frames with rotation angles ranging between $\pm 75^{\circ}$ for yaw, $\pm 60^{\circ}$ for pitch, and $\pm 40^{\circ}$ for roll, where the frames are not uniformly distributed over the pose angles. The images are in the VGA resolution (640×480 pixels), where the mean inter-pupil distance is 45 pixels in the frontal pose.

4.2.2 ICT-3DHP

This database was produced by Baltrusaitis et al. [11]. Each frame instance is represented by two images (RGB and depth), simultaneously stemmed from a Kinect sensor. It contains 10 sequences of 10 different people (6 men and 4 women). To obtain the pose ground truth for each frame, they measured the difference in pose between the current and first frames using Polhemus Fastrack flock of birds tracker attached to a cap the participants were wearing. The Initial frame is assumed to be frontal. In total, the database contains about 1400 frames with rotation angles ranging around $\pm 75^{\circ}$ for yaw, $\pm 50^{\circ}$ for pitch, and $\pm 45^{\circ}$ for roll, where the frames are not uniformly distributed over the pose angles. The images are in VGA resolution (640×480 pixels), where the mean inter-pupil distance is 40 pixels in the frontal pose.

4.3 Facial Expression Databases

Two databases were exploited for the facial expression recognition throughout this dissertation. Both include samples of the six basic expressions: happiness, surprise, anger, disgust, fear, and sadness. The two datasets do not offer an intensity rating of their expressions; however, it is noticeably that the maximum expression intensity varies in one of them. The databases are arranged in sequences showing the evolving of the expression. Our proposed methods provide a frame-based decision about the expressions, consequently, I consider only the frames that depict the apex of each expression.

4.3.1 CK+

The Extended Cohn-Kanade database (CK+) was produced by Lucey et al. [104]. It contains 593 sequences of 123 subjects. Each image sequence starts with onset (neutral expression) and ends with an expression at its peak; however, the sequences vary in duration from 10 to 60 frames. The frames of expressions with full intensity are fully coded by Facial Action Coding System using FACS investigator guide. After applying perceptual judgment to the facial expression labels, only 327 of the sequences were labeled for the human facial expressions: 45 for anger; 18 for contempt; 59 for disgust; 25 for fear; 69 for happiness; 28 for sadness; 83 for surprise. Lucey et al. supplement their database with an annotation of 68 facial point for each frame. They manually annotated key frames within each sequence and then used a gradient descent active appearance model (AAM) to fit these points in the remaining frames. The images are in a resolution (640×490 pixels), where the mean inter-pupil distance is 130 pixels.

4.3.2 BU-4DFE

Binghamton University 3D dynamic Facial Expression (BU-4DFE) database was produced by Yin et al. [174]. It comprises 101 subjects displaying the six basic expressions (happiness, surprise, anger, disgust, fear, and sadness) in 606 image sequences. The subjects are 58 females and 43 males, with a variety of ethnics. Each expression sequence contains about 100 frames [174], beginning with the onset, continuing through the apex, and ending with the offset. Additionally, 3D sequences are provided, which allow rendering sequences of different poses. To

produce their dataset, they employed a system of two stereo cameras and one texture video camera. The three cameras are placed on a tripod with two lighting lamps located in the two sides. For calibration and background segmentation, they used a uniform blue background with a calibration board. The produced sequences are at a speed of 25 frames per second. The images are in a resolution of 1040×1329 pixels, where the mean inter-pupil distance is 300 pixels. Each 3D model of a 3D video sequence has the resolution of approximately 35,000 vertices. Each subject was requested to perform the six universal expressions with the guidance of a psychologist; however, the intensity of the apex frames varies noticeably across subjects in comparison with CK+ database.

CHAPTER 5

Facial Point Localization

Evaluating each facial point using only a local patch enclosing it shall perform poorly, since a small patch is not discriminative enough as similar local patches could exist in different locations within the face as well. Moreover, calculating the local response for each potential point location (or some of them) and optimizing the final search through a global shape model are time-consuming processes. Factually, far patches hold cues about a facial point as local patch enclosing it does, they are more useful in situations of bad illuminations and non-frontal views. Consequently, I propose a cascade regression based method, in which features extracted from the entire face patch participate in the predication of each facial point. In particular, I propose a method to locate 49 facial points via neural networks in a cascade regression framework. Those points describe 4 main components of the face (eyes, eyebrows, nose, and mouth) in adequate detail to infer most of the facial action unites [49], facilitating any further facial analysis. The performance of the proposed approach is enhanced due to:

- performing a guided initialization rather than employing the mean shape or any random shape, accordingly the approach is prevented from falling in a non converge situation especially for the faces with higher pose angles,

- performing a feature selection at each iteration prior the regression step, which improves the generalization capability of the approach,

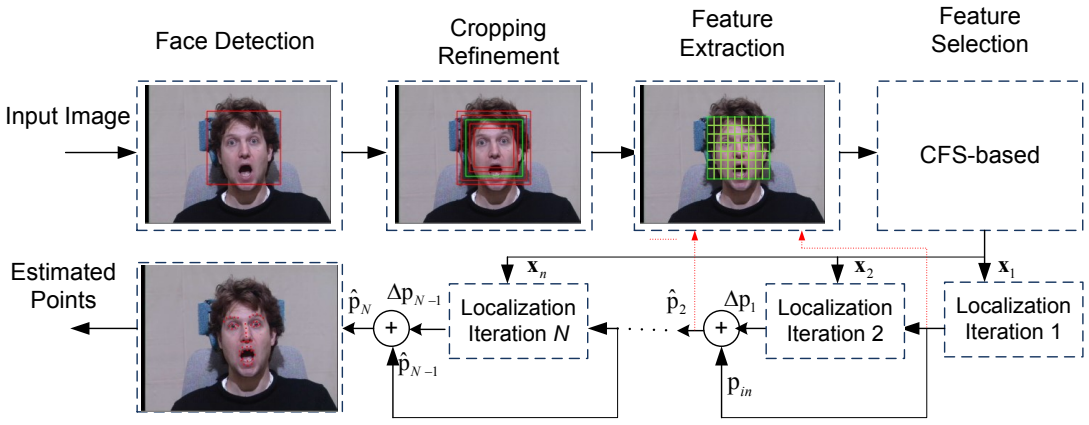


Figure 5.1: Workflow of our proposed approach for the facial point detection.

- performing a cropping refinement, which stabilizes the face detector output leading to more accurate localization.

Figure 5.1 depicts the workflow of our proposed approach for automatically locating 49 facial points. First, I detect the face. Then, I perform a cropping refinement task guaranteeing a unified perspective of the face across scales. Next, I extract texture features (HoG) from the entire face patch, where some selected features are fed into the neural networks to initialize the point location. Next, texture features are extracted from patches each encloses one priorly located point. Then only the representative features are fed into the next neural network to infer the displacement to final location. I iterate the latter process three times more before considering its output as the final estimated location. In what follows, I give more detail about each stage of the approach. To assess the performance of the proposed approach, I perform a comprehensive evaluation, including fair comparisons with state-of-the-art approaches and commercial software packages. Moreover, special experiments are conducted to validate the proposed enhancement to the cascade regression frame work, e.g. the use of feature selection, guided initialization, cropping refinement.

5.1 Face Cropping

To spot the face, I employ the frontal model of VJ detector, available in the OpenCV Library. With this model faces of poses within $\pm 30^\circ$ pitch, $\pm 20^\circ$ roll, $\pm 40^\circ$ yaw can be detected with high detection rates [140]. Due to the scanning mechanism, a consistent cropping across scales is not guaranteed, see Sec. 3.3. To overcome this issue, I perform a cropping refinement task on the detector outputs. To this end, a multivariate Gaussian Mixture Model (GMM) for the face is built from face patches that were uniformly cropped. Those patches vary in pose, illumination, scale, expression. The likelihood of a face patch given the face model is given as follows.

$$p(\mathbf{x}|\Phi) = \sum_{i=1}^{m} \alpha_i p_i(\mathbf{x}|\phi_i), \tag{5.1}$$

where $\mathbf{x} = (x_1, ..., x_d)' \in \mathbb{R}^{d\times 1}$ is a texture-based feature vector, extracted from the entire candidate patch. Here HoG features are exploited, see Sec. 3.2.3. $\phi_i = (\mu_i, \Sigma_i)$, $\Phi = (\alpha_1, ..., \alpha_m, \phi_1, ..., \phi_m)$ is the face model, which was estimated via the Expectation Maximization (EM) algorithm [44]. Each p_i represent a d-dimensional multivariate Gaussian distribution given by

$$p_i(\mathbf{x}|\phi_i) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma_i|^{1/2}} \exp\{-\frac{1}{2}(\mathbf{x}-\mu_i)^{\mathrm{T}}\Sigma_i^{-1}(\mathbf{x}-\mu_i)\}, \tag{5.2}$$

$\mu_i \in \mathbb{R}^{d\times 1}$ is the mean vector of the i^{th} subpopulation; Σ_i is its $d \times d$ covariance matrix. $\alpha_i \in [0, 1]$ for all i and the α_i's are constrained to sum to one.

Since the VJ outputs vary only in scale and share the same center (see Figure 6.4b), only cropping refinement across scales is performed. To this end, for each detected face by VJ detector, I examine several windows sharing the detected center but with different scales. The face window of the maximum likelihood value is then considered for further process. Let $\mathbf{x}_s$ denote the extracted features with respect to size s, then the chosen feature vector for further process is given as

$$\mathbf{x} = \arg\max_{\mathbf{x}_s} p(\mathbf{x}_s|\Phi). \tag{5.3}$$

Eq. (5.3) is simplified to

$$\mathbf{x} = \arg\max_{\mathbf{x}_s, i} \log(p_i(\mathbf{x}_s|\phi_i)). \tag{5.4}$$

Eq. (5.4) is less sensitive to the unbalanced training data in comparison to Eq. (5.3), as the dependency upon the subpopulation weight α_i is removed. To perform a fast calculation of $\log(p_i(\mathbf{x}_s|\phi_i))$, the features are assumed to be independent. A satisfactory accuracy is obtained by evaluating only six scales and building a face model of 15 subpopulations. This refinement task can be performed on top of any face detector with a wider range of poses as well. It can be used as a standalone detector in which a search across scales and spatial locations is then required, but in this dissertation it is only used to refine the output of the VJ face detector.

5.2 Feature Extraction and Selection

Obviously, each configuration of the facial points has its own different impact on the face appearance. Consequently, it is reasonable to employ appearance-based features here. Throughout this chapter, I employ the HoG descriptor as it shows superior performance in comparison to other texture-based features in encoding the face appearance, refer to Sec. 6.2 for more detail.

As the feature set may incorporate redundant and irrelevant features. Redundant features correlate with others and therefore, provide no more predictive information [87]. Irrelevant features provide no predictive information at all [6]. In this approach, I perform a feature selection to nominate only the optimal predictive features leading to:

- enhance the model performance in terms of complexity, allocated memory size, and prediction time,
- improve the model generalization capability.

Many filter and wrapper methods have been developed to perform the feature selection task [62, 99]. Wrapper methods aim to minimize a loss function developed based on the predicted values of the regressors [13, 118]. Here, I adapted a Correlation-based Feature Selection (CFS) method [63] to select the more predictive features. Its advantages are as follows:

- it has relatively a lower computational cost owing to a directly employment of both the original features (not a transformed version of them) and the ground truth value (not the predicted value from the regressor),

- it has only few parameters to be configured.

The core idea of the developed feature selection method is to select features that are highly correlated with the ground truth value and lowly with each other. First, the correlation coefficients between each feature pair (x_i, x_j) were computed as follows.

$$R(x_i, x_j) = R(x_j, x_i) = \frac{C(x_i, x_j)}{\sqrt{C(x_i, x_i)C(x_j, x_j)}}, \tag{5.5}$$

where $C(x_i, x_j)$ denotes the covariance of x_i and x_j and is defined as follows.

$$C(x_i, x_j) = \frac{1}{N-1}\sum_{m=1}^{N}(x_{im} - \bar{x_i})(x_{jm} - \bar{x_j}), \tag{5.6}$$

where $\bar{x_i}$ is the mean value of x_i over the N training samples, x_{im} denotes the m^{th} sample of the feature x_i. Additionally, the correlation coefficients between the target output r and each feature $R(r, x_i)$ were computed. r, here, is the ground truth value of the facial point location within the face patch in the first iteration, or the displacement of a priorly estimated point from the corresponding ground truth value in the following four iterations. Let N_s denote the aimed number of the selected features. Those N_s features must have the maximum $|R(r, x_i)|$ and satisfy

$$|R(x_i, x_j)| < \tau_{th} \qquad \text{for all selected features.} \tag{5.7}$$

τ_{th} is iteratively adjusted, in a way preventing the algorithm from falling in an infinite loop, to ensure that the number of features satisfying Eq. (5.7) is ranging between N_s and $1.1 \times N_s$. $|x|$ denotes the absolute value of x. The whole selecting process is summarized in Algorithm 3.

5.3 A Cascade of Neural Networks

Five neural networks in a cascade were exploited to infer 49 facial points from appearance-based features. A guided initialization, performed in the first neural network, prevents the approach from falling in a non converge situation especially for faces of high pose angles. In particular, I employ the feed-forward neural network (known as Multi-Layer Perceptrons (**MLP**)) owing to its better efficiency, either in training or testing. Moreover, the **MLP** models occupy considerably lower memory size compared to **SVM** models. Random Trees (**RT**) method shares the **MLP** the latter advantage but has lower generalization capability, empirically proven.

Algorithm 3: Correlation-based feature selection algorithm. Adjustment of τ_{th} is done in a way preventing the algorithm from falling in infinite loop. $R(x_i, x_j)$ is the correlation coefficient between the feature pair (x_i, x_j)

Data: Training Samples $(\mathbf{x}_1, r_1), ..., (\mathbf{x}_N, r_N)$, N_s the aimed number of features to be selectetd out of N_t

Result: $List(x_1, ..., x_{N_s})$ (list of selected features)

for $i \longleftarrow 1$ **to** N_t **do**
    (1) Calculate $R(r, x_i)$.
    (2) **for** $j \longleftarrow i$ **to** N_t **do**
        Calculate $R(x_i, x_j)$

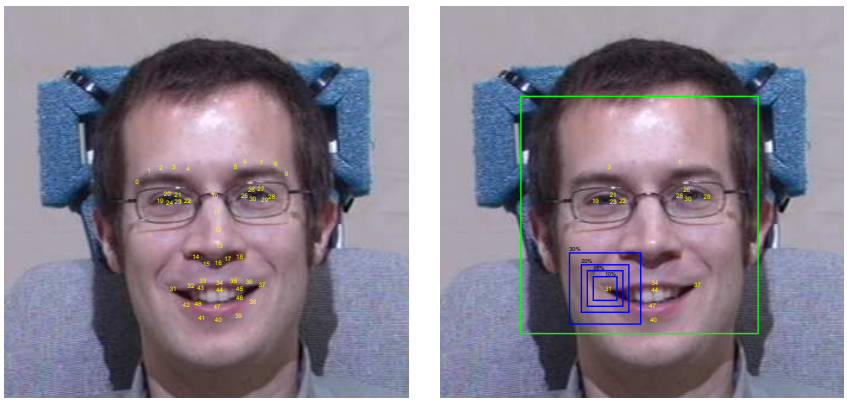
- *sort* the features $(x_1, ..., x_{N_t})$ with respect to $|R(r, x_i)|$ in descending order

while *true* **do**
    $N_{lf} = 0$ (number of listed features)
    Empty $List()$
    for $i \longleftarrow 1$ **to** N_t **do**
        x_i=$get\_sorted\_feature(i)$
        $Append\_state = true$
        for $j \longleftarrow 1$ **to** N_{lf} **do**
            if $|R(x_i, List(j))| > \tau_{th}$ **then**
                $Append\_state = false$
                break
        if $Append\_state$ **then**
            $append\ x_i$ to $List()$
            $N_{lf} = N_{lf} + 1$
    if $N_s \leq N_{lf} < (1.1 \times N_s)$ **then**
        truncate the list to the first N_s features
        break
    else
        $adjust(\tau_{th})$

I defined the point location relatively to the center of the cropped face patch. In the first **MLP**, HoG features extracted from the entire patch were mapped into the ground truth values of the point location. Precisely, the face patch was scaled to 200×200 pixels before being divided into cells of 20×20 pixels. For each cell, a histogram of eight orientation bins was computed. Blocks of four cells and a block spacing stride of one cell were employed. The final feature vector is of length 2592. Next, only 1000 features for each component of each facial point were selected via the modified CFS method explained in Section 5.2, where the target output r is the ground truth of the facial point location. $x-$ and $y-$ coordinates were separately and independently considered. Finally, a regression-based **MLP** of one hidden layer is trained to map the features to the point location. The built model is then used to estimate the initial location of the facial points as follows.

$$p_{in_{ik}} = \acute{h}\left(\sum_{j=1}^{M} w_j^{(2)} h\left(\sum_{m=1}^{N_s} w_{jm}^{(1)} x_m^{p_{in_{ik}}} + w_{j0}^{(1)}\right) + w_0^{(2)}\right), \quad i = 1, ..., 49;\ k \in \{x, y\}. \tag{5.8}$$

$\mathbf{p}_{in} = (p_{in_{1x}}, p_{in_{1y}}, ..., p_{in_{49x}}, p_{in_{49y}})$ denotes the points' initial locations. $\mathbf{x}^{p_{in_{ik}}} = (x_1^{p_{in_{ik}}}, ..., x_{N_s}^{p_{in_{ik}}})$ represents the selected features for the component $p_{in_{ik}}$. $h(.)$ is the sigmoidal function, $\acute{h}(.)$ is the sigmoidal function followed by a proper scaling function. M is the total number of the hidden layer units (empirically set to 10). The weights of both layers $(w_{\cdot}^{(2)}, w_{\cdot\cdot}^{(1)})$ were estimated using the back propagation algorithm. Next, the point locations were refined through four **MLP**s. At each **MLP**, a HoG feature vector of length 128 is extracted from each patch of a set of patches centered at the estimated point location from the previous **MLP**. The sizes of those patches are defined relatively to the width of the face patch and decreased after each **MLP**; it is set to 30% of the face width in the second **MLP**; 20% in the third, 15% in the fourth, and 10% in the fifth **MLP**, see Figure 5.2b. Starting with bigger patch size helps to cope with occlusions and pose variations leading to faster convergence. While ending with small patch size improves the localization resolution. For each component of each facial point in each refinement **MLP**, I nominated only the most predictive 1000 features out of the extracted feature vector of length 6272 (128×49) via the modified CFS method, where the target r, this time, is the displacement from the estimated point in the earlier **MLP** to the ground truth of the point location. In accordance with Eq. (5.8), a **MLP** of one hidden layer is exploited to map the selected features to the corresponding displacement $\Delta\mathbf{p}_{it} = (\Delta p_{it_{1x}}, \Delta p_{it_{1y}}, ..., \Delta p_{it_{49x}}, \Delta p_{it_{49y}})$. Finally, the estimated points



(a) (b)

Figure 5.2: (a) The 49 facial points detected by our proposed approach. (b) The 16 facial points used for comparison with the state-of-the-art approaches in Sec. 5.4.2. The resulted box from the cropping refinement process is depicted in green. The blue boxes depict the considered patch size around each facial point, their size decreases for each added **MLP**.

$\hat{\mathbf{p}}_N$ after N iterations is given as follows.

$$\hat{\mathbf{p}}_N = \mathbf{p}_{in} + \sum_{it=2}^{N} \Delta \mathbf{p}_{it-1}. \tag{5.9}$$

All the parameters, the neural network parameters (e.g. the number of hidden layers and their units; activation function); the HoG parameters (e.g. cell, block, and stride sizes); CFS parameters (e.g. number of selected features), were optimized using a grid-search along with five fold cross-validation carried out only on the training set with a goal of obtaining an accurate localization at reasonable processing and resource cost.

5.4 Experimental Results and Analyses

Several experiments were carried out to evaluate the proposed approach, demonstrating its effectiveness and generalization capability. The experiments were conducted in both within and cross database scenarios. A fair comparison with state-of-the-art approaches and two commercial software packages was carried out in terms of accuracy and efficiency. Additionally, the performance of the developed approach is investigated under various scenarios validating the proposed enhancement to the cascade-regression based methods.

5.4.1 Cross-validation for the Proposed Method

To evaluate the proposed method, I exploited 6500 samples collected from Helen [89], Multi-PIE [58], Head pose [56], AFW [185] databases. Those samples vary in illumination, pose, resolution, skin tone, facial expression, age, and other factors. The face is detectable via VJ detector in all samples. 49 facial points were manually labeled as shown in Figure 5.2a. In the evaluation, the whole samples were divided into 10 folds. 9 folds were used as a training set and the remaining fold as a testing set, taken into consideration samples of the same person do not exist in both sets. The average results on the testing folds were considered and reported. The detection error of each point is computed as follows.

$$\text{Err} = \frac{\sqrt{(p_x - \hat{p_x})^2 + (p_y - \hat{p_y})^2}}{l}, \tag{5.10}$$

where, (p_x, p_y) and $(\hat{p_x}, \hat{p_y})$ are the ground truth and the detected location, respectively. l is the width of face patch, stemmed from cropping refinement process, as shown in Figure 5.2b. $\text{Err}_{x\text{avg}}$ denotes the detection error averaged over x points.

The cross-validation for the proposed approach were performed twice. The first time using the proposed combination of **MLP** and the modified CFS method, while the second time using **RT** as a regression-based method. The latter method is built on top of its own wrapper feature selection technique whose loss function hinges on the predicted values. The average errors for each facial point using both regression-based methods are presented in Table 5.1. The point localization is more accurate using the **MLP** than the **RT**, the average localization errors over all facial points were 0.98% and 1.15%, respectively. For all further analyses, I considered only the proposed **MLP** in combination with the modified CFS. The average error of the eyes' points ($\mathbf{p}_{19}, ..., \mathbf{p}_{30}$), using **MLP**, is 0.83% highlighting the approach capability of detecting eye signals, e.g. eye blinking. The average error of the nose and eyes' points (0.84%) is lower than that of the mouth and eye brows' points (1.1%), which is reasonable since the latter points are subject to higher deformations.

5.4.2 Cross-database Validation and Comparisons

In this section, cross-database validation and comparison with state-of-the-art methods are conducted. The model trained on the collected data in the previous

Table 5.1: The average detection error ($\overline{\text{Err}}$) for each facial point, where the **RT** column represents the localization mean error when the random tree regressor was used, and **MLP** column when the combination of **MLP** and the modified CFS method was used. This cross-validation experiment was carried out on the collected database. The point number (P-ID) is as shown in Figure 5.2.

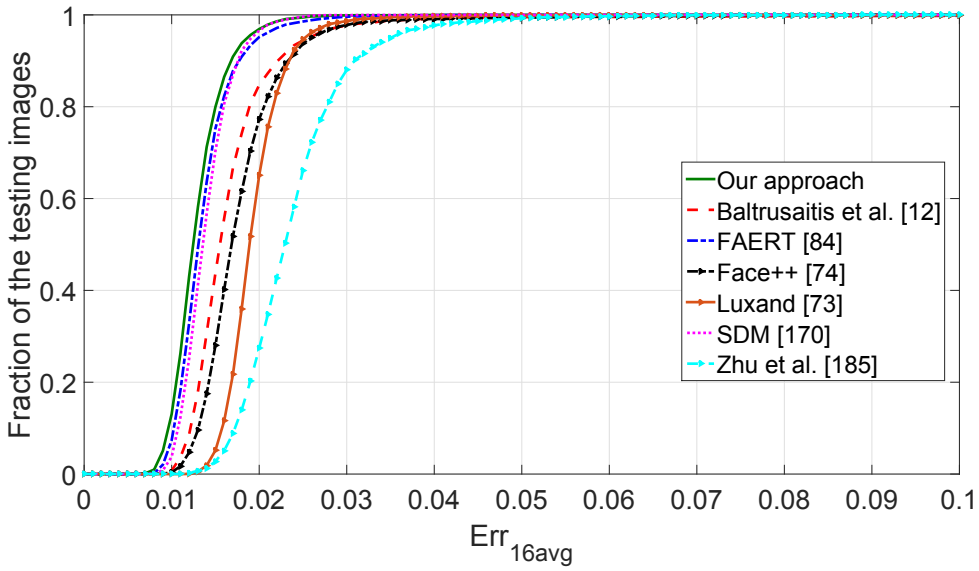
P-ID	**RT** $\overline{\text{Err}}$%	**MLP** $\overline{\text{Err}}$%	P-ID	**RT** $\overline{\text{Err}}$%	**MLP** $\overline{\text{Err}}$%	P-ID	**RT** $\overline{\text{Err}}$%	**MLP** $\overline{\text{Err}}$%	P-ID	**RT** $\overline{\text{Err}}$%	**MLP** $\overline{\text{Err}}$%
0	1.43	1.37	1	1.35	1.17	2	1.05	0.90	3	1.13	**0.81**
4	1.26	1.15	5	1.18	1.11	6	1.14	0.92	7	1.18	1.03
8	1.18	1.11	9	1.35	1.27	10	0.98	0.81	11	0.95	**0.72**
12	1.01	**0.84**	13	1.34	1.12	14	1.09	0.90	15	0.92	**0.82**
16	0.99	0.86	17	1.03	0.85	18	0.96	**0.82**	19	1.02	0.91
20	1.07	0.88	21	**0.83**	**0.72**	22	0.89	**0.80**	23	0.97	**0.84**
24	0.92	0.87	25	1.01	**0.84**	26	0.89	**0.78**	27	0.98	0.87
28	1.02	0.91	29	0.89	**0.83**	30	0.91	**0.78**	31	1.19	1.02
32	1.04	0.89	33	1.09	0.89	34	1.22	1.01	35	1.23	0.99
36	1.37	0.97	37	1.24	1.08	38	1.31	1.12	39	1.42	1.28
40	1.56	1.37	41	1.42	1.28	42	1.35	1.15	43	1.33	0.92
44	1.17	0.93	45	1.27	1.01	46	1.49	1.30	47	1.49	1.27
48	1.51	1.29	-	-	-	-	-	-	-	-	-

experiment was applied on MUCT database [112]. Our results and those of the state-of-the-art [185, 12, 170, 84] and two commercial software packages [73, 74] are presented here. In this evaluation, I use their pre-trained models that are publicly available. They support a wider range of poses compared to [108] which is constrained only to the frontal faces and consequently, not involved in the comparison. The pre-trained model of [84] that is available in Dlib Library [85] was exploited in this evaluation. Zhu et al. [185] provide three pre-trained models; I employed here the most accurate model, the *multipie-independent* model. The samples of MUCT database do not appear in the training data of our model, while its unknown which databases were used to train the publicly available models of some of the evaluated approaches. Consequently, this database is suitable for the cross-database validation and comparisons. For the comparison, I considered only

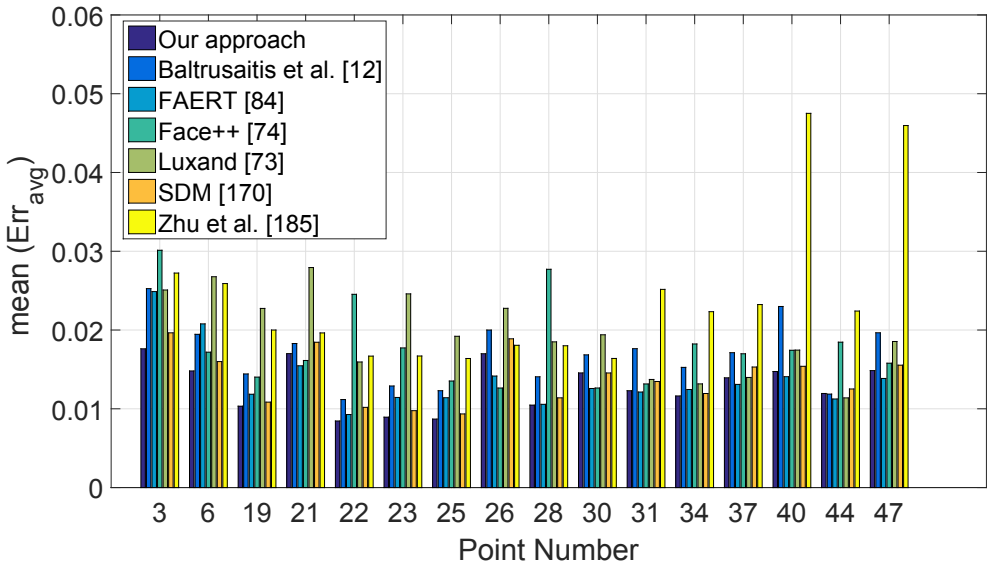
16 facial points, shown in Figure 5.2b, since they are common among all the evaluated approaches and the available ground truth. The face was detected in 3593 out of 3755 images by all approaches; consequently, only those detected faces participate in this evaluation. The error of each estimated point is calculated by Eq. (5.10), where l here is the face width according to our detection. Figure 5.3a presents the cumulative proportion of the images that were within a certain average error of the chosen 16 facial points ($\text{Err}_{16\text{avg}}$). Obviously shown that our proposed approach is slightly superior to those of Baltrusaitis et al. [84] and SDM [170], while it is significantly more accurate than that of Zhu et al. [185]. Additionally, our results are more accurate than those provided by the two commercial software packages [73, 74] and the approach of [12]. I present the mean error of each estimated facial point using all evaluated approaches in Figure 5.3b. Our approach provides almost the most accurate estimation. The localization accuracy of our approach here is as accurate as in the previous experiment, which proves the better generalization capability of our approach among all evaluated approaches.

5.4.3 Comparisons According to the 300-w Competition

In this experiment, I conduct a comparison with publicly known setup according to the 300-w competition. Here, the approaches were trained and tested with the same sets. In a similar way to [155], I trained all models using both Helen and LFPW training sets and then evaluated the resulting model using their testing sets separately. I refer to the training and testing sets of Helen and LFPW that are available on the i.bug website [2]. This evaluation was conducted for the case of 49 facial points. The achieved results along with those of [155, 170, 84, 73, 74] are depicted in Figure 5.4a for LFPW testing set and in Figure 5.4b for Helen testing set. The non-detected faces using our approach, 4.87%, were manually cropped with a similar perspective. The approach of Tzimiropoulos et al. [155] was applied to the test images based on perfect cropped patches provided by them. The results of [170] are as reported by [155], stemmed using the same setup. The approach of [84] was retrained using only Helen and LFPW training sets as the other approaches, where the faces were cropped according to our method. For the two commercial software packages, I employed their pre-trained models; the non-detected faces were omitted from their results. Additionally, I removed the non common points (3 points in [74] and 8 points in [73]). Therefore, $\text{Err}_{49\text{avg}}$ is actually $\text{Err}_{46\text{avg}}$ for



(a)



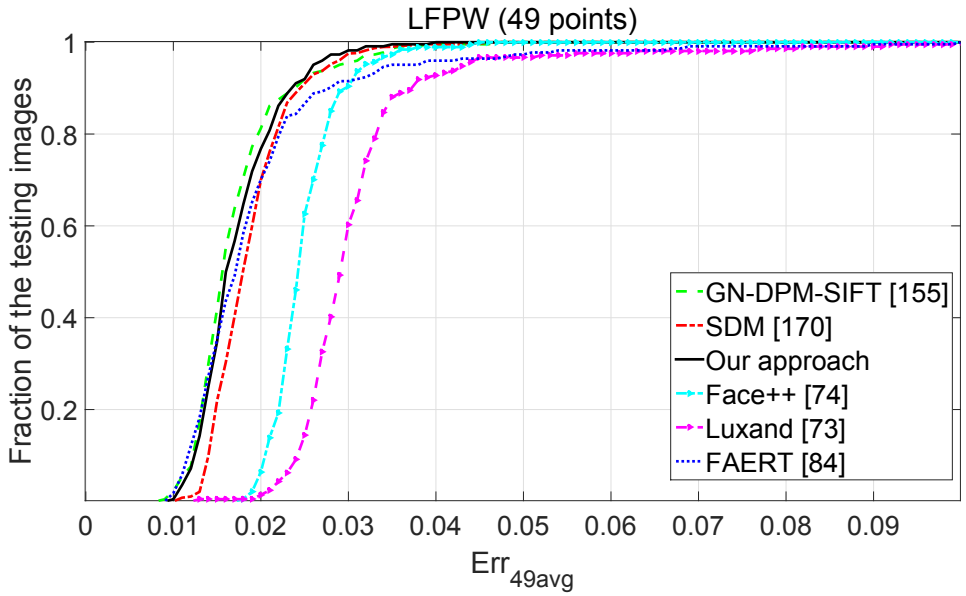
(b)

Figure 5.3: (a) Cumulative proportion of the images that are within a certain average error of the chosen 16 facial points (Err_{16avg}). (b) The mean error for each facial point. This cross-database experiment was carried out on the MUCT database [112]. Our models were trained on a data gathered from other 4 datasets, while we use the pre-trained models of [12, 84, 74, 73, 170, 185] that are publicly available.

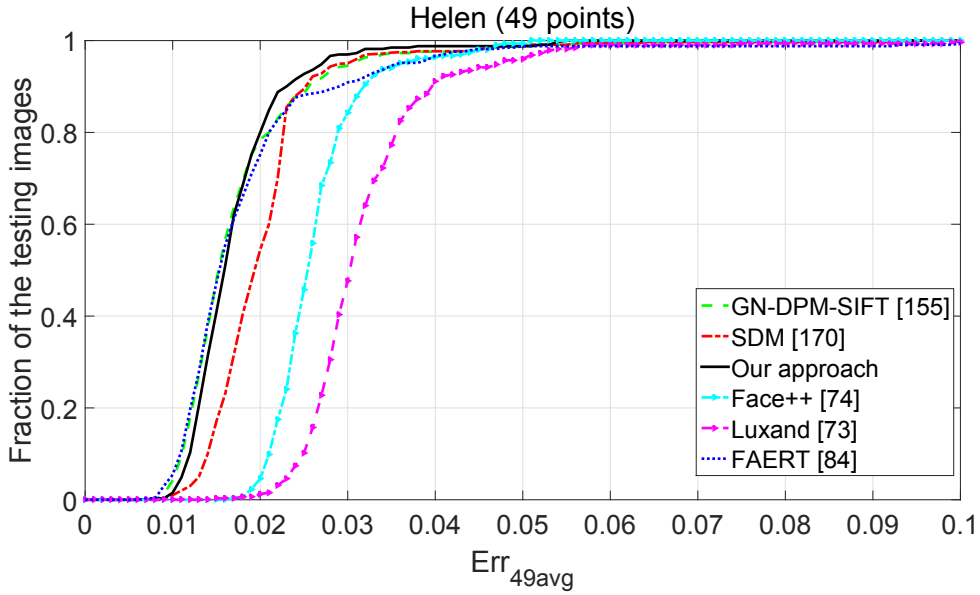
[74] and $\text{Err}_{41\text{avg}}$ for [73]. Following the evaluation in [155], the localization errors were computed relatively to the face width that is derived from the dataset annotation, as the minimum width of a rectangle enclosing all the offered facial points (68 points). In particular, the face width is the difference between the maximum and minimum x components of the offered 68 annotated facial points. A number of points can be drawn from the depicted results in Figure 5.4. The proposed approach achieved more accurate results than those of SDM [170] and two commercial software packages [73, 74] on both datasets. In comparison to GN-DPM-SIFT [155], slightly fewer samples with localization accuracy ($\text{Err}_{49\text{avg}}$) less than 0.024 in LFPW and 0.018 in Helen were obtained. Similarly in comparison to [84], fewer samples with accuracy ($\text{Err}_{49\text{avg}}$) less than 0.015 in LFPW and 0.016 in Helen were obtained, but more samples as $\text{Err}_{49\text{avg}}$ increases. Obviously, these results highlight the fact that the proposed approach has a better generalization capability of among the other considered approaches.

5.4.4 The Efficiency Analysis

Here, I investigate the processing times both to detect a face and to locate the facial points for each method involved in the accuracy comparisons in Sec. 5.4.2 and Sec. 5.4.3. Those two times besides the face detection rate (**FDR**) are summarized in Table 5.2. The processing time involves both face detection (**FD**) time and point localization (**PL**) time, each is the average time required to process one frame of 640×480 pixels by a machine of Intel quad Core 2.33 GHZ, 8 GB RAM, under Windows 7 environment. The values of **FDR** were obtained by applying each approach to the test sets of Helen and LFPW databases. All the rates vary between 94.58% and 97.47%. The approach of [155] was applied to a predefined face location and size; consequently, no value for **FDR** is presented for it. The VJ face detector, employed by the proposed approach and [170], achieved a detection rate of 95.13%. The developed face detector by [85] is superior to that of VJ by 2.34% since it was trained on wider poses, but at cost of more processing time.
The propose approach here consumes $0.135s$ to both locate and crop the face consistently, in which all the scales starting from 24×24 pixels are evaluated. While the face detector of [85] consumes $9.08s$, [175] $3.56s$, [73] $0.91s$. VJ detector would perform faster if it is configured to detect only one face as in [170] or if I use its GPU implementation. Zhu and Ramanan [185] perform the face detection, pose



(a)



(b)

Figure 5.4: Cumulative proportion of the images that are within a certain average error of the 49 facial points (Err_{49avg}), where the error is normalized to face width derived from the datasets annotation. The figures depict our estimation results in comparison to those of state-of-the-art GN-DPM-SIFT [155], SDM [170], FAERT [84], Face++[74], and Luxand [73]. (a) The results of applying models trained on the Helen and LFPW training sets on the LFPW test set (b) The results of applying models trained on the Helen and LFPW training sets on the Helen test set. Except for Face++[74] and Luxand [73] we employed their pre-trained models and plotted Err_{46avg} for Face++[74] and Err_{41avg} for Luxand [73].

Table 5.2: The process time of the facial point detectors in terms of face detection time (**FD**) and facial point localization time (**PL**). The presented time is the average time required to process one frame of 640×480 pixels by our machine (Intel quad Core 2.33 GHZ, 8 GB RAM, under Windows 7 environment). The face detection rates (**FDR**) were obtained by applying each approach to the test sets of helen and LFPW database.

Approach	**points**	**FD** (s)	**PL** (s)	**FD+PL** (s)	**FDR** $(\%)$
GN-DPM-SIFT [155]	68	-	0.3240	-	-
SDM [170]	49	-	-	**0.0841**	95.13
Zhu et al. [185]	68	-	-	111	96.57
Baltrusaitis et al. [12]+[175]	68	3.5651	0.8037	4.3647	96.21
Face++[74]	83	-	-	-	94.58
Dlib[85]+ FAERT[84]	68	9.08	**0.03**	9.11	**97.47**
Luxand [73]	66	0.91	0.94	1.85	96.57
Our approach	49	**0.135**	**0.11**	**0.245**	95.13

estimation, and facial point localization simultaneously in $111s$. The commercial software package of Face++ [74] works on a remote server, consequently, one cannot get comparable values for its processing times.

The approach of [84] is the fastest in locating the facial points within a cropped face patch consuming only $0.03s$ to locate 68 facial points. The SDM approach [170] consumes $0.084s$ to detect the face and locate 49 points; it configures the VJ detector to search for one face. The proposed approach here (unoptimized code) expends $0.11s$ to locate 49 points within the face patch. This time is distributed almost equally among the 5 iterations. The commercial package of [73] takes $0.94s$ to locate 66 facial points, while the approaches of [155] and [12] takes $0.32s$ and 0.80 to locate 68 points, respectively. Complementing to the processing times shown in Table 5.2, the circumstances of the evaluation of each approach are summarized in Table 5.3.

5.4.5 Analyses of the Proposed Approach

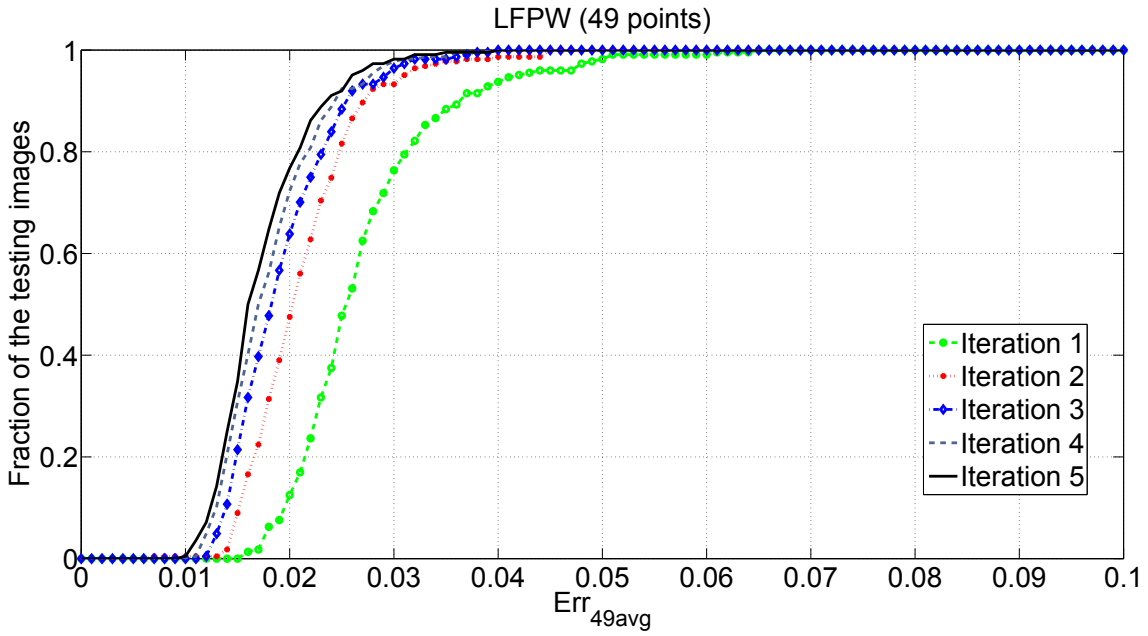
Here, I validate the proposed enhancement to the regression-based methods. In particular, I investigate the performance of the proposed approach under the impact of employing a guided initialization, a feature selection, a face cropping refinement. Additionally, I present the achieved localization accuracy across iterations.

Table 5.3: Complementary notes to Table 5.2 describing the circumstance of the evaluation of each approach.

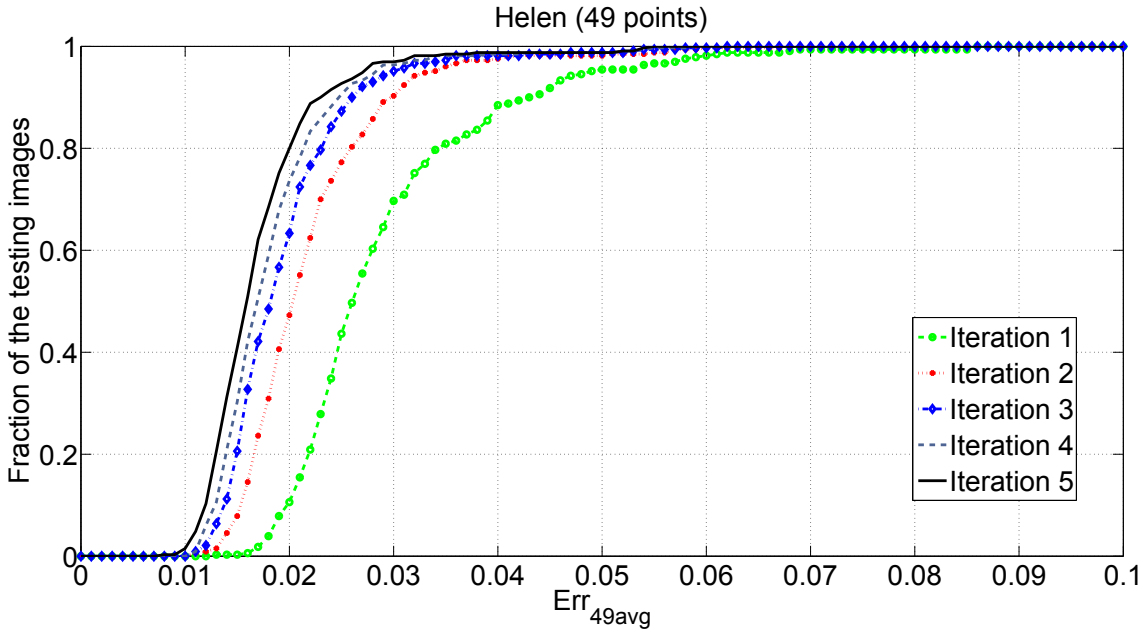
Approach	Note
GN-DPM-SIFT [155]	the face annotations (size and location) were provided along with the trained models
SDM [170]	his approach is built on top of VJ face detector and is optimized to detect only one face in the image.
Zhu et al. [185]	this approach performs jointly face detection, pose estimation, and facial point localization. We evaluated the most accurate (consequently slowest) model.
Baltrusaitis et al. [12]+[175]	the facial point localization [12] is employed on top of the face detector of [175].
Face++[74]	the detection process is conducted on a remote server belonging to the software owner.
Dlib[85]+FAERT[84]	[85] offered the face detector and an implementation of the facial point localization [84].
Luxand [73]	the results were obtained using the Luxand FaceSDK ver. 6.1.
Our approach	face detection time corresponds to the search for faces with all sizes (starting from 24×24 pixels with scale step factor of 1.5) plus the cropping refinement. This time decreases to 0.04 sec. when the search stops on the first detected face.

5.4.5.1 The Number of Iterations

Adding more iterations leads to improve the localization accuracy, but at the cost of more processing effort. Meanwhile, these improvement steps become smaller with each added iteration. Here, I propose performing only five iterations as improvement in the localization accuracy starts to be insignificant to add more iterations. The cumulative proportions of the images that were within a certain average error over the 49 facial points after each iteration are summarized in Figure 5.5. Clearly shown that the localization accuracy is improved after each iteration with steps being shorter after each iteration. Samples of the facial point localization using the proposed approach after three iterations are shown in Figure 5.6. The first row shows the point localization after the first iteration (the initialization stage), the second row shows the localization after three iterations, the last row shows the final results obtained after five iterations.



(a)



(b)

Figure 5.5: Cumulative proportion of the images that are within a certain average error of the 49 facial points (Err_{49avg}), where the error is normalized to face width derived from the datasets annotation, as the minimum width of a rectangle enclosing all the offered facial points (68 points). The figures depict the estimation results obtained after a number of iterations. (a) The results of applying models trained on the Helen and LFPW training sets on the LFPW test set (b) The results of applying models trained on the Helen and LFPW training sets on the Helen test set.

Figure 5.6: Samples of the facial point localization taken from LFPW and Helen testing sets. The first row shows the localization results after the first iteration, the second row after the third iteration, the third row after the fifth iteration.

5.4.5.2 The Number of Selected Features

In this section, I validate the importance of employing a feature selection prior the mapping in each iteration, as proposed by this work. To this end, I conducted an experiment on Helen and LFPW data sets to measure the localization accuracy at the second iteration. At this stage, 6272 ($= 128 \times 49$) features are extracted from the surrounding of facial points estimated in the first iteration. Before mapping them to the points' displacement from their ground truth, I applied the modified CFS method to select subset of the features that are mapped afterward to the displacement via the **MLP**. In a similar way to the evaluation in Sec. 5.4.3, I trained the models using the training set of Helen and LFPW datasets, while evaluating on their test sets separately. The resulting values of the mean of the $\text{Err}_{49\text{avg}}$ across the number of selected features are summarized in Figure 5.7. It is clear to note that, a better generalization capability is achieved by selecting around 1000 features for Helen data set and 500 for LFPW. To minimize the configuration parameters of the proposed approach, I selected a fixed number of features (1000 features) for all iterations and both sets. This specific number represents the average of the optimal numbers obtained by experiments conducted for all iterations and both sets as well. The latter experiments were carried out only using the training sets, where part of them was used for the testing purpose.

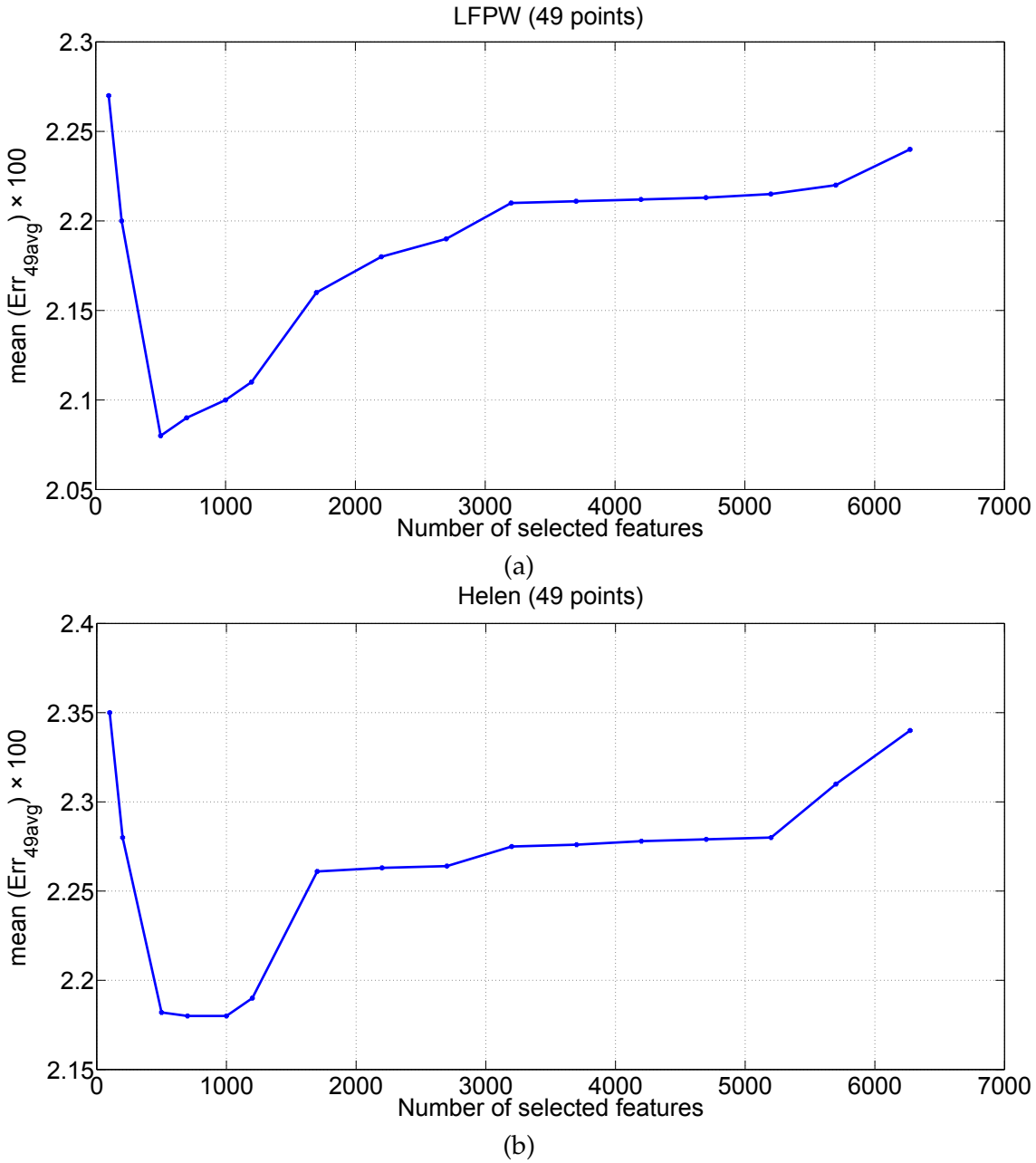


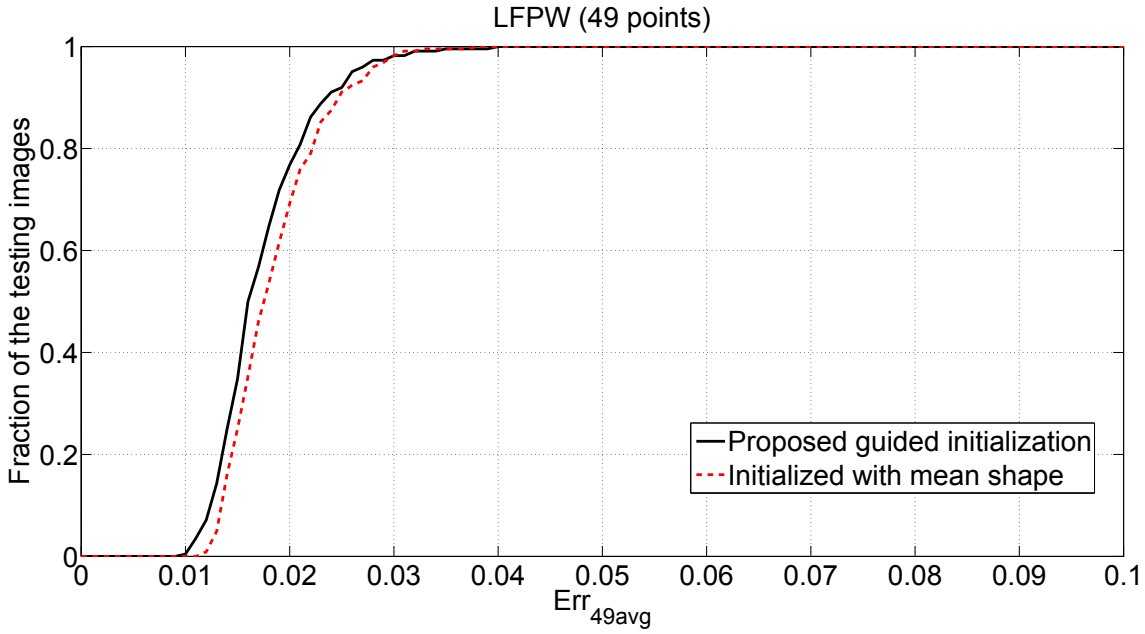
Figure 5.7: The mean of $\text{Err}_{49\text{avg}}$ across the number of selected features measured on the LFPW and Helen test sets at the second iteration.

5.4.5.3 A Guided Initialization

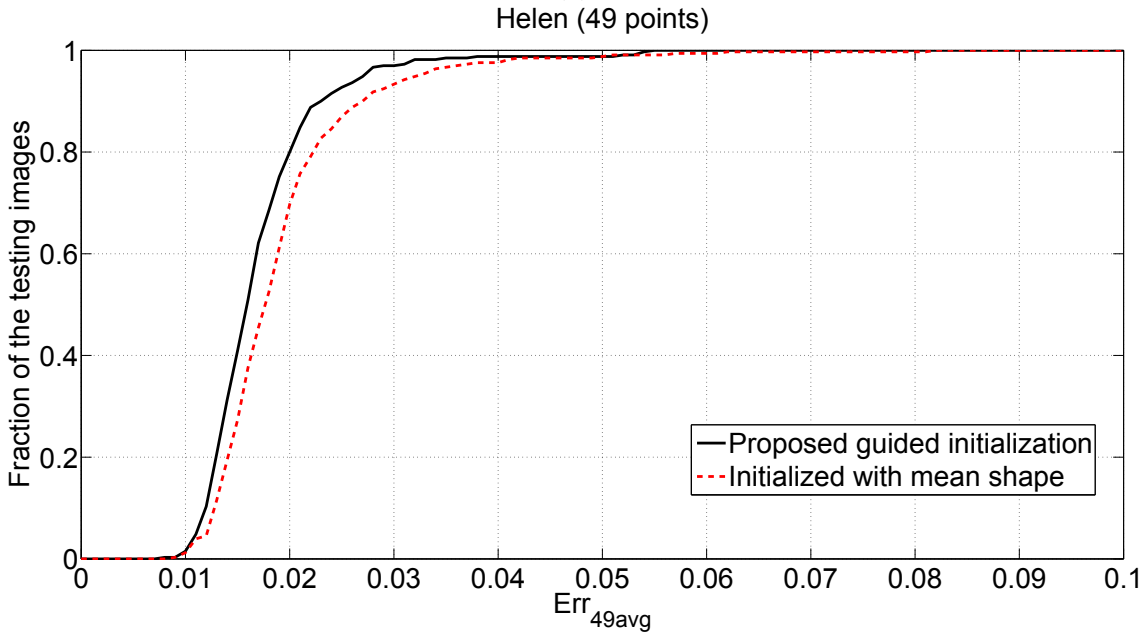
In comparison to the facial point initialization via a random location or the mean location of the training data, the guided initialization procedure, proposed in this work, improves the approach robustness against falling into a non converge situation, especially for the faces of higher pose angles. Figure 5.8 presents the cumulative error using the peoposed approach for two cases: initialization with a mean location computed based on the training sets, and with the suggested guided initialization; taking into consideration that the former initialization does not count as an iteration, hence five iterations in that case follow. The models were trained on Helen and LFPW training sets and tested on their test sets separately. Clearly shown that the enhanced accuracy and generalization capability, i.e., for less than 0.02 of $\mathrm{Err}_{49\mathrm{avg}}$, the facial points were located in 80% of the Helen test set with the guided initialization method, and only in 70% with the ordinary initialization using the mean location.

5.5 Discussion

In this section, I have presented an approach to locate 49 facial points, exploiting the neural networks in a cascade-regression manner. To enhance the performance of the cascade regressors, I performed a guided initialization, in which I mapped holistic-based features (extracted from the entire face patch) to the facial point location, instead of employing the ordinary initializing with a mean shape derived from the training sets. With only four iterations following the initialization in which HoG features were extracted from local patches surrounding the estimated points from the predecessor iteration, I refined the point localization. Additionally, performing a feature selection at each iteration led to a better generalization capability. A cross-validation of a model built using 6500 samples varying across poses, illuminations, expressions, and other factors showed that our approach can detect the facial points with a point average error varies between 0.72% and 1.57% of the face width. Cross-database evaluation and comparison with state-of-the-art approaches confirmed the competitiveness of our results. Moreover, an evaluation according to the 300-w competition, in which all methods were trained and tested with specific sets, demonstrated the better generalization capability of the proposed approach in comparison to three state-of-the-art approaches and two



(a)



(b)

Figure 5.8: Cumulative proportion of the images that are within a certain average error of the 49 facial points (Err_{49avg}) for the proposed approach in two cases: using the guided initialization and using the mean shape.

commercial software packages. From the efficiency point of view, our approach is built on top of the fastest face detector, and is one of the fastest approaches in locating the facial points. The evolution of the localization over the iterations was presented as well. Finally, I conducted two experiments validating the proposed enhancements to the cascade-regression based methods: the guided initialization and employing a feature selection at each iteration.

CHAPTER 6

Head Pose Estimation

Head pose estimation is not only a crucial pre-processing task in applications such as facial expression and face recognition, but also the core task for many others, e.g. gaze; driver focus of attention; head gesture recognitions. I propose here a framework, incorporating different methods, to tackle the pose estimation on a frame basis and in a full automatic scenario starting with a face detection. Instead of the conventional 2D color cameras, the research community is nowadays using current RGBD sensor technology, which provides additionally depth information. By exploiting these depth data many traditional problems such as separating foreground from background pixels, unknown object scales, and some lighting issues can be overcome. Some of those sensors, e.g. Kinect sensor, provide a high-resolution depth-sensing at a consumer price. This camera type was launched worldwide in November 2010, where it was the first time that computer vision played a pivotal role in a mass-market [53].

Based on evaluations conducted on two publicly available RGBD databases, I show throughout this chapter the pose estimation enhancements as a result of employing the depth data either in face spotting, feature extraction, or in both. Additionally, I present two methods to guarantee a consistent face cropping from the VJ face detector, one method is depth-based while the other one is RGB-based. Several appearance-based features besides developed depth-based ones are employed, where a fair comparison between them in terms of estimation accuracy

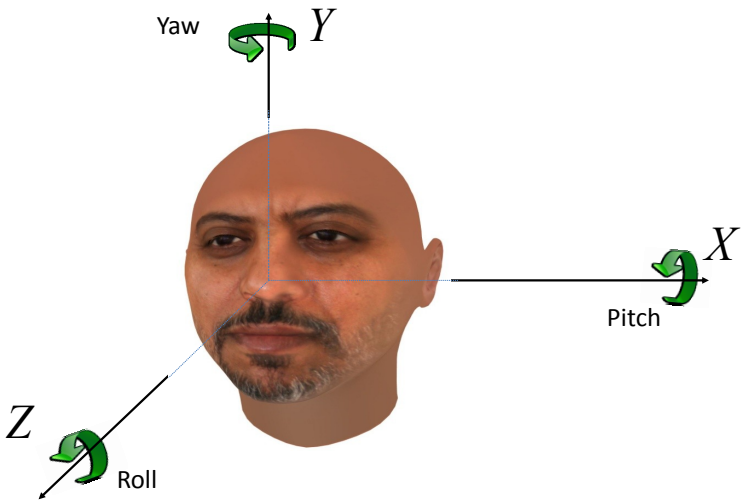


Figure 6.1: The head pose rotation angles. Yaw is the rotation around Y-axis, Pitch around X-axis, Roll around Z-axis.

and computation time is provided.

Throughout this dissertation and complying with many computer vision applications, the face pose estimation is defined as the process of deducing the face orientation from single image/sequence of 2D/3D images. The face is modeled as a rigid object, with 3 DOF in pose characterized by three rotation angles: pitch, roll, and yaw. With a human head facing the camera, yaw is the angle of moving the head left and right (rotation around Y-axis); pitch of moving the head up and down (rotation around X-axis); roll is the tilt angle (rotation around Z-axis), as shown in Figure 6.1.

The structure of the proposed approach is depicted in Figure 6.2. Two images are provided by the RGB-D sensor, the convenient 2D RGB image with its corresponding depth image in which each pixel value represents the distance of the corresponding part of the object to the camera. The two data sources are utilized to locate and crop the face. For this purpose, I developed three methods. One of them is only based on the RGB image, while the other two employ both images. Consequently, the proposed approach is applicable to the 2D cameras as well. As the facial appearance and geometry vary considerably across the head pose, I exploited several texture descriptors, which encode the face appearance

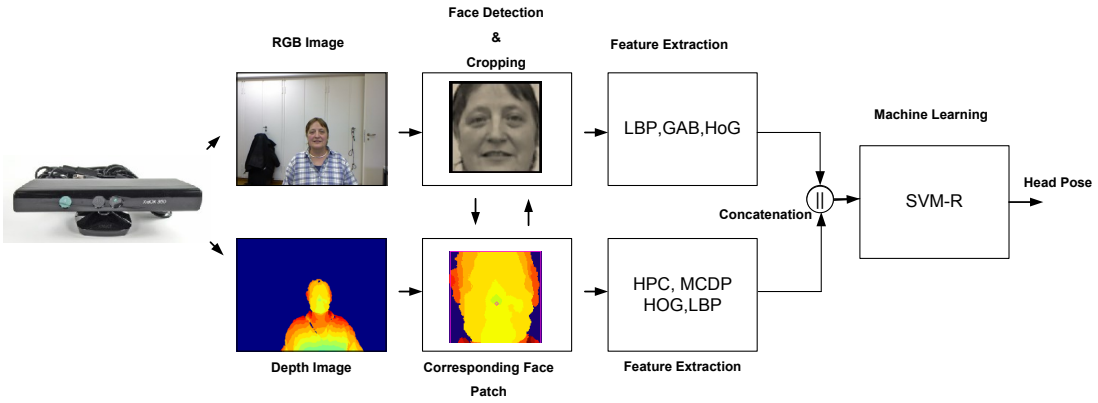


Figure 6.2: The structure of the proposed approach for head pose estimation.

and geometry, to infer the head pose. Those features were extracted from the face patches in both images (RGB and depth). Finally, I formulated the pose estimation as a regression learning process, where one regressor is assigned to each pose angle. Among several supervised learning algorithms, I employed the SVM [28] due to its well-known generalization capability and overfitting avoidance [4]. Each regressor is trained to map the extracted features to the ground truth of the pose angle.

In what follows, I present the developed methods for the face detection and cropping. Then, I describe the exploited feature descriptors. Finally, I report the conducted experiments, including comparisons with state-of-the-art methods, cross-database assessment, and examination of the effectiveness and efficiency of the different feature descriptors.

6.1 Face Detection and Cropping

Due to varying scanning parameters, VJ face detector provides inconsistent face crops, which are inadequate to be direct inputs into any machine learning procedure. A similar problem exists by using fixed search parameters to locate faces of different scales, as explained in Sec. 3.3. In this section, I propose three methods to stabilize the face detector output, two of them are built on top of the VJ face detector.

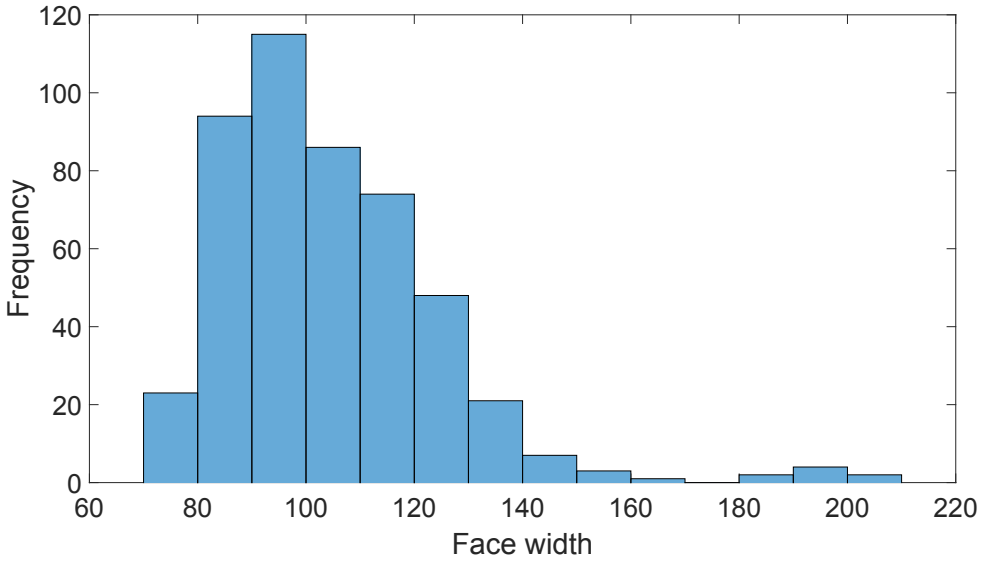
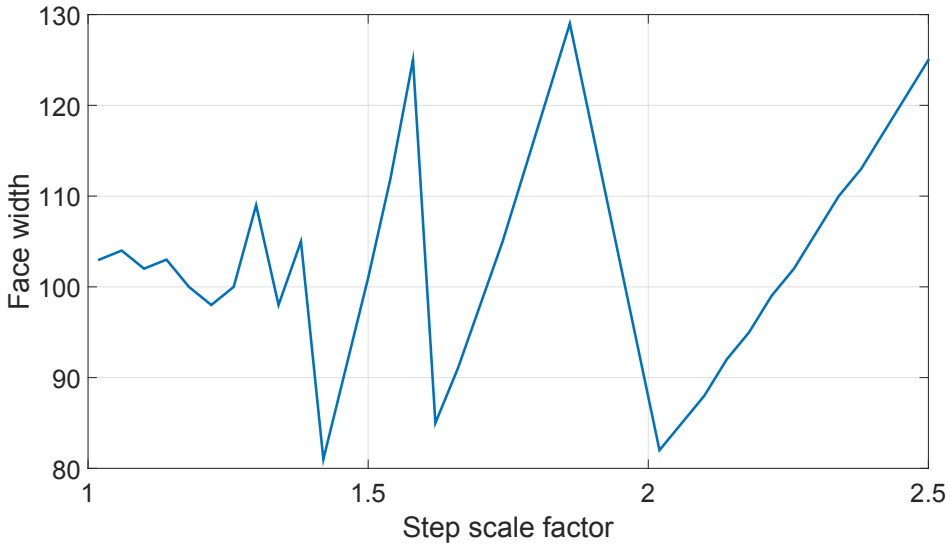


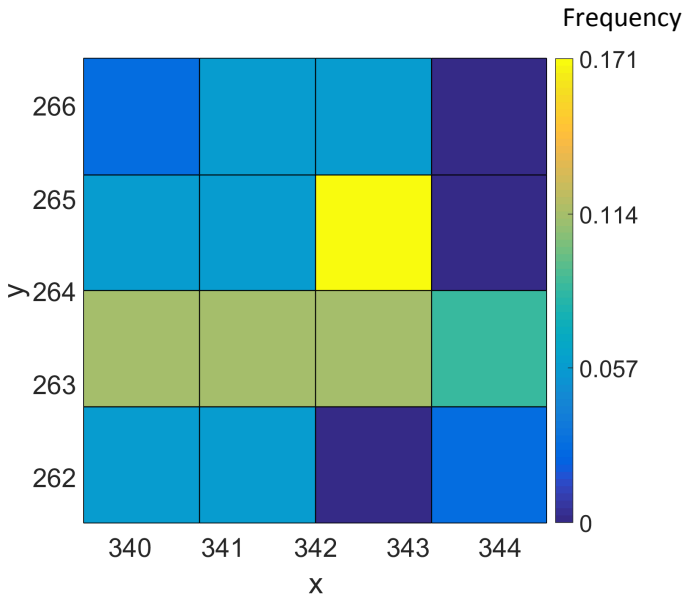
Figure 6.3: A histogram of the width of positive detection windows, stemmed from scanning an image of one face using the VJ approach.

6.1.1 RGB-VJ Face Detection

As the name suggests, this method utilizes only the RGB image and is built on top of the VJ face detector. Figure 3.12 shows samples of face detections stemmed from applying VJ detector on one image but with different searching parameters. A detailed histogram of the positive scanned windows is shown in Figure 6.3. Those results are obtained by scanning the previous image sample for all potential face size with a scale factor of 1.01. These windows share approximately the same center and consequently will be averaged to produce the final detection size. It is clear from the histogram that the average rectangle would be towards the lower values, but this fact does not hold true if you increase the size of searching start window or increase the scale factor seeking faster searching. Figure 6.4a depicts the width of the detected face across the scale factor. Obviously, the detected size varies slightly for lower values of the scale factor and greatly for the higher values. Meanwhile, the variation on the rectangle center is low as Figure 6.4b shows. To cope with this inconsistent cropping, I make the returning rectangle insensitive to the face scale. To this end, I perform two iterations of search. The first search performs a coarse localization where the margin between the minimum and maximum search sizes is bigger and spans all the potential face scales. For efficiency purposes, I use

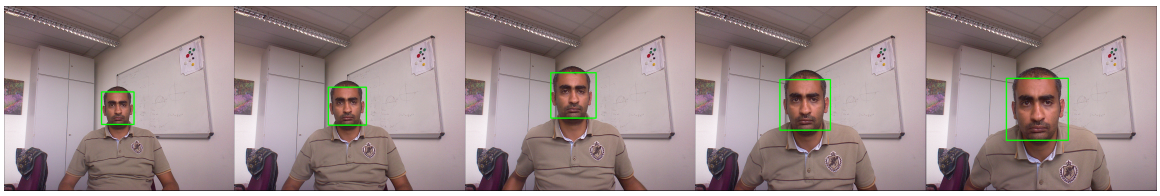


(a)



(b)

Figure 6.4: (a) Detected face width as a function of the search parameter: scale factor. The results are obtained by applying VJ detector to an image of one face each time with different scale factor. (b) A Histogram of the detected face center points, almost sharing the same center



(a) [181 × 181] (b) [211 × 211] (c) [252 × 252] (d) [281 × 281] (e) [344 × 344]

Figure 6.5: Using VJ face detector to perform a two-stage search for the face. The face is consistently cropped in different scales. The size of the returned box is shown beneath each sub image. The images are captured in our lab with a Kinect sensor working at SXGA resolution (1280×1024).

a relatively large step scale factor (1.5) and lower minimum neighboring threshold (3). In the second search, I perform a fine localization, where the minimum and maximum search sizes are taken by 200% and 70% of the face size detected in the first stage, the neighboring threshold is larger (6), and the step scale factor is lower (1.03). The second search is faster since I narrow the search region to the area surrounding the detected face from the coarse localization. The returning box from the fine search is then considered for feature extraction stage. As the final return face box by VJ approach is an averaging of all overlapping detections, by performing the fine search, similar patches are detected invariant to the face scale, this then leads into a consistent cropping. Experimental results of the two-stage search are depicted in Figure 6.5, where the same face is consistently cropped in different scales.

The proposed method here has no effect on the detection rate of the VJ detector, also the false detections in the first search could not be corrected in the second search. Building an RGB-based method to refine the face cropping gains importance from the wide spread of using only 2D images as a source in various applications such as facial expression recognition [37][141] and human age recognition [60].

6.1.2 RGBD-VJ Face Detection

As the name suggests, this method utilizes both depth and RGB images and is built on top of the VJ face detector. The face is located in the grayscale image (deduced from the RGB image) using the VJ approach, while the search parameters are set with the help of the depth image. I employ the frontal model to the images of

absolute yaw angles lower than 30° and the profile model to the other images. Ensuring the consistent face cropping, I narrow the searching window sizes and use a fixed scale factor. To this end, I exploit the depth information along with the intrinsic parameters of the Kinect camera. Let h_n denote the head width in nature measured in millimeters (mm). The expected head width in the image h_m in pixels could be approximately calculated by

$$h_m = \frac{f_x \times h_n}{h_z}, \tag{6.1}$$

where f_x denotes the camera focal length multiplied by the scale parameter in x-direction, and is measured in pixels. h_z is the face distance to the camera measured in mm. Next, the face is searched with square windows of side length varying between $0.75 \times h_m$ and $1.05 \times h_m$, where h_n is fixed at 200 mm, the scale factor is 1.05. Besides stabilizing the detector output in terms of the returned width, many false-positive detections would be avoided with the proposed setup.

6.1.3 RGBD-GMM Face Detection

Building a head pose estimator on top of an automatic face detector is necessary for real-world applications. The main shortages of most available face detectors are their inconsistent face cropping across scales and their limitation to a small range of poses. The VJ face detector work as well within a limited pose range; however, both frontal and profile models are utilized.

In this section, I propose a method to spot the face in the D-RGB images. To this end, I built a Gaussian Mixture Model (GMM) for the face under pose variations. Exploiting the BIWI database [52], I divided the entire pose range into discrete groups spaced by 5 degrees in each angle (pitch, yaw, roll) direction. Then, for each cube I selected one sample from each subject, if available. Next, I annotated those samples by enclosing each face sample with a box of fixed size in real-world units. Annotated samples of three subjects are shown in Figure 6.6. I empirically set the head width (h_n) in the real-world units to 150 mm. Then using a simple pinhole camera model, I calculated the corresponding head width (h_m) in pixels as in Eq. (6.1), and aligned the faces inside the face box as shown in Figure 6.6. Next the faces were cropped, each with the corresponding h_m. Features extracted from those patches were used to build a multivariate GMM face model, where the

Figure 6.6: Samples of our annotations on three subjects, taken from the BIWI database, at different poses.

likelihood of a face feature vector is calculated as follows.

$$p(\mathbf{x}|\Phi) = \sum_{i=1}^{m} \alpha_i p_i(\mathbf{x}|\phi_i), \tag{6.2}$$

where $\mathbf{x} = (x_1, ..., x_d)' \in \mathbb{R}^{d\times 1}$ is a feature vector encoding the face patch. $\phi_i = (\mu_i, \Sigma_i)$, $\Phi = (\alpha_1, ..., \alpha_m, \phi_1, ..., \phi_m)$ is the face model, which is estimated via the Expectation Maximization (EM) algorithm. Each p_i is a d-dimensional multivariate Gaussian distribution given by

$$\begin{aligned} p_i(\mathbf{x}|\phi_i) &= \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma_i|^{1/2}} \\ &\quad \times \exp\{-\frac{1}{2}(\mathbf{x}-\mu_i)^{\mathrm{T}}\Sigma_i^{-1}(\mathbf{x}-\mu_i)\}, \end{aligned} \tag{6.3}$$

$\mu_i \in \mathbb{R}^{d\times 1}$ is the mean vector of the i^{th} subpopulation; where Σ_i is its $d \times d$ covariance matrix. $\alpha_i \in [0, 1]$ for all i and the α_i's are constrained to sum to one. To spot a face inside an image, I evaluate all potential face locations; the window of $\mathbf{x}^*$ of

maximum $p(\mathbf{x}|\Phi)$ is considered the cropped face.

$$\mathbf{x}^* = \arg\max_{\mathbf{x}} p(\mathbf{x}|\Phi). \tag{6.4}$$

The patch of $\mathbf{x}^*$ is then used for the pose estimation process. A satisfactory correct localization rate was achieved with a face model of five subpopulations. The face patch is scaled to 100×100 pixels, with cell size of 20×20 pixels, block of 40×40 pixels, block spacing stride of 20 pixels and eight bins orientation histogram. Therefore, the final HoG descriptor has a length of 512. I evaluated the proposed method on the BIWI database with features extracted from each source, and from both. Accordingly, I achieved face localization rates of 98%, 80%, and 88% utilizing features from the grayscale image (derived from the RGB image) HoG_g, depth image HoG_d, and both HoG_{g+d}, respectively. The face localization is considered correct if the intersection area of the predicted patch and its corresponding ground truth patch is at least 60% of the union of the two. The localization rate using HoG_g is better than that using HoG_d due to the distinctive unique texture of the face in the grayscale image. Additionally, the search with only one scale precludes many false-positive detections. Although the depth-based features show better performance on estimating the head pose [52], they perform poorly in the face localization, where the face pattern is often confused with parts from the body as they look like a face in profile views. The HoG_d vector adversely affects the localization rate when it is concatenated with HoG_g. Consequently for further processes, only HoG_g is employed as a feature vector in Eq. (6.2), while the patch size is determined based on its depth data. With a goal of maximizing the face localization rate, I set the parameters for the aforementioned process (the number of Gaussian subpopulations, HoG parameters, h_n) through a grid search with cross-validation evaluation conducted on the BIWI database.

6.1.4 Discussion

Three methods for locating the human face with a consistent cropping have been proposed in this Section. As the VJ face detector has been being employed in various computer vision applications besides the availability of an optimized fast code of it, the first two methods were built on top of it stabilizing its output. In other words, the two methods make the VJ output perspective invariant to the face scale. The first method (RGB-VJ face detection) was developed on the RGB basis making

Table 6.1: Face localization rates (%) resulting from cross-database evaluation using ICT-3DHP database. The DRGB-GMM method and the approach of [52] were completely developed based on BIWI database, while DRGB-VJ and RGB-VJ methods employed the frontal and profile models of VJ detector whose parameters were set with respect to BIWI database.

Approach	**Localization rate** %
DRGB-VJ and RGB-VJ	85
Fanelli et al. [52]	82
DRGB-GMM	**95**

it applicable to 2D cameras as well. Using the depth in the second and third methods removes a lot of false-positive detections whose real size does not match a potential face. For a wider range of poses, the third method was designed, while the second is limited to the range of the VJ face model.

I conducted a cross-database evaluation to assess the generalization capability of the aforementioned methods for the face localization. The three methods were trained/optimized using BIWI database and then evaluated on the ICT-3DHP. I also applied the models of [52] that were trained using BIWI database as well. Table 6.1 summarizes the localization rates stemmed from this cross-database evaluation. I achieved a localization rate of 95% using RGBD-GMM method compared to 82% of [52], 85% of RGBD-VJ and RGB-VJ methods, an improvement by at least 10% proves the better generalization capability of RGBD-GMM method for the face localization. Both frontal and profile models were exploited for the face detection using the VJ method.

6.2 Feature Extraction

As the facial geometry and appearance vary significantly across the head pose when the face is captured using a fix mounted camera, exploiting mixed feature types that encode those characteristics is the way to infer the head pose. Accordingly, I extract several feature types from the face two patches in depth and RGB images. Those features can be divided into two groups based on their data source as follows.

6.2.1 RGB-based Features

I employed three texture-based descriptors to encode the face appearance. These descriptors were applied to a gray-scale image, derived from the face patch in the RGB image.

6.2.1.1 Gabor Filter-based Features

A brief introduction of this descriptor is given in Sec. 3.2.1. To describe the face patch here, I generate a Gabor filter bank consisting of two frequencies (λ), two standard deviation values ($\sigma_x = \sigma_y$), 7 rotation angle values (θ). After applying each kernel to the face patch of 100×100 pixels, I divide the resulting patch into smaller 10×10 pixel cells. Next, I represent each cell by its median value and then normalize these values to generate a kernel feature vector. Finally, I concatenate the vectors from all kernels to produce the GAB feature vector of length 2800.

6.2.1.2 Local Binary Pattern Features

A brief introduction of this descriptor is given in Sec. 3.2.2. To tackle the pose estimation, I calculate the LBP_v (as in Eq. (3.33)) for each pixel of the scaled 200×200 pixels face patch. Then, I divide the face patch into smaller 10×10 pixels cells. Next, I calculate an 8 bin histogram for each cell. Finally, those histograms are concatenated to form the LBP feature vector of length 3200.

6.2.1.3 Histograms of Oriented Gradient Features

A brief introduction of this descriptor is given in Sec. 3.2.3. Here, I scale the face patch to 200×200 pixels before dividing it into smaller cells of 20×20 pixels. The x- and y- gradients of each pixel are then estimated with the help of horizontal (Gx) and vertical (Gy) Sobel kernels. Next, I calculate an 8 bin orientation histogram for each 40×40 pixels block region, where each block region comprises 4 cells. A block spacing stride of 20 pixels is used. The final feature vector is of length 2592.

6.2.2 Depth-based Features

As the name suggests, those features are extracted from the face patch in the depth image encoding the face shape. The observed 3D face shape varies across the head

poses, which makes those features valuable for the pose estimation. Besides extracting the aforementioned descriptors (GAB, LBP, HoG) from the depth patch as well, in what follows I propose new depth-based features.

6.2.3 Head Point Cloud Features (HPC)

This feature type encodes roughly the orientation of the point cloud of the considered face, which is equivalent to the head pose in some cases. First, I retrieve the corresponding point cloud to the face patch (**HP**) in the depth image. Preventing the inclusion of background objected, I allow only a certain depth range (**DR**). Here, I include only the points that are not far by more than $50\ mm$ from the closest head point to the camera. Using a simple pinhole camera model, for a pixel (x_i, y_i), I get its corresponding 3D point $\mathbf{p}_i$ as follows.

$$\mathbf{p}_i = \begin{pmatrix} \frac{-z_i(x_i - c_x)}{f_x} \\ \frac{-z_i(y_i - c_y)}{f_y} \\ z_i \end{pmatrix}, \qquad (x_i, y_i) \in \mathbf{HP}, z_i \leq \mathbf{DR}. \tag{6.5}$$

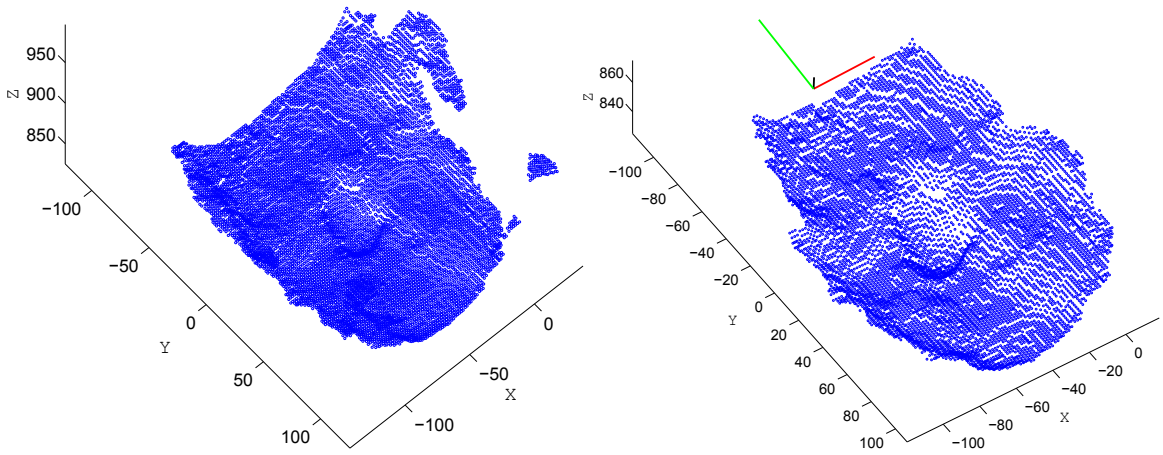
z_i denotes the corresponding depth value for the pixel (x_i, y_i). (c_x, c_y) is the principal point, usually at the image center. Here the image distortions are ignored due to their negligible effect using the Kinect sensor. f_x, f_y are the focal lengths multiplied by the scale parameter in x- and y- directions, respectively, expressed in pixels. For each depth patch, I consider the 3D point $\mathbf{p}$ as a random variable with n samples satisfying Eq. (6.5). Next, I calculate its covariance matrix as follows.

$$\Sigma = \mathrm{E}\left[(\mathbf{p} - \mathrm{E}[\mathbf{p}])\,(\mathbf{p} - \mathrm{E}[\mathbf{p}])^{\mathrm{T}}\right], \tag{6.6}$$

where $\mathrm{E}(x)$ is the expected value (or mean) of x. $[\,]^{\mathrm{T}}$ denotes the matrix/vector conjugate transpose. Following this, I apply the Singular Value Decomposition (SVD) to the covariance matrix Σ, obtained by Eq. (6.6), to be written as follows.

$$\Sigma = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}}. \tag{6.7}$$

$\mathbf{U}, \mathbf{S}, \mathbf{V}$ are matrices of size 3×3. $\mathbf{U}$ and $\mathbf{V}$ are unitary matrices. $\mathbf{S}$ is a diagonal matrix with diagonal entries $(\sqrt{\tilde{\lambda}_i})$ equal to the square root of eigenvalues from $\Sigma\Sigma^{\mathrm{T}}$. And the eigenvectors of $\Sigma\Sigma^{\mathrm{T}}$ make up the columns of $\mathbf{U}$, while the eigenvectors of $\Sigma^{\mathrm{T}}\Sigma$ make up the columns of $\mathbf{V}$. The eigenvectors describe the orthogonal principal directions of the variations in the point location, with corresponding standard



(a) (b)

Figure 6.7: Extracting the Head point cloud features (HPC). (a) shows the retrieved 3D points from the depth patch of the located face. (b) The filtered points by Eq. 6.2 and the eigenvector direction shown on the top of the sub-image. X, Y, Z represent the real coordinates in mm.

deviation $(\sqrt{\tilde{\lambda}_i})$. Finally, I concatenate the eigenvectors and use them as a feature vector of length 9 encoding the point cloud general orientation. The process of extracting this feature type is summarized in Figure 6.7, where the whole 3D points of the captured face are depicted in Figure 6.7a, and the filtered points (according to Eq. (6.2)) along with the extracted eigenvectors in Figure 6.7b.

6.2.4 Multi-scale Comparative Depth Patches (MCDP)

Here, I extract straightforward features describing the spatial distribution of the points depth within the face patch. The face patch is divided into smaller equally sized cells, four times each with different cell scale. Then, for each scale, I represent each cell by the average value of its points' depth. Next, I normalize the cell values to produce a scale vector. Finally, I concatenate feature vectors stemmed from the four scales to produce a final MCDP descriptor vector of length 512.

6.2.5 Machine Learning Approach

To produce a continuous estimate of the head pose, I exploit a regression-based machine learning approach rather than a discrete classification-based one. Consequently, our approach can be used as a base for a head gesture recognition system. I employ three nonlinear SVM regressors, each for one angle, to map the extracted features to the corresponding pose angle.
The parameters of each regressor along with those of the feature extraction were chosen using grid search with cross-validation experiments conducted on the training set with a goal of achieving an accurate estimation with a reasonable resources and processing time.

6.3 Experimental Results

Many experiments were conducted to evaluate the proposed method for head pose estimation in both within-database and cross-database scenarios. To this end, I employed two databases, publicly available. Comparisons among several texture and geometry- based feature types were performed in terms of accuracy and effectiveness. I studied the detection capability of VJ models under a wider range of poses. Finally, I boosted the pose estimation accuracy by exploiting the depth image, either in feature extraction or in face localization.

6.3.1 Analysis using the Frontal Model of VJ Detector

To investigate the pose range of the frontal model of VJ detector, I applied it to the entire BIWI database (15,678 images). The face was successfully located in approximately 75% of the entire database, those face patches participated in the following evaluation. The scanning results were summarized in Figure 6.8. Figure 6.8a shows the grid detection rate across the pitch and yaw rotation angles, where the detection rate measures the percentage of the images in which the human face was detected to the entire images of the underlying grid. To complement the results in Figure 6.8a, I provide the total number of the images across the yaw and pitch angles in Figure 6.8b. In a similar way, the results are depicted for pitch-roll axes in Figures 6.8c and 6.8d. Obviously seen that the lower pose angles ($\pm 15^\circ$ pitch, $\pm 15^\circ$ yaw, $\pm 10^\circ$ roll) are represented in the database with more samples, other grids comprise 50 samples at the minimum. It has been proven that frontal model

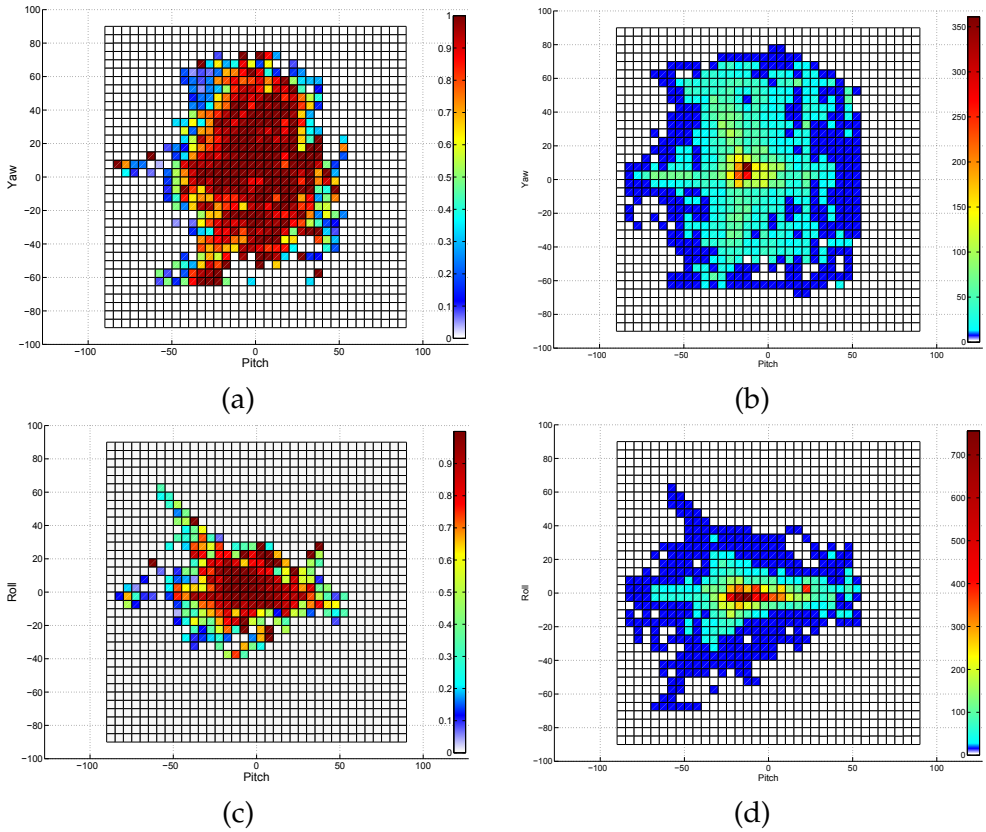


Figure 6.8: The results of applying the frontal model of VJ face detector on the entire BIWI database, showing the pose range of model (a) is showing the detection rate across yaw and pitch angles in degree. (b) is complementing (a) by showing the number of samples for each yaw-pitch grid. (c) is showing the detection rate across roll and pitch angles. (d) is complementing (c) by showing the number of samples for each roll-pitch grid.

of VJ approach is capable of detecting faces of poses spanning $\pm 30^{\circ}$ pitch, $\pm 20^{\circ}$ roll, $\pm 40^{\circ}$ yaw, with 80% detection rate at the minimum. For example, at lower pitch angle values, the faces of yaw angle spanning $\pm 60^{\circ}$ can be detected with 90% detection rate. Meanwhile, the poses beyond those ranges could be detected but with lower detection rates.

Next, I performed a comparison in terms of accuracy among separate/combination of the aforementioned feature descriptors, where the face was located using the RGB-VJ method with only the frontal face model. The error is defined as the absolute value of the difference between the ground truth angle and the predicted one. For each experiment, I reported the mean and standard deviation of the estimation error for each rotation angle. In a similar way to Fanelli et al. [52], I

Table 6.2: The mean/standard deviation of the absolute error for each estimated head pose angle. Feature column indicates the used single feature type or concatenation of more than one. This experiment was carried out on BIWI Database. The subscript d ($-_{\mathrm{d}}$) is added to indicate that the data source here is the depth patch, and ($-_{\mathrm{g}}$) from the gray-scale image of the RGB image.

Feature	**Pitch error** $(^\circ)$	**Yaw error** $(^\circ)$	**Roll error** $(^\circ)$
$\mathrm{LBP_d}$	8.9 / 8.5	8.8 / 8.9	4.8 / 5.9
$\mathrm{LBP_g}$	12.4 / 10.5	12.6 / 13.5	4.7 / 5.3
GAB	9.8 / 8.5	7.6 / 7.5	4.4 / 4.6
$\mathrm{HOG_g}$	6.9 / 6.8	6.3 / 7.7	3.1 / 4.2
$\mathrm{HOG_d}$	4.9 / 5.8	6.1 /6.8	3.8 / 4.7
$\mathrm{HOG_g + HOG_d}$	4.3 / 5.5	5.5 / 6.5	3.0 / 4.3
$\mathrm{LBP_g + LBP_d}$	8.6 / 8.7	8.8 /8.9	4.6 / 5.8
$\mathrm{HPC + MCDP}$	5.3 / 5.4	5.6 /5.4	4.3 / 4.8
$\mathrm{HOG_g + HOG_d + HPC + MCDP}$	**4.0 / 5.1**	**4.3 /5.4**	**2.9 / 4.2**

divided the database into training and test sets of 18 and 2 subjects (leave-two-out cross validation), respectively. Samples of the same person do not exist in both training and testing sets. Table 6.2 summarizes the obtained results. A number of conclusions can be drawn from the depicted results. Employing the same feature type from the depth image leads to more accurate estimates than from the RGB image. For example, for the pitch angle I achieved a mean error of 8.9° with $\mathrm{LBP_d}$ compared with 12.4° with $\mathrm{LBP_g}$, and 8.8° versus 12.6° for the yaw angle, whereas the estimation mean errors for the roll angle are equal using both $\mathrm{LBP_d}$ and $\mathrm{LBP_g}$. In a similar way, $\mathrm{HOG_d}$ provides more accurate estimation compared with $\mathrm{HOG_g}$ for both pitch and yaw angles, while slightly less accurate for roll angle. Regarding the appearance-based features, the most accurate pose estimation is achieved by $\mathrm{HOG_g}$, where the pose mean errors are $6.9^\circ, 6.3^\circ, 3.1^\circ$ for pitch, yaw, roll, respectively, compared to $9.8^\circ, 7.6^\circ, 4.4^\circ$ achieved by GAB and even greater errors ($12.4^\circ, 12.6^\circ, 4.7^\circ$) by $\mathrm{LBP_g}$. Regarding the depth-based features, the $\mathrm{HOG_d}$ provides more accurate pose estimations, which are slightly better than those using the newly introduced descriptors: $\mathrm{HPC + MCDP}$. On the other side, I got the biggest error in the estimated pose angles with $\mathrm{LBP_d}$. Concatenating depth-based

(a)
GT[5.3 61.7 17.5]
PR[-15.2 30.7 -10.7]

(b)
GT[5.3 61.7 17.5]
PR[no detection]

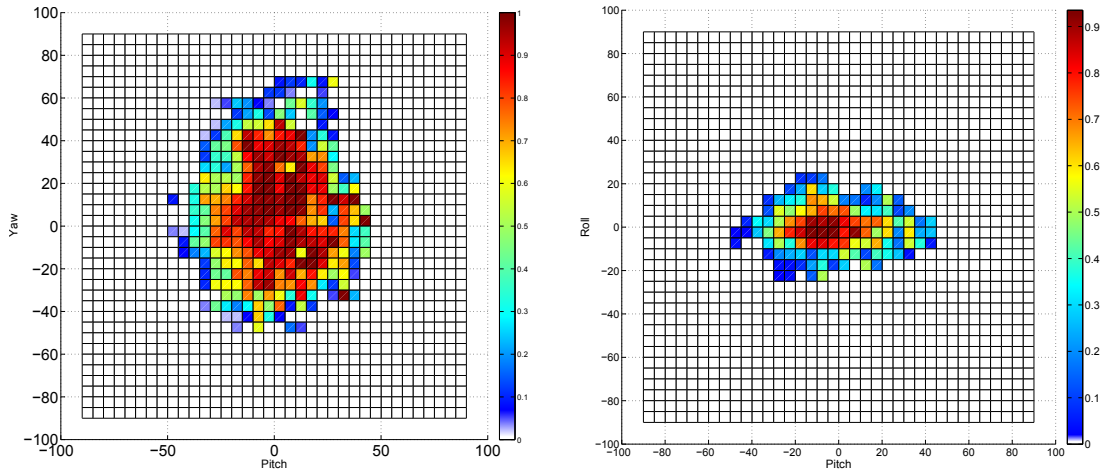
Figure 6.9: Sample of inconsistent face cropping and detecting due to the background texture. (a) wrong face cropping using frontal model. (b) the face with whited background, not detected using the frontal model. GT denotes the ground truth rotation angles [Pitch Yaw Roll] and PR is the estimated angles.

and appearance-based features leads to an improvement in the estimation accuracy, as it can be noticed $\mathrm{HOG_d} + \mathrm{HOG_g}$ performed better than using them individually; and in a similar way, $\mathrm{LBP_g} + \mathrm{LBP_d}$ performed better than using them separately. Our most accurate estimations were obtained by concatenating the RGB-based feature $\mathrm{HOG_g}$ with $\mathrm{HOG_d} + \mathrm{HPC} + \mathrm{MCDP}$ depth-based features, where I estimate the pitch angle with $4.0°$ as a mean error, yaw with $4.3°$, roll with $2.9°$. An insignificant improvement by concatenating all the features type was not considered since it costs unreasonable computation time and resources.

6.3.2 Analysis using the Frontal and Profile Models of VJ Detector with Background Removal

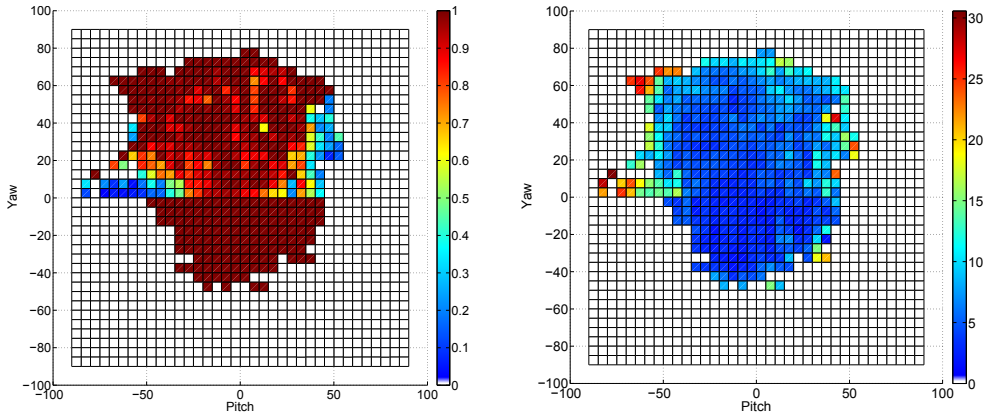
The frontal model of VJ detects profile faces as frontal ones when a complementary background exists, leading to a larger pose estimation error. Figure 6.9 highlights this issue; our approach predicted a pose that is far from the ground truth. I avoid these detections by assuming a uniform background. To this end, I whited the background out exploiting the depth data, as shown in Figure 6.9b. Accordingly, I updated Figures 6.8a and 6.8c to be 6.10. Obviously, the detection rate using the frontal model for faces in profile poses (faces with higher yaw angle) is decreased, which helps avoiding the most noisy estimations.

To cover a wider range of head poses, I exploited both frontal and profile models



(a) (b)

Figure 6.10: The results of applying frontal model of VJ face detector on the BIWI database with whited background. (a) is showing the detection rate across yaw and pitch angles in degree. (b) is showing the detection rate across roll and pitch angles.



(a) (b)

Figure 6.11: The results of applying frontal and profile models of VJ face detector on the BIWI database after whitening the frames background. (a) is showing the detection rate across yaw and pitch angles in degree. (b) is showing the average error of the estimated angles across yaw and pitch angles.

(a) GT[1.3 33.2 -1.9] PR[-4.5 26.9 -0.7]
(b) GT[-34.2 36.7 2.7] PR[-33.7 34.9 -1.1]
(c) GT[5.3 61.7 17.4] PR[-6.3 65.0 1.7]
(d) GT[-21.1 -18.4 5.4] PR[-22.4 -17.4 3.8]
(e) GT[51.0 5.45 -2.46] PR[No detection]

Figure 6.12: Samples of head pose estimations taken from BIWI Database, where a concatenation of $\mathrm{HOG} + \mathrm{HOG_d} + \mathrm{HPC} + \mathrm{MCDP}$ feature types is employed. GT denotes the ground truth rotation angle [Pitch Yaw Roll] and PR is the estimated angle. The face is located by the RGB-VJ method on frames with whitened background.

of VJ face detector. The profile model was applied once the frontal model fails to return a true positive detection. In particular, I employed the RGB-based face localization method on the frames with a whitened background. The detection rates using both models across yaw and pitch angles are summarized in Figure 6.11a. Obviously, employing the profile model results in higher detection rates, reaching more than 95% in most grid cells, for the faces at significant yaw angles. Meanwhile, faces with extreme pitch angles are still hard to detect, as shown in Figure 6.12e. Table 6.3 summarizes the cross validation using BIWI database of several concatenations. A number of points can be drawn from the obtained results as follows. The pose estimation with features from depth image is more accurate than that from the gray image; however, this advances in the accuracy are less than that in Table 6.2 as the depth data already affect the gray image by whiting the background out. Similarly to the results in Table 6.2, concatenating both feature types leads to more accurate estimates. The pose estimation using HOG features (or any concatenation involving it) is more accurate in comparison to others. The features of $\mathrm{HPC} + \mathrm{MCDP}$ provide competitive estimation accuracy as well. The concatenation of $\mathrm{HOG_g} + \mathrm{HOG_d} + \mathrm{HPC} + \mathrm{MCDP}$ achieves the most accurate estimates, where the average errors are not exceeding $5.1°, 4.6°, 4.2°$ for pitch, yaw, and roll respectively. The mean error of the estimated three angles, resulting from the use of $\mathrm{HOG_g} + \mathrm{HOG_d} + \mathrm{HPC} + \mathrm{MCDP}$ concatenation, is depicted in Figure 6.11b. Interestingly, the estimation is accurate for high yaw angles as for the low ones, resulting from the employment of the VJ profile model. On the other hand, the estimation error is increasing as the pitch angle gets high, which in

Table 6.3: Pose estimation results stemmed from Cross-validation experiments conducted on the BIWI database using several concatenations from the feature types. The mean and standard deviation of the absolute error for each estimated head pose angle are presented. Here, I employed RGB-VJ localization method, both frontal and profile models, on frames with background removal. The subscript d ($-_{\mathrm{d}}$) is added to indicate that the data source here is the depth patch, and ($-_{\mathrm{g}}$) from the gray-scale image of the RGB image.

Algorithm	**Pitch error** $(^\circ)$	**Yaw error** $(^\circ)$	**Roll error** $(^\circ)$
$\mathrm{LBP_d}$	10.9 / 9.8	9.1 / 7.8	6.4 / 6.1
$\mathrm{LBP_g}$	11.3 / 10.5	9.7 / 8.5	7.0 /6.7
GAB	11.2 / 10.9	8.5 / 7.1	6.1 / 6.2
$\mathrm{HOG_g}$	7.4 / 7.2	6.1 / 5.7	4.5 / 4.9
$\mathrm{HOG_d}$	7.1 / 8.2	5.9 /6.3	5.3 / 5.7
$\mathrm{LBP_g + LBP_d}$	8.9 / 9.1	7.9 /7.2	6.3/ 5.9
$\mathrm{HOG_g + HOG_d}$	6.3 / 6.1	5.4 / 5.3	4.3 / 4.2
$\mathrm{HPC + MCDP}$	7.6 / 7.4	6.6 /6.1	4.9 / 4.8
$\mathrm{HOG_g + HOG_d + HPC + MCDP}$	**5.12 / 5.3**	**4.6 /4.5**	**4.2 / 4.1**

most cases due to false cropping. Samples of our cross-validation evaluation using $\mathrm{HOG_g + HOG_d + HPC + MCDP}$ on the BIWI database are depicted in Figure 6.12; 6.12a, and 6.12b are samples of the frontal model detection while 6.12c and 6.12d of the profile model. The face in 6.12e cannot be detected by both models.

6.3.3 Boosted Head Pose Estimation via RGBD-based Localization

In the previous experiment, the whited background simulates a uniform background, while the face was located via the RGB-VJ method. The face localization can be enhanced using the depth information, as explained in RGBD-VJ and RGBD-GMM methods. Here, I highlight the corresponding improvement in the pose estimation. In this section, I consider only $\mathrm{HoG_g}$ and HoG_d features. Table 6.4 summarizes our results in comparison with those of state-of-the-art approaches for the within-database evaluation. This evaluation was conducted on the BIWI database. The participation in the comparison was limited to the approaches that incorporate an automatic face localization and work in a frame-based mode as well, except the approach in [117] which is built on top of manual annotations.

Table 6.4: The mean/standard deviation of the absolute error for each head pose angle. Within-Biwi database evaluation.

Approach	Pitch Er °	Yaw Er °	Roll Er °
Fanelli et al. [52]	8.5 / 9.9	8.9 / 13.0	7.9 / 8.3
Yang et al. [172]	9.12 / 7.40	8.92 / 8.27	7.42 / 4.9
Mukherjee and Robertson [117]	4.76/-	5.32/-	-/-
Our using RGB-VJ	6.3 / 6.1	5.4 / 5.3	4.3 / 4.2
Our using RGBD-VJ	5.0 / 5.8	3.9 / 4.2	4.3 / 4.6
Our using RGBD-GMM	**4.19 / 4.30**	**3.84 / 3.9**	**4.13 / 4.4**

Yang et al. [172] built his own face detector that exploits both RGB and depth information. Fanelli et al. [52] built a face detector based on the depth data only. The aforementioned methods annotate the face by a box exactly enclosing it, while in the RGBD-GMM method the box size is depth-based calculated. I achieved a competitive results by employing RGBD-VJ method. Meanwhile, our approach using RGBD-GMM method provided the best estimation accuracy, where the major improvement was achieved for the pitch angle by at least 11.9% ($(4.19 - 4.76)/4.76$). This improvement can be attributed to the employed method for the face localization and cropping. The depth-based cropping is considered as a compromised solution to arrange the faces with a goal of minimizing intra-class variation (of faces with similar poses) while maximizing inter-class variation (of faces with different poses). Clearly shown in Figure 6.6 that within a box of fixed real-world size, the variations of the face texture and 3D structure across the head poses are greater than those among individuals of the same pose. Although the size of the projected (observed) face changes across the pitch angles, the cropping size stills fixed leading to more accurate pitch estimates. The estimated error of the yaw angle across the yaw-pitch angles using our method of RGBD-VJ is depicted in Figure 6.13. Obviously shown that the estimation is less accurate for higher yaw values compared to the lower ones, resulting from less appearance variation at the higher rotation angles. Figure 6.14 shows pose estimates over an image sequence stemmed from our approach of RGBD-GMM. The estimation error increases when the head is under higher pose angles as well, which is reasonable as the appearance variations are barely visible at those rotation values. Interestingly, it is apparent that our approach is qualified to be used in applications of head gesture recognition.

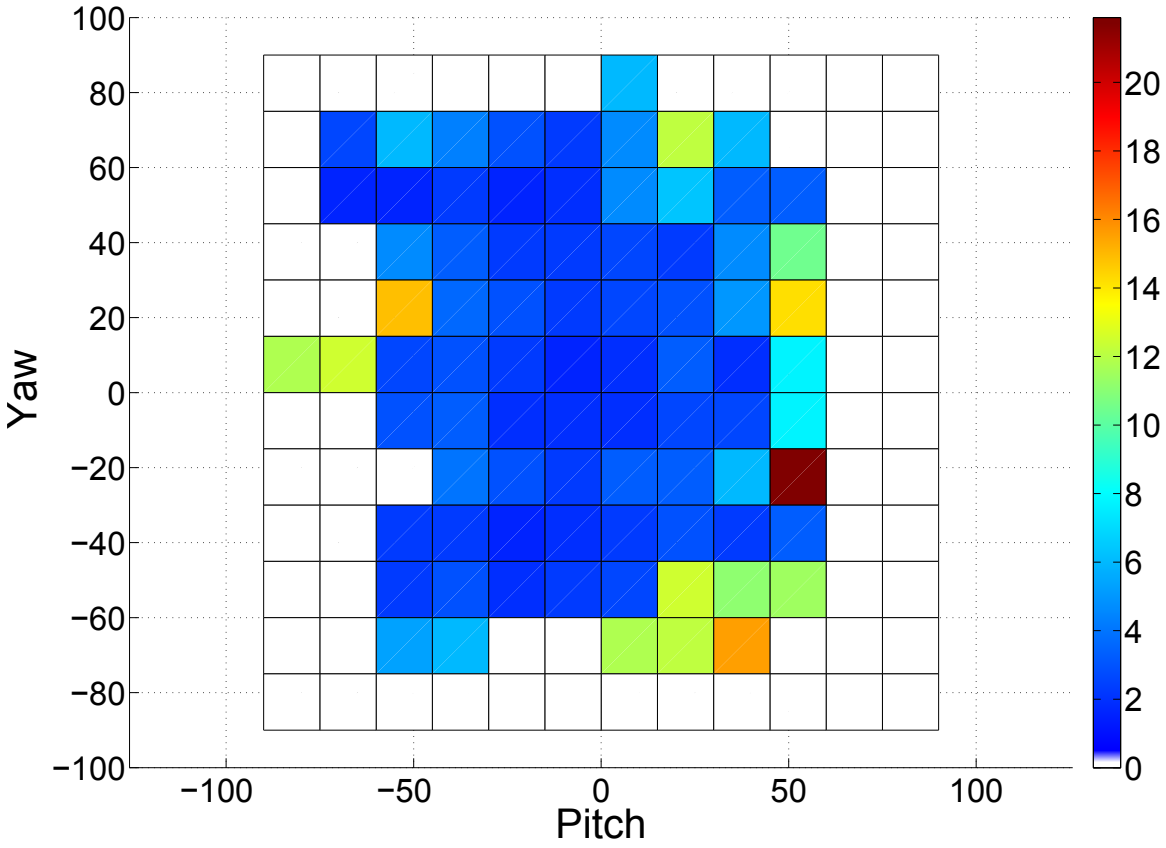


Figure 6.13: The distribution of mean absolute error of the estimated yaw angle over yaw-pitch angles. This experiment was carried out on BIWI Database using our approach of DRGB-VJ, where a concatenation of HoG_g+HOG_d is employed. The complete white grid denotes that no samples at this grid participated in the evaluations.

I performed an additional cross-database evaluation to assess the generalization capability of our methods. To this end, I evaluated a pre-trained head pose models on the ICT-3DHP database. Those models were trained using the BIWI database. The annotations of the ICT-3DHP database are actually the angles difference from the first frame; consequently, I considered the estimation of the first frame as a bias. Table 6.5 shows the pose estimation results, where I only considered the frames in which the face was correctly detected by the four methods. The online available code of Fanelli et al. [52] approach does not estimate the roll angle; therefore, their error estimate of this angle is not presented. The results using RGBD-GMM method are superior to all other approaches, which emphasizes the better generalization capability of the developed method. Interestingly, all our proposed methods generalized better than the approach of Fanelli et al. [52]. Pose estimates over a sequence of images, taken from the cross-database evaluation on

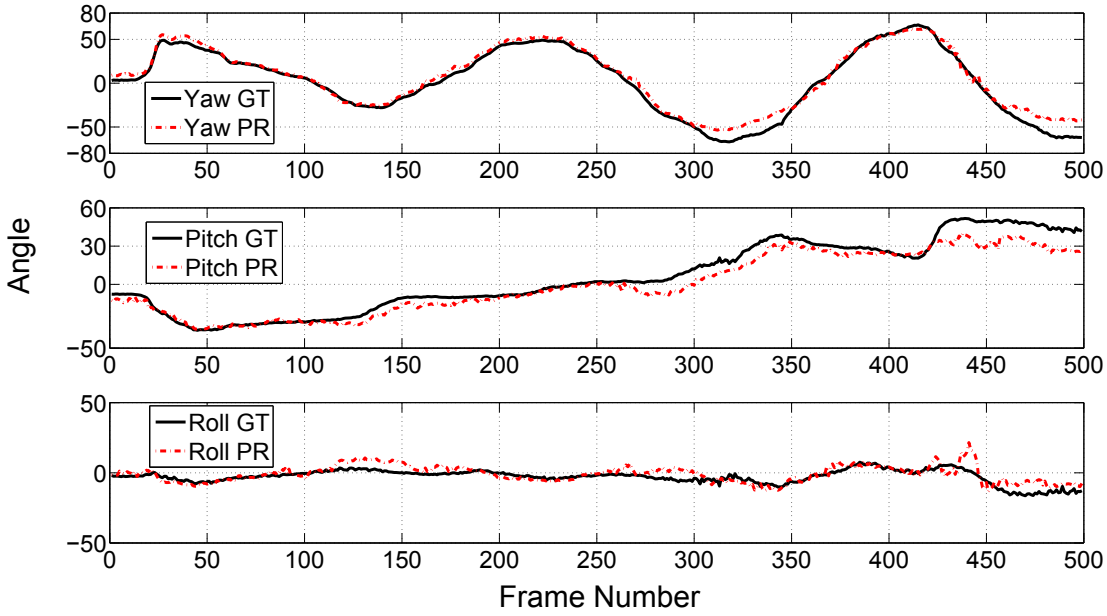


Figure 6.14: Samples of the head pose estimation over an image sequence using our approach of DRGB-GMM. They were taken from the cross-validation experiment, conducted on the BIWI database. GT denotes the ground truth, PR the predicted angles.

ICT-3DHP database, are depicted in Figure 6.15. Similar to the within-database evaluation, the estimation error increases only when the face experiences greater rotation angles; however, the results are promising to be used for head gesture recognition.

6.3.4 Processing Time Analysis

In this section, I extend the comparison between the feature types, appeared in Tables 6.2 and 6.3 in terms of accuracy, to be in terms of efficiency as well. The efficiency here is represented by feature extraction and regression times, taken into consideration that I employed SVM regressor for all feature types. This experiment was carried out on the BIWI database using Intel quad Core 2.33 GHZ, 8GB RAM, under windows environment, without any employment of a parallel programming possibility or a GPU implementation.

Table 6.6 shows the time required to extract and apply the regression models for each feature type. These time values were recorded when I used each feature type alone, assuming concatenated cases will consume the sum of all contained

Table 6.5: The mean/standard deviation of the absolute error for each head pose angle stemmed from the cross-database validation. These head pose estimators were trained on the BIWI database and tested on the ICT-3DHP database.

Approach	**Pitch Er** $^\circ$	**Yaw Er** $^\circ$	**Roll Er** $^\circ$
our RGB-VJ	5.32 / 5.7	5.3 / 5.5	4.3 / 4.5
our DRGB-VJ	4.9 / 5.3	5.1 / 5.4	4.4 / 4.6
Fanelli et al. [52]	5.9 / 6.3	6.3 / 6.9	-
our DRGB-GMM	**4.23 / 4.41**	**4.64 / 4.9**	**4.33 / 4.6**

Table 6.6: The process time of the pose estimation in terms of feature extraction and regression times. To get an intuitive meaning, the times are presented as second / frame per second (s/fps).

Feature	**Extraction Time** (s/fps)	**Regression Time** (s/fps)
LBP	0.011/90	0.016/60
GAB	0.1/10	0.0142/70
HOG	0.010/100	0.0149/67
HPC + MCDP	**0.003/300**	**0.0025/400**

feature times. Extracting the GAB features is the most time-consuming process among other feature types; however, it does not provide the best accuracy. HOG and LBP are extracted and classified at approximately the same time. Interestingly, the depth-based features HPC + MCDP, developed here, are extracted and classified at a higher speed. Furthermore, they provide competitive estimating results as shown in Tables 6.2 and 6.3. Concatenating several feature types enhances the estimation accuracy, but at the cost of more processing time.

6.4 Discussion

In this chapter, I have proposed an algorithm to automatically estimate the head pose from RGB/RGBD images on a frame basis. The automatic process here involves locating the face as well. Several feature types, extracted from RGB and depth images or retrieved head 3D point cloud, were exploited to encode the face appearance and shape. I have presented a fair comparison between them in terms

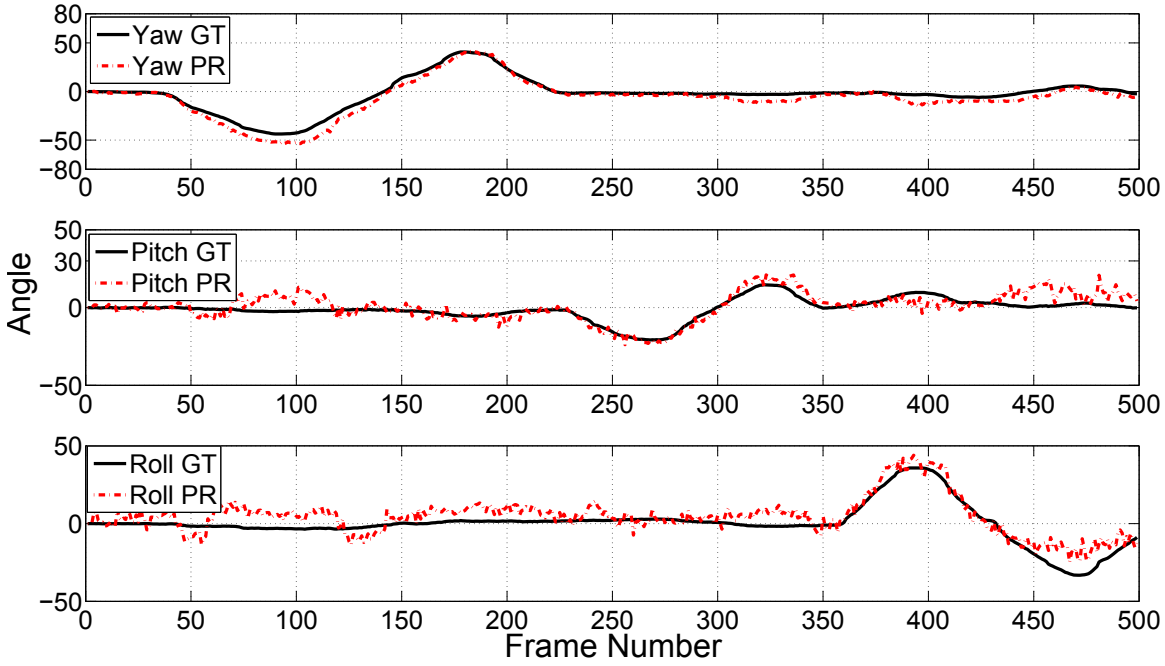


Figure 6.15: Samples of the head pose estimation over an image sequence. They were taken from a cross-database validation; the pose models were trained using the BIWI database and tested on the ICT-3DHP database using our approach of DRGB-GMM. GT denotes the ground truth, PR the predicted angles.

of accuracy and efficiency. Our depth-based features were the most efficient that provide competitive results as well. Meanwhile, HoG descriptor has proven to be the best among other investigated types in terms of accuracy.

Exploiting depth-based features improves the pose estimation accuracy; and employing the depth information in the face localization stabilizes the detector output which advances the pose estimation as well. Three methods to consistently crop the face have been proposed; two of them were built on top of the VJ face detector. One method is RGB-based; consequently, it is applicable to any conventional 2D RGB camera. Finally, the conducted cross-database evaluations have proven the generalization capability of our methods.

CHAPTER 7

Facial Expression Recognition

Each facial expression has its own distinctive print on the face. In this chapter, I propose several methods to recognize the six basic expressions (happiness (Ha), surprise (Su), anger (An), disgust (Di), fear (Fe), sadness (Sa)) along with the neutral state (Ne) based on reading their prints. Unless I state the opposite, throughout this work I consider the full automatic pipeline for the proposed approaches, starting from the face detection, passing by the cropping refinement of the located face patch and by the feature extraction, ending with assigning an expression to the face patch. This work aims to provide a frame-based decision; therefore, spatio-temporal features are not exploited. Precisely, each image sequence from the considered databases is represented by its apex frame, a frame in which the expression is at its apex; however, I highlight the importance of having prior-knowledge of person-specific neutral state, which shall minimize the intra-class variation due to the variability of the face geometry across individuals.

As each expression causes different facial deformations, consequently, an expression can be recognized by encoding the observed deformations and mapping them to it using a machine learning method. Instead of estimating the facial AUs and accordingly the expression, I recognize the expressions from features directly extracted from the face patch, of course those features implicitly encode the AUs. Mainly, two feature types (texture and geometry) are usually adapted to encode the facial deformations. Appearance-based methods, in which texture features are

extracted from the entire face patch, are more suitable to work on a frame basis as they reveal the entire facial appearance variations among the expressions including the facial wrinkles, bulges, and furrows. On the other hand, they are more susceptible to illumination variations and noisy textures. The second type is geometry-based methods, in which the relative locations of the facial points to each other are encoded. Upon the availability of knowledge of a person-specific neutral state, these features can be generalized across individuals by being normalized with respect to their values at the neutral expression. The performance of the geometry-based approaches strongly relies upon the accuracy of the facial point localization. Interestingly, the geometry-based methods can be extended to work with a wide range of poses, at cost of more processing as one can correct the extracted geometric features according to the processed pose. On the other hand, extending an appearance based method to handle a wide range of poses would be at cost of more resources as more classifiers for each group of poses are required. Note that extending the proposed methods to work across head poses is out of the scope of this work. The performance of the proposed approaches here is investigated under various factors, e.g. type of employed machine learning method, feature descriptor, availability of person-specific neutral state, and number of the exploited facial points.

7.1 Appearance-based Method

Locating the face within a processed image is the first task in the appearance-based approach. In my proposed approach, shown in Figure 7.1, I employ the VJ face detector. Next, the detected face goes through a cropping refinement process, as described in Sec. 5.1. A consistent cropping yields better recognition results due to minimizing the intra-class variations. The facial appearance is then encoded with texture-based descriptors. Finally, the extracted features are mapped to the facial expression via a Machine learning (ML) algorithm. In what follows, I present the recognition rates resulting from exploiting the most effective texture descriptors along with different ML algorithms.

7.1.1 Local Binary Pattern Features

LBP descriptor has been proven to be a powerful means of texture description. Additionally, it was successfully employed in many computer vision systems, in

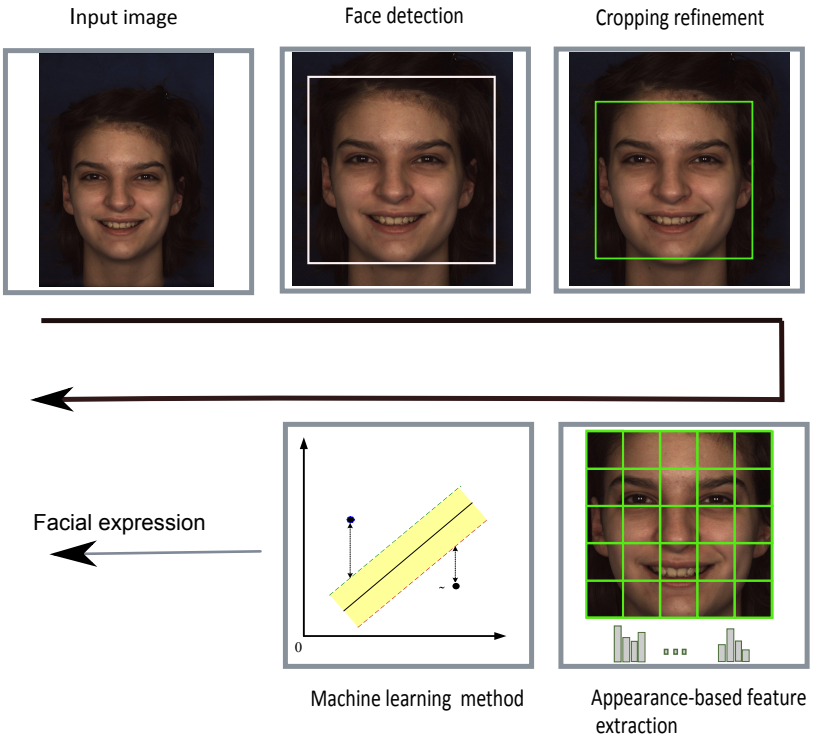


Figure 7.1: The structure of the proposed appearance-based algorithm for facial expression recognition.

expression recognition as well but with different setups and assumptions. Besides its computational simplicity and illumination invariance, LBP descriptor encodes different texture primitives, such as spot, edge, and corner. To build the LBP feature vector, first, I apply its operator to each pixel of the gray image of the cropped face patch, as explained in Sec. 3.2.2. Then, the resulting values are grouped in histograms, each histogram represents a specific cell and is of 10 bins. The face patch is scaled into 150×150 pixels before being divided into cells of 10×10 pixels. Finally, those histograms are concatenated to form the LBP feature vector of length of 2250. To assess the reliability of each proposed approach here, I evaluated it using two facial expression databases that were summarized in Sec. 4.3. Due to the lack of training samples, I employed a leave-one-out subject cross-validation (LOOCV) strategy. As the name suggests, one subject is left out for testing, while the rest samples are used for training; therefore, no samples from the same subject coexist in the training and testing sets.

The results here are obtained by using LBP features in the algorithm depicted in Figure 7.1. Tables 7.1, 7.2 summarize the results of the evaluations conducted on CK+ database, Table 7.1 for the 6-class case and Table 7.2 for the 7-class case. Each

column represents samples of the predicted class while each row represents samples of the ground truth class. For a comparison purpose, the results were obtained using four different machine learning algorithms: SVM, NNe, RF, and kNN, where their parameters in addition to the LBP parameters (bin number, cell size) were optimized using a grid-search with cross-validation experiment conducted on the training data with a goal of achieving better average recognition rate at reasonable processing and resource cost.
A number of points can be drawn from these depicted tables. In Table 7.1, the achieved average recognition rates were $70.91\%, 66.22\%, 59,40\%,$ and $54,07\%$ using SVM, NNe, RF, and kNN, respectively. By adding the neutral state as a new category to the classification method, the average recognition rates degraded to $64.89\%, 59.79\%, 58.48\%, 49.69\%$ using SVM, NNe, RF, and kNN, respectively. This lower performance is reasonable as considerable confusions between the neutral and the other expressions were introduced, especially the expressions of subtle deformations like sadness and anger. In both cases, the best results were obtained via SVM, then NNe, next RF and finally using kNN. The expressions characterized by higher facial deformations, like happiness and surprise, have been always recognized with higher rates. The evaluations of LBP on the BU-4DFE database are summarized in Appendix A.1.1.

7.1.2 Gabor Filter-based Features

GAB features have been shown to be suitable for texture description due to their spatial and frequency locality besides the orientation selectivity. In addition to the facial expression recognition, GAB features have been successfully used for texture discrimination, image matching, and object recognition, more information about GAB were provided in Sec. 3.2.1. In this section, I employ the GAB features in the proposed appearance-based approach for facial expression recognition (Figure 7.1). The classification is performed using SVM as it has provided the best performance in the evaluation conducted in Sec. 7.1.1.
To produce the GAB feature vector, I exploit a bank of Gabor kernels, in which I use two values of λ, 6 for θ, 2 for σ, where $\sigma_x = \sigma_y$. I convolve each Gabor kernel with a scaled face patch of 100×100 pixels. Then, I divide the resulting patch into smaller cells of 10×10 pixels. Only a median value is taken from each cell to form

Table 7.1: Confusion matrix of 6-class facial expression recognition using LBP features based on evaluation conducted on CK+ database. For each expression, four rows are presented each corresponds to specific machine learning algorithm. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

		Ha	Su	An	Di	Fe	Sa
Ha	SVM	**0.8857**	0	0.0429	0.0429	0.0286	0
	NNe	**0.9000**	0.0714	0	0.0286	0	0
	RF	0.7183	0.1268	0.0704	0.0845	0	0
	KNN	0.7143	0.1714	0.0429	0.0714	0	0
Su	SVM	0.0122	**0.8902**	0.0244	0	0.0488	0.0244
	NNe	0	**0.9268**	0	0.0122	0.0610	0
	RF	0.0122	**0.8659**	0.0122	0.0732	0	0.0366
	KNN	0.1220	0.6341	0.0366	0.1707	0.0122	0.0244
An	SVM	0.0222	0	**0.6444**	0.0889	0.0889	0.1556
	NNe	0.2000	0.0667	0.5111	0.2000	0	0.0222
	RF	0.1333	0.1556	0.5333	0.1556	0	0.0222
	KNN	0.1556	0.0667	0.4889	0.1778	0.0667	0.0444
Di	SVM	0.0508	0.0169	0.1017	**0.7627**	0.0169	0.0508
	NNe	0.1525	0.1525	0.0169	**0.6780**	0	0
	RF	0.1356	0.1695	0.1017	0.5593	0	0.0339
	KNN	0.1525	0.1525	0.0339	0.5932	0.0508	0.0169
Fe	SVM	0.1600	0.0800	0	0.0400	0.5200	0.2000
	NNe	0.2400	0.2000	0.0400	0.0800	0.4400	0
	RF	0.1538	0.1154	0.1154	0.1538	0.4231	0.0385
	KNN	0.0400	0.2800	0.1600	0.0800	0.4000	0.0400
Sa	SVM	0.0345	0.0690	0.1724	0.1379	0.0345	0.5517
	NNe	0.0690	0.1724	0.1034	0.1379	0	0.5172
	RF	0.0357	0.2143	0.1429	0.1071	0.0357	0.4643
	KNN	0	0.2759	0.1724	0.1034	0.0345	0.4138

Table 7.2: Confusion matrix of 7-class facial expression recognition using LBP features based on evaluation conducted on CK+ database. For each expression, four rows are presented each corresponds to specific machine learning algorithm. Each column represents samples of the predicted class, ncfcv while each row represents samples of the ground truth class.

		Ha	Su	An	Di	Fe	Sa	Ne
Ha	SVM	**0.8571**	0	0.0143	0.0429	0.0571	0	0.0286
	NNe	**0.9143**	0.0429	0	0	0	0	0.0429
	RF	**0.8143**	0	0	0	0.0571	0	0.1286
	KNN	0.5286	0.1143	0.0286	0.1000	0	0	0.2286
Su	SVM	0.0125	**0.8625**	0	0	0	0	0.1250
	NNe	0.1039	**0.6104**	0.0130	0.0390	0.0779	0.0649	0.0909
	RF	0.0122	**0.6463**	0	0.0122	0.0854	0.0732	0.1707
	KNN	0.0610	0.6098	0.0366	0.0732	0	0.0244	0.1951
An	SVM	0	0	0.3182	0.0455	0.0227	0.0455	0.5682
	NNe	0.0889	0.1778	0.4667	0	0.0889	0.0444	0.1333
	RF	0.0465	0.0233	0.5349	0.0698	0	0.1163	0.2093
	KNN	0.1778	0.0667	0.4222	0.0444	0.0444	0	0.2444
Di	SVM	0.0172	0	0.0690	**0.7414**	0	0.0172	0.1552
	NNe	0.1912	0.1471	0	0.3824	0.0735	0.0588	0.1471
	RF	0.0208	0.0208	0.0208	0.4792	0.0833	0.0833	0.2917
	KNN	0.2034	0.0678	0.0678	0.4746	0	0	0.1864
Fe	SVM	0.1250	0.2083	0.1250	0	0.4167	0	0.1250
	NNe	0.1935	0.2258	0	0	0.4839	0	0.0968
	RF	0.2800	0.0800	0.0400	0.0400	0.3200	0	0.2400
	KNN	0	0.1429	0.0952	0	0.4762	0	0.2857
Sa	SVM	0	0	0.1034	0.0345	0.0345	0.4483	0.3793
	NNe	0.0465	0.1628	0	0	0	0.4651	0.3256
	RF	0	0.0690	0.0345	0.1034	0	0.4828	0.3103
	KNN	0.1071	0.0714	0.0714	0.1071	0	0.4286	0.2143
Ne	SVM	0.0051	0.0203	0.0457	0.0203	0.0051	0.0051	**0.8985**
	NNe	0.0355	0.1015	0	0	0	0	**0.8629**
	RF	0	0.0204	0.0102	0	0.0765	0.0765	**0.8163**
	KNN	0.2030	0.1371	0.0457	0.0761	0	0	0.5381

Table 7.3: Confusion matrix of 6-class facial expression recognition using GAB features based on LOOCV evaluation conducted on CK+ database via SVM. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.9429**	0	0.0143	0.0143	0.0143	0.0143
Su	0.0122	**0.9512**	0.0122	0	0.0122	0.0122
An	0.0222	0	**0.8000**	0.0889	0.0222	0.0667
Di	0.0169	0	0.0847	**0.8983**	0	0
Fe	0.0400	0.0400	0.0800	0.0400	**0.7200**	0.0800
Sa	0.0345	0.0345	0.2069	0.0690	0.0690	0.5862

Table 7.4: Confusion matrix of 7-class facial expression recognition using GAB features based on LOOCV evaluation conducted on CK+ database via SVM. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.9286**	0	0	0.0143	0.0143	0	0.0429
Su	0	**0.9512**	0	0	0.0122	0	0.0366
An	0	0	**0.6444**	0.0444	0	0	0.3111
Di	0	0	0.0508	**0.8814**	0	0	0.0678
Fe	0.0400	0.0400	0.0400	0	**0.7200**	0.0400	0.1200
Sa	0	0.0345	0.0345	0	0	0.4138	0.5172
Ne	0	0	0.0203	0	0.0051	0.0254	**0.9492**

the kernel feature vector. Finally, I concatenate the vectors obtained by all kernels to produce the GAB feature vector of length 2400. The parameters of both GAB and SVM were optimized using a grid-search with cross-validation experiment conducted on the training data with a goal of achieving better average recognition rate at reasonable processing and resource cost.

Tables 7.3 and 7.4 summarize the confusion matrices resulting from the evaluations (LOOCV) that were conducted on CK+ database for the 6-class and 7-class

cases, respectively. The average recognition rate dropped from 81.64% for the 6-class case to 78.41% for the 7-class case due to confusions between the added neutral state and the facial expressions of subtle deformations (anger and sadness). The recognition rate of the anger expression fell from 80% to 64.4%, where 31% of the anger samples are recognized as neutral. The sadness expression is confused with anger by 20% in the 6-class case, and with neutral by 51% in the 7-class case. Happiness and surprise expressions are recognized with high rates in both cases as they have obvious facial deformations. The evaluations on BU-4DFE database are presented in Appendix A.1.2, where the achieved rates are lower than the rates obtained using the CK+ database, as the expression intensity varies significantly across individuals in BU-4DFE database.

7.1.3 Histogram of Oriented Gradient Features

HoG descriptor is one of the most effective low-level texture descriptors. It has been successfully employed in several computer vision systems. Originally, it was developed to tackle the pedestrian detection before exploiting it in many other applications such as facial analysis and object recognition, more information about it were provided in Sec. 3.2.3. In this section, I employ the HoG descriptor in the proposed appearance-based approach for facial expression recognition (Figure 7.1). The classification was accomplished using SVM as it has provided the best performance in the evaluation conducted in Sec. 7.1.1. To produce the HoG feature vector, I scale the face patch to 160×160 pixels and use a cell size of 20×20 pixels, block of 40×40 pixels, block stride of one cell. For each cell, I create an eight-bin orientation histogram where each pixel orientation is weighted by its magnitude. By concatenating the normalized histograms over the block regions, I form the final HoG feature vector of length 1568. The parameters of both HoG and SVM were optimized using a grid-search with cross-validation experiment conducted on the training data with a goal of achieving better average recognition rate at reasonable processing and resource cost.

Tables 7.5 and 7.6 summarize the confusion matrices resulting from the evaluations (LOOCV) that were conducted on CK+ database for the 6-class and 7-class cases, respectively. While the average recognition rate reached 87.26% for the 6-class case, it dropped to 83.71% for the 7-class case due to confusions with the added neutral

Table 7.5: Confusion matrix of 6-class facial expression recognition using HOG features based on LOOCV evaluation conducted on CK+ database via SVM. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.9714**	0	0.0143	0.0143	0	0
Su	0	**1.0000**	0	0	0	0
An	0.0667	0	**0.7333**	0.0667	0	0.1333
Di	0	0	0.0678	**0.9322**	0	0
Fe	0.0800	0.0400	0.0400	0	**0.8400**	0
Sa	0.0345	0.0345	0.1724	0	0	**0.7586**

class. In Table 7.5, surprise, happiness, and disgust expressions were recognized with high rates $100\%, 97.14\%, 93.22\%$, respectively. Sadness and anger, the expressions of subtle facial deformations, were recognized with relatively lower rates $75.86\%, 73.33\%$, respectively. The most confusions occurred between anger and sadness, where 17% of the sadness samples were recognized as anger, and 13% of anger samples were identified as sadness. The fear expression was recognized with 84%, while 8% of its samples were recognized as happiness, mainly because both expressions involve mouth opening. In the 7-class case (Table 7.6), happiness, disgust, surprise expressions were still recognized with higher rates $100\%, 100\%, 89\%$, respectively. Neutral and anger were recognized with higher rates as well. On the other side, the recognition rates of fear and sadness dropped dramatically from 84% and 75.86% to 67% and 50%, respectively. The main reason for that is the confusions with the added neutral class, mainly 40% of sadness samples and 22% of the fear were recognized as neutral. Sadness is the most difficult expression to recognize due to its subtle deformations. The evaluations on the BU-4DFE database are presented in Appendix A.1.3, where the achieved rates are lower than the rates obtained using the CK+ database. Additionally, dedicating a separate class for the neutral state leads to bigger drop in the average recognition rate, which highlights the higher variation within same class samples of BU-4DFE database in terms of intensity.

Table 7.6: Confusion matrix of 7-class facial expression recognition using HOG features based on LOOCV evaluation conducted on CK+ database via SVM. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**1.00**	0	0	0	0	0	0
Su	0	**0.89**	0	0	0	0	0.11
An	0	0	**0.93**	0	0	0.07	0
Di	0	0	0	**1.00**	0	0	0
Fe	0.11	0	0	0	**0.67**	0	0.22
Sa	0	0	0.10	0	0	0.50	0.40
Ne	0	0	0.10	0.03	0	0	**0.87**

7.1.4 Discussion

Appearance-based approaches are suitable to perform facial expression recognition in a frame basis, especially when no prior information is available. By prior-information, I refer here to a person-specific face model for the neutral state. The pipeline of my proposed approach is depicted in Figure 7.1. To stabilize the output of the face detector, I performed a cropping refinement as explained in details in Sec. 5.1. The main objective of this refinement is to minimize the intra-class variation and consequently to enhance the learning process. As the face appearance changes across the facial expressions, I encoded the face appearance using texture descriptors and then utilized the resulting features to train a multi-class classifier. I have presented the recognition rates that were obtained by exploiting three different texture-based descriptors (HoG, GAB, LBP). In the evaluation of LBP, SVM classifier has provided the best performance in terms of recognition rate in comparison to NNe, RF, kNN. SVM has proven its effectiveness in the case when the number of training samples is lower than the feature vector dimension. Consequently, the SVM classifier is exclusively employed in the evaluations of GAB and HoG features. LBP features are simple to calculate, but perform poorly. GAB features are time-consuming to calculate, and it is hard to find the proper set of its parameters. On the other hand, they have more distinctive power than LBP features. HoG was the most effective feature descriptor, easy to calculate and configure. Using the SVM classifiers, I achieved average recognition rates 87.26%, 81.64%,

and 70.91% using HoG, GAB, and LBP, respectively, in the 6-class evaluations on CK+ database. Those rates dropped to 83.71%, 78.41%, and 64.89%, respectively, due to confusions with the added neutral class in the 7-class evaluations on CK+ database. Furthermore, HoG performed the best in the evaluations conducted on BU-4DFE database, but with lower rates due to the more variations in the intensities of the BU-4DFE samples of the same expression class. Happiness and surprise are the most obvious expressions, they were recognized with high rates in both cases: 6-class and 7-class. On the other hand, sadness and anger, the expressions of subtle deformations, were recognized with relatively low rates and confused more with the neutral state in the 7-class case.

7.2 Geometry-based Method

For facial expression recognition, geomtric-based approaches utilize the facial point position, relative location to specific model, or movements across an image sequence. They are mainly used to provide video-based decisions based on spatio-temporal features extracted from tracking the facial points across the entire image sequence. Differently here as our approach is constrained to be frame-based, I extract geometry-based features encoding the facial point position and their relative location to neutral models to recognize the facial expression on a frame basis. In what follows, I tackle the facial expression recognition via developing geometry-based approaches that vary in their configuration: number of the utilized facial points, type of the employed neutral model either person-specific or general. Figure 7.2 presents the general structure of the proposed geometry-based approach. After locating the facial points within the detected face, I extract several features encoding the relative location of the facial points to each other and to their location in the person-specific or general neutral model. Finally, those features are assigned to a facial expression via SVM classifier.

7.2.1 A Method of 49 Facial Points

In this section, I propose a geometry-based approach that utilizes the relative location of 49 facial points to their location in a person-specific neutral model. To minimize the individual variation, I weight each displacement with respect to the person-specific face configuration in the way to produce the final feature descriptor. Figure 7.3 shows the 49 facial points used here; I developed an approach

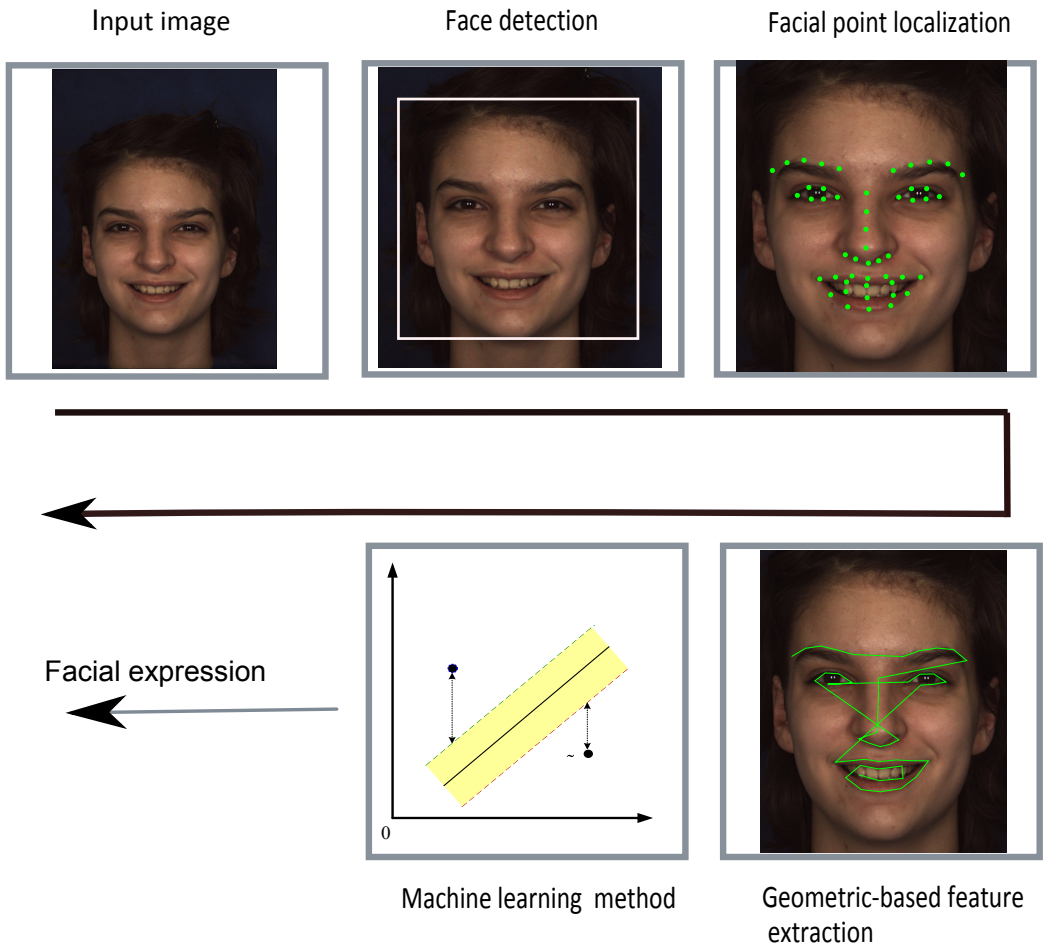
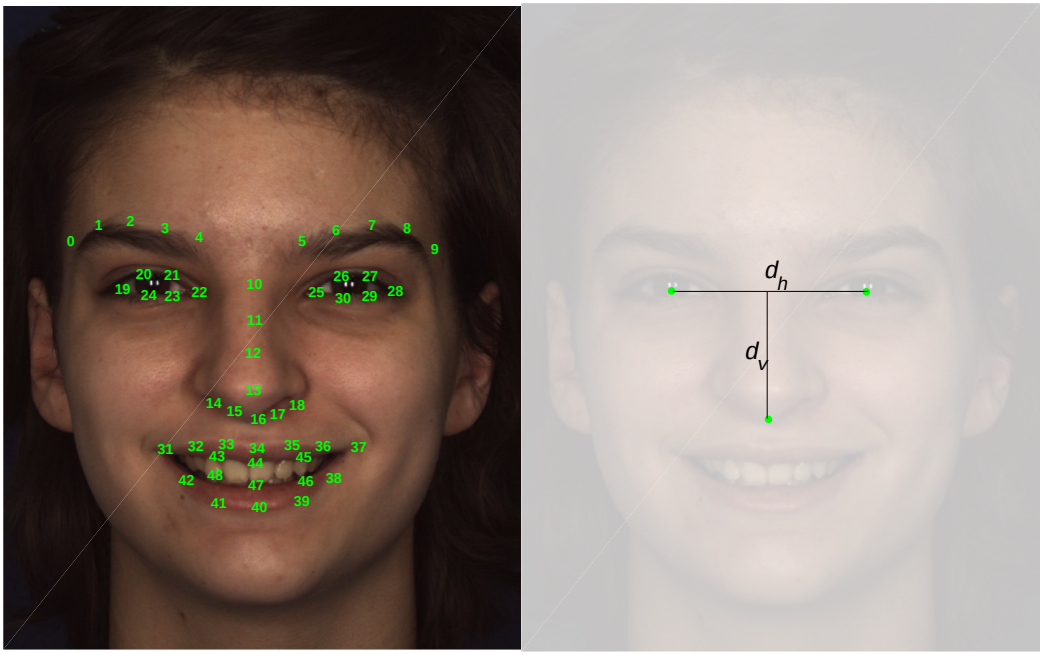


Figure 7.2: The structure of the proposed geometric-based algorithm for facial expression recognition.

to automatically locate them as detailed in Ch. 5. Before extracting the geometric features, I estimate the affine transformation between the facial points in the processed frame and its prior-known person-specific location in the neutral state, under the assumption that the processed face is in near frontal pose. To this end, I use 3 facial points as they are not influenced by any deformations of the facial muscles, the eyes center and the nose bottom point as shown in Figure 7.3b. The locations of the three points ($\mathbf{p}_{3p} = \{\mathbf{p}_{ecr}, \mathbf{p}_{ecl}, \mathbf{p}_{nb}\} \in \mathbb{R}^{2\times 3}$) are inferred with respect to the 49 points in Figure 7.3a as follows.

$$\begin{aligned} \mathbf{p}_{ecr} &= \frac{\mathbf{p}_{19} + \mathbf{p}_{22}}{2} \\ \mathbf{p}_{ecl} &= \frac{\mathbf{p}_{25} + \mathbf{p}_{28}}{2} \\ \mathbf{p}_{nb} &= \mathbf{p}_{16} \end{aligned} \tag{7.1}$$



(a) (b)

Figure 7.3: (a) The 49 facial points used in the proposed geometric-based approach for facial expression recognition. (b) Person-specific normalized factors for horizontal and vertical distances.

Next, I derive the affine transformation, in terms of a multiplication matrix $\mathbf{A} \in \mathbb{R}^{2\times2}$ and a translation vector $\mathbf{t} \in \mathbb{R}^{2\times1}$, that maps the three points in the processed frame to their equivalent in the neutral frame, satisfying the following augmented matrix.

$$\begin{bmatrix} \mathbf{p}_{N3p} \\ \mathbf{1}_{1\times3} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{t} \\ \mathbf{0}_{1\times2} & \mathbf{1}_{1\times1} \end{bmatrix} \times \begin{bmatrix} \mathbf{p}_{3p} \\ \mathbf{1}_{1\times3} \end{bmatrix} \tag{7.2}$$

Some straightforward calculations using Eq. (7.2) lead to the values of $\mathbf{A}$ and $\mathbf{t}$. $\mathbf{1}_{a\times b}$ is an $a \times b$ matrix whose all elements are one. $\mathbf{0}_{a\times b}$ is an $a \times b$ matrix whose all elements are zero. Then, I exploit the obtained matrices ($\mathbf{A}$, $\mathbf{t}$) to transfer the located 49 facial points ($\mathbf{p}_{49p}$) in the processed frame to the neutral space ($\mathbf{p}_{N49pt}$) as follows.

$$\begin{bmatrix} \mathbf{p}_{N49pt} \\ \mathbf{1}_{1\times49} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{t} \\ \mathbf{0}_{1\times2} & \mathbf{1}_{1\times1} \end{bmatrix} \times \begin{bmatrix} \mathbf{p}_{49p} \\ \mathbf{1}_{1\times49} \end{bmatrix} \tag{7.3}$$

To produce the geometric features, I first calculate the displacement between the transformed points and their equivalence in the neutral frame.

$$\Delta\mathbf{p}_{49} = \mathbf{p}_{N49pt} - \mathbf{p}_{N49p}, \qquad \{\Delta x_i, \Delta y_i, \ldots, \Delta x_{48}, \Delta y_{48}\} \wedge i \neq 16 \tag{7.4}$$

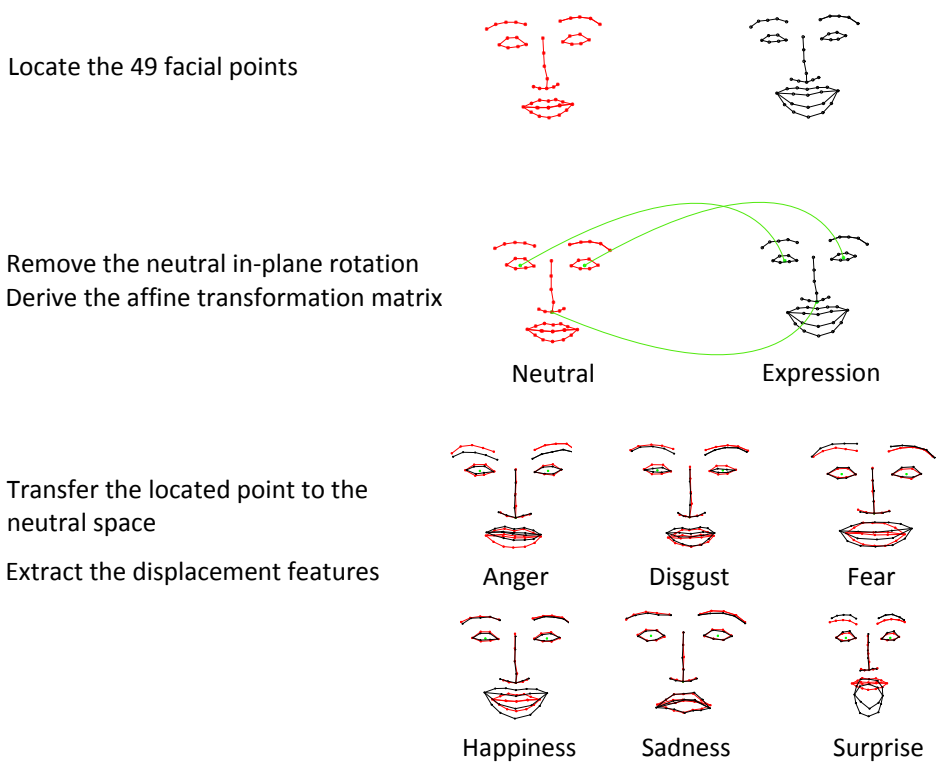


Figure 7.4: The feature extraction process of the propose geometric-based approach that exploits 49 facial points.

To minimize the individual variation, the displacements are evaluated with respect to person-specific face configuration (d_h, d_v, See Figure 7.3b) as follows.

$$\Delta\tilde{\mathbf{p}}_{49} = \{\frac{\Delta x_1}{d_h}, \frac{\Delta y_1}{d_v}, \ldots, \frac{\Delta x_i}{d_h}, \frac{\Delta y_i}{d_v}, \ldots, \frac{\Delta x_{48}}{d_h}, \frac{\Delta y_{48}}{d_v}\} \wedge i \neq 16 \tag{7.5}$$

Finally, to remove the dominant effect of the large range features, a standardized version of $\Delta\tilde{\mathbf{p}}_{49}$ is considered the geometric feature vector that is of length 96. The feature extraction process is summarized in Figure 7.4. I employ the SVM to assign a facial expression to the extracted feature vector. Similar to the appearance-based evaluation, due to the lack of samples I performed LOOCV. The resulting confusion matrix is shown in Table 7.7. The first row of each class summarizes the person-specific case, while the second row stemmed from employing a general neutral model instead of the person-specific one. In this evaluation, the facial point localization within the neutral frame (first frame of each sequence) is considered as the person-specific neutral model, while an averaging of the neutral points' location over all training subjects makes the general neutral model. The faces were arranged to have a similar interocular distance, the distance between the eye centers, prior the averaging. In real scenarios, the person-specific neutral state could

Table 7.7: Confusion matrix of 6-class facial expression recognition using geometric features extracted from 49 facial points, based on evaluation conducted on CK+ database via SVM. For each expression, two rows are presented. The first row is dedicated for the person-specific scenario, the features are calculated with respect to a priorly known person-specific neutral model. The second row is the case where a general neutral model is used. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**1.0000**	0	0	0	0	0
	0.9710	0	0.0145	0	0.0145	0
Su	0	**0.9878**	0	0	0	0.0122
	0	**0.9756**	0	0	0.0122	0.0122
An	0	0	**0.9111**	0.0444	0	0.0444
	0	0	**0.7778**	0.1333	0	0.0889
Di	0	0	0.0169	**0.9831**	0	0
	0.0339	0	0.0847	**0.8814**	0	0
Fe	0.1200	0.0800	0	0.0400	**0.6800**	0.0800
	0.0800	0.0400	0	0	**0.8000**	0.0800
Sa	0	0	0.1429	0	0.0357	**0.8214**
	0	0	0.1429	0.0357	0.0357	**0.7857**

be obtained, with human intervention, during an initial registration step. It can be derived automatically as well by averaging the facial point detection of the considered person for a long-period based on the assumption that emotional expressions spread just over few frames.

In Table 7.7, a conducted cross-validation on CK+ database shows that I can achieve an average recognition rate of 89.72% when the features are extracted with respect to person-specific neutral state and a rate of 86.52% in the case of using a general neutral model. A drop of 3.2% cannot be avoided without the person-specific prior information. More confusions are experienced among the expressions of subtle facial deformations (sadness, anger) upon using the general neutral model. In the meantime, the recognition rates of happiness and surprise, expressions of high facial deformations, are not affected. Personalized geometric features

Table 7.8: Confusion matrix of 7-class facial expression recognition using geometric features extracted from 49 facial points, based on evaluation conducted on CK+ database via SVM. Here, we infer the neutral state as person-specific neutral state is not available Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.9855**	0	0	0	0	0	0.0145
Su	0	**0.9634**	0	0	0.0122	0.0122	0.0122
An	0	0	**0.7111**	0.0667	0	0.0222	0.2000
Di	0	0	0.0678	**0.8814**	0	0	0.0508
Fe	0.0800	0.0800	0	0	**0.8000**	0	0.0400
Sa	0	0	0.1071	0	0	0.5357	0.3571
Ne	0	0	0.0435	0.0290	0.0145	0.0580	**0.8551**

lead to better performance, which highlights potential improvements when one personalizes the training and testing as well.

In the case of lack of person-specific information, it is sensible to automatically infer the neutral state as well. Table 7.8 summarizes the results of the conducted cross-validation in the case of 7-class without prior person-specific information. The features were extracted with respect to the general neutral model derived from the training data. Normally, adding new categories to the classifier drops its average recognition rate due to new confusions with existing categories. Here, the major confusions arise between neutral and sadness; neutral and anger, due to the small subtle facial deformations in both sadness and anger expressions. For the CK+ database, the achieved average recognition rate is 81.89%, where happiness and surprise are recognized with 98.55% and 96.34%, respectively. Disgust and neutral are recognized with high rates as well, 80% and 85.51%, respectively. on the other side, the recognition rate of sadness dropped dramatically to 53.57%, where 35.71% of the sadness samples are identified as neutral. A similar trend in drop of the recognition rates was experienced in the conducted cross-validation on BU-4DFE database, more details are provided in Appendix A.2.1.

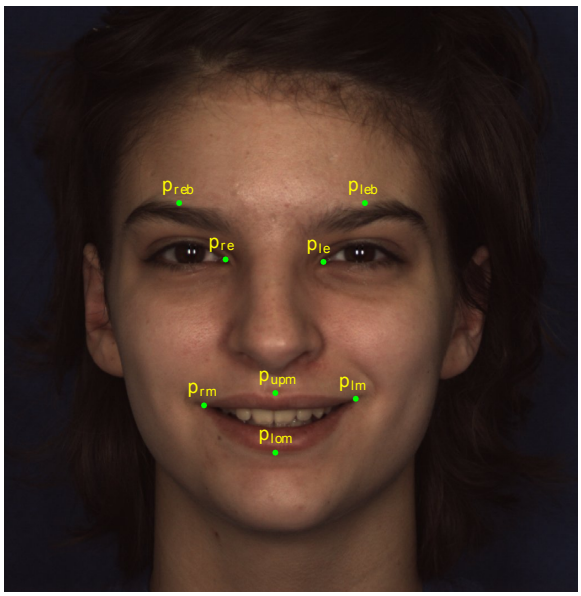


Figure 7.5: The eight facial points exploited by our proposed geometric-based approach to recognize the facial expressions.

7.2.2 A Method of 8 Facial Points

In this section besides emphasizing the improvement caused by exploiting a person-specific neutral model, I propose an approach that achieves satisfactory recognition rates with fewer facial points.

Here, I consider only eight facial points sampling three face components: four points for the mouth corners ($\mathbf{p}_{rm}, \mathbf{p}_{lm}, \mathbf{p}_{upm}, \mathbf{p}_{lom}$), two points for inner eye corners ($\mathbf{p}_{re}$,$\mathbf{p}_{le}$), two points for the center position of the eyebrow ($\mathbf{p}_{reb}$,$\mathbf{p}_{leb}$), as shown in Figure 7.5.

$$P_s = \{p_{reb}, p_{leb}, p_{rec}, p_{lec}, p_{rm}, p_{lm}, p_{upm}, p_{lom}\}, \quad p_i \in \mathbb{R}^2. \tag{7.6}$$

Those eight facial points describe three main facial components, additionally they are edge points, consequently could be accurately located compared to others [14]. Moreover, they have shown to perform well in the facial expression recognition based on 3D video input [122], the eye points here differ. The evaluations were conducted on the two facial expression databases CK+ and BU-4DFE as well. The eight facial points were localized as follows.

7.2.2.1 The Localization of the 8 Facial Points

For the CK+ database, I used the 8 points from the provided ground truth. These annotations were obtained by manually labeling Key frames within each image sequence with 68 facial points and after that using a gradient descent active appearance model (AAM) to fit these points in the remaining frames.

For the BU-4DFE database, the eight facial points were detected in the first frame of each sequence (neutral expression frame) using the developed point detector by [157], and then tracked using a dense optical flow tracking algorithm [103] through the remaining sequence. To constrain the facial points to be within the variance of the training set at each processed frame, I apply a developed Point Distribution Model (PDM) to the tracked points.
The first step in building the PDM is to align the facial points of all training samples. I consider only frontal faces, therefore, calculating the facial point positions with respect to the detected face width is supposed to satisfy the PDM requirements. The normalized eight facial points are concatenated to produce a vector of length 16. Each sample is given as

$$\mathbf{z} = (x_{p1}; y_{p1}; \cdots ; x_{p8}; y_{p8}), \quad \mathbf{z} \in \mathbb{R}^{16\times 1}. \tag{7.7}$$

Next, I calculate the covariance matrix over all the training samples (of all expressions) as follows.

$$\Sigma = \mathrm{E}\left[(\mathbf{z} - \mathrm{E}[\mathbf{z}]) \, (\mathbf{z} - \mathrm{E}[\mathbf{z}])^{\mathrm{T}} \right]. \tag{7.8}$$

Following this, I apply the SVD to the covariance matrix Σ (Eq. (7.8)) to be written as

$$\Sigma = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}}, \tag{7.9}$$

where $\mathbf{U}, \mathbf{S}, \mathbf{V}$ are matrices of size $16{*}16$. $\mathbf{U}$ and $\mathbf{V}$ are unitary matrices. $[\,]^{\mathrm{T}}$ denotes the matrix conjugate transpose. $\mathbf{S}$ is a diagonal matrix with diagonal entries $(\sqrt{\lambda_i})$ equal to the square root of eigenvalues from $\Sigma\Sigma^{\mathrm{T}}$. And the eigenvectors of $\Sigma\Sigma^{\mathrm{T}}$ make up the columns of $\mathbf{V}$. Each eigenvector describes a principal direction of variation within the training set with a corresponding standard deviation $(\sqrt{\lambda_i})$. Finally, each detected facial point $\grave{\mathbf{z}}$ should satisfy the following linear combination of the eigenvectors.

$$\grave{\mathbf{z}} = \bar{\mathbf{z}} + \mathbf{V}\mathbf{b}, \tag{7.10}$$

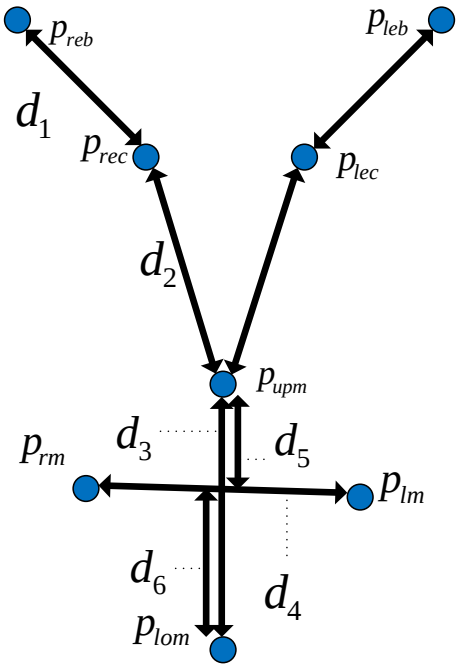


Figure 7.6: The six relative distances between the 8 exploited facial points, d_1 and d_2 are the average of two mirrored values on the left and right sides of the face.

where $\bar{\mathbf{z}}$ represents the mean of $\mathbf{z}$ across all training samples, $\mathbf{b}$ is a vector of scaling values for each principal component. Simply, to guarantee that the facial points fall within the variance of the training set, I truncate each element of $\mathbf{b}$ as follows

$$|b_i| \leq 2\sqrt{\lambda_i}\,, \qquad i = 1, ..., 16. \tag{7.11}$$

7.2.2.2 Person-specific Neutral State

In this experiment, I assume the prior-knowledge about the point location in the neutral state is available for each subject. To encode the facial deformations on a frame basis, I extract the feature vector $\mathbf{f}$ (Eq. (7.12)) that includes the ratio of distances between fiducial points measured in action apex $(d_{i(apex)})$ to that at the neutral state $(d_{i(neut)})$ (see Figure 7.6).

$$\mathbf{f} = (f_1, f_2, f_3, f_4, f_5, f_6), \quad f_i = \frac{d_{i(apex)}}{d_{i(neut)}} \tag{7.12}$$

d_1 and d_2 are the average of two mirrored values on the left and right sides of the face. These features implicitly include some AUs defined by Ekman et al. [47]. For example, f_1 carries information about the eyebrow AU1 (*Inner Brow Raiser*)

and AU4 (*Brow Lowerer*); f_2 indicates the movements of the upper lip AU10 (*Upper Lip Raiser*). The other four features cover some of the AUs related to the mouth. Although (f_5, f_6) look like redundant of f_3, they are not. The effect of (f_5, f_6) on overall recognition rates is discussed later. To remove the dominant effect of the large range features before passing the features into a machine learning algorithm, the feature vector $\mathbf{f}$ (Eq. (7.12)) is normalized to be $\tilde{\mathbf{f}} = (\tilde{f}_1, \tilde{f}_2, \tilde{f}_3, \tilde{f}_4, \tilde{f}_5, \tilde{f}_6)$ as follows.

$$\tilde{f}_i = \frac{\frac{f_i - \mu_i}{2\sigma_i} + 1}{2}, \qquad i = 1, ..., 6. \tag{7.13}$$

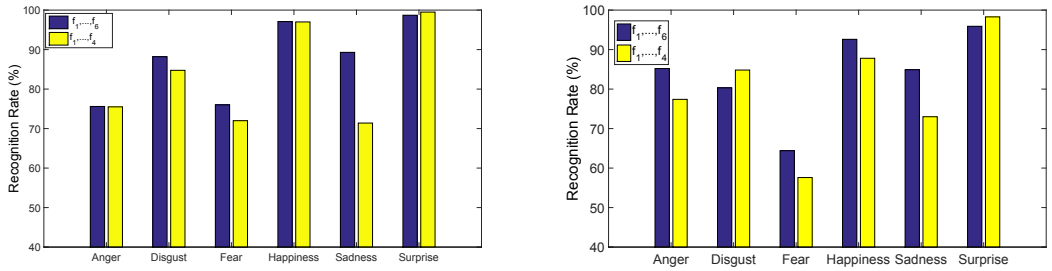
Where μ_i and σ_i are mean and standard deviation of the ith feature across the training data, respectively. If f_i is normally distributed, Eq. (7.13) guarantees 95% of $\tilde{f}_i$ to be in the [0,1] range.

As before, the facial expression recognition is formulated as a multi-class learning process, where I assign one class to each expression. I exploited here SVM as it has shown better generalization capability in Sec. 7.1.1. Table 7.9 shows our recognition rates along with the published results of Lucey et al. [104] for the six basic expressions based on evaluations conducted on the CK+ database. Lucey et al. extracted two types of feature from 68 facial points: similarity-normalized shape (SPTS) and canonical appearance (CAPP) features. They assumed prior-knowledge of the neutral state as well. Due to the lack of training samples, I both employed the LOOCV strategy. We both obtained high recognition rates for the happiness, surprise, and disgust expressions due to their obvious facial deformations. On the other hand, the proposed approach achieved lower recognition rates for anger and fear expressions due to their subtle facial deformations. In contrast with Lucey et al. [104] approach, I achieved higher recognition rate for sadness expression. This improvement is achieved by the use of f_5 and f_6. The subtle expressions suffer from confusions, e.g. anger is confused with sadness and disgust. In summary, I achieved here an average recognition rate of 87.48% compared to 83.15% achieved by Lucey et. al [104], taken into consideration that removing contempt expression from their classification algorithm can lead to an improve in their result. Interestingly, features from eight facial points lead to recognition rates that are superior to or on par with that of 68 points. The results stemmed from the evaluations on the BU-4DFE database are provided in Appendix A.2.2.1

Table 7.9: Confusion matrix facial expression recognition based on LOOCV evaluation conducted on CK+ database using eight points, SVM, and person-specific neutral model. The first row in each expression represents our results. The other row shows the results of Lucy et al. as reported in [104].

	Ha	Su	An	Di	Fe	Sa	Co
Ha	**0.971**	0.00	0.00	0.00	0.029	0.00	-
	1	0.00	0.00	0.00	0.00	0.00	0.00
Su	0.00	**0.987**	0.00	0.00	.013	0.00	-
	0.00	**0.96**	0.00	0.00	0.00	4.00	0.00
An	0.00	0.00	**0. 756**	0.133	0.00	0.111	-
	0.00	0.025	**0.75**	0.075	0.05	0.05	0.05
Di	0.00	0.00	0.084	**0.882**	0.0169	0.0169	-
	0.00	0.00	0.053	**0.947**	0.00	0.00	0.00
Fe	0.16	0.00	0.00	0.00	**0.76**	0.08	-
	0.087	0.13	0.044	0.00	**0.652**	0.00	0.087
Sa	0.00	0.00	0.0357	0.0357	0.0357	**0.893**	-
	0.00	0.04	0.12	0.04	0.04	**0.68**	0.08

Although the distance between (p_{upm}, p_{lom}) the upper and lower mouth facial points is divided into two distances used afterwards to generate features (f_5, f_6), the newly generated features behave differently with each facial expression. For example, pulling down of lip corners can be easily detected with help of (f_5, f_6). This action unit is crucial for the recognition of sadness expression. To evaluate the usefulness of (f_5, f_6), I recalculated the confusion matrices (shown in Tables 7.9 and A.9) using just (f_1, f_2, f_3, f_4) features. I have found that the use of (f_5, f_6) improves the total recognition rate from 83.46% to 87.48% for CK+ database; and from 79.83% to 83.88% for BU-4DFE database. The detailed improvement in the recognition rate of each expression is depicted in Figure 7.7. The use of f_5 and f_6 caused a significant increase in the recognition rate of sadness expression (17.9% for CK+ and 11.9% for BU-4DFE) at cost of small reduction in the recognition rates of other expressions (0.8% in surprise emotion for CK+; and 4.5% in disgust, 2.3% in surprise for BU-4DFE). To illustrate our proposed geometric features in an intuitive way, I depicted the first two principal components of feature vectors in Figure 7.8. Clearly shown, the features are more discriminative for surprise and happiness expressions. On the other hand, a confusion is expected between the



(a) (b)

Figure 7.7: The expression recognition rate with/without f_5, f_6 for the case of geometric-based approach of 8 facial points in the person-specific case. (a) CK+ database. (b) BU-4DFE database.

other expressions, mainly between anger and sadness.

7.2.2.3 General Neutral State

In this section, I evaluate the proposed geometry-based approach of eight facial points without any prior-knowledge of the person-specific neutral state. To cope with that, all the features are calculated with respect to the cropped face box $\mathbf{f}_b = \{x_b, y_b, w_b, h_b\}$, illustrated in Figure 7.9. Obviously seen from Figure 7.10 that the cropping position is invariant to the expressions and has always a fixed distance to the eye centers. This motivates us to exploit the distances from the points to the box corner as features. The cropped faces were resized to a scale of fixed width, precisely 200×200 pixels. Then, from all neutral samples in the training set, I computed the average values of the six distances shown in Figure 7.6. Next, Eq. (7.12) is reformulated as follows.

$$f_i = \frac{d_{i(apex)}}{d_{i(neut_g)}}, \qquad i = 1, ..., 6. \tag{7.14}$$

$d_{i(neut_g)}$ is the average distance of d_i over all training samples of the neutral state. Besides the above six features, the relative location of the eight points within the face patch results in an additional 16-dimensional feature vector, generated from both x- and y- coordinates of each point, as shown below.

$$f_{i+6} = \frac{x_{p_i} - x_b}{w_b}, \qquad i = 1, ..., 8. \tag{7.15}$$

$$f_{i+14} = \frac{y_{p_i} - y_b}{h_b}, \qquad i = 1, ..., 8. \tag{7.16}$$

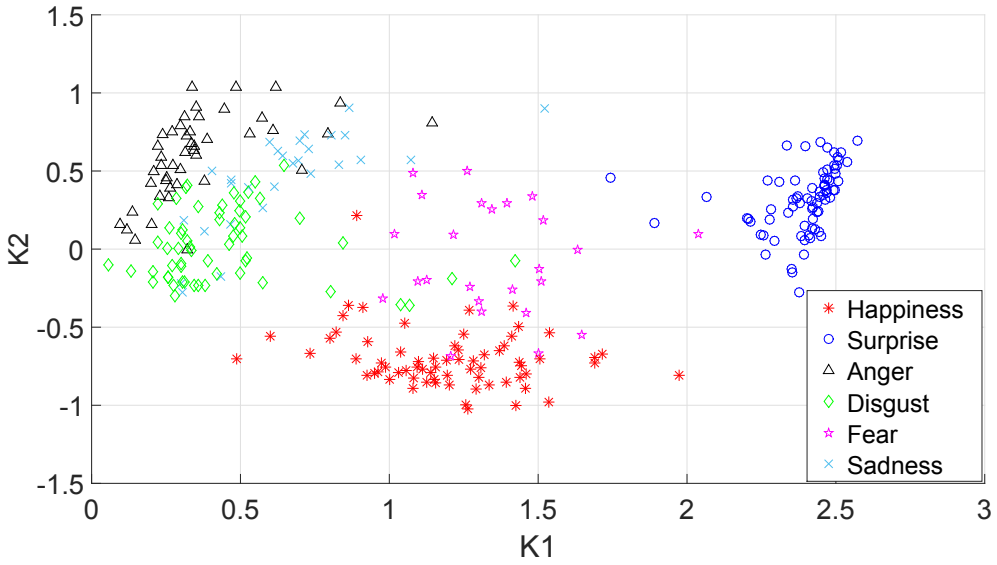


Figure 7.8: The first two principal components of our proposed feature vector for expression recognition using geometric-based approach of 8 facial points in person-specific case; the evaluation was conducted on CK+ database.

The final feature vector is of length 22. I removed the dominant effect of large range features, as in Eq. (7.13), before passing the feature vector to a SVM classifier. The confusion matrix, depicted in Table 7.10, summarizes the results obtained by employing the proposed features in the LOOCV conducted on the CK+ database.

I achieved an average recognition rate of 83.01%, which is comparable to the 83.15% of Lucey et. al [104] (see Table 7.9); however, I used only eight points in person-independent mode. For a comparison purpose, Figure 7.11 shows the recognition rate of each expression for both cases: person-specific and person-independent. The average recognition rate is improved by approximately 6% by exploiting the prior knowledge of the person-specific neutral state. Expressions with subtle facial deformations, such as sadness and fear, are more improved compared to other ones. Happiness and surprise are recognized with higher rate in both cases, and this is most likely due to the distinctive facial deformations they cause, and consequently could be easily measured by our method.

It is not applicable to just classify images into the six basic expressions without automatically recognizing the neutral expression, which is a pitfall of the approaches

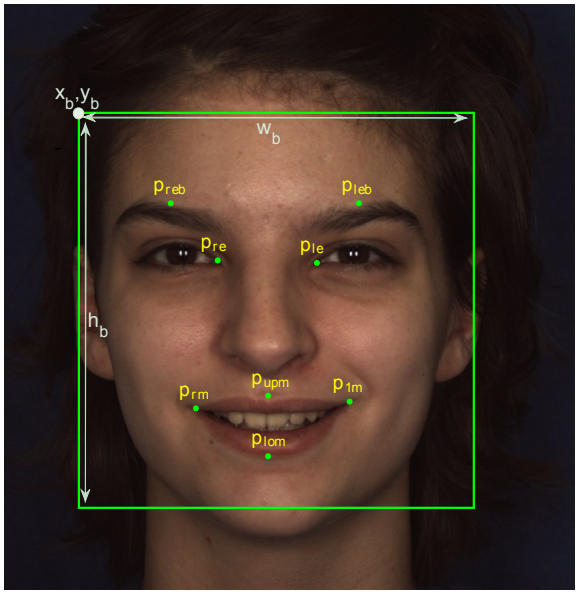


Figure 7.9: A cropped face dimension.

Figure 7.10: The face cropping is invariant to the expression (mouth and eyebrow deformations). A same face expresses four expressions (neutral, happiness, disgust, and surprise), while the cropping has a fixed distance to the eyes center. (Images from Cohn-Kanade database, © Jeffrey Cohn.)

that require prior knowledge of person-specific neutral state. To this end, I dedicated a separate class to the neutral expression. Moreover, I employed two machine learning algorithms (SVM and kNN) for the classification task in the following evaluation.

The LOOCV strategy holds here for the neutral class as well, the samples that participate in building the general neutral model are not involved in testing. A number of points can be drawn from the resulting confusion matrix, depicted in Table 7.11, the happiness and surprise expressions are still recognized with high rates of 98.55%, 98.75% using SVM and 92.75%, 98.75% using kNN, respectively. On the other hand, the perception of the other expressions in particular sadness is confused with neutral; however, the neutral expression is recognized

Table 7.10: Confusion matrix of 6-class facial expression recognition using geometrical features extracted from 8 facial points and a general neutral model, based on cross-validation evaluation on CK+ database via SVM. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**1**	0.00	0.00	0.00	0.00	0.00
Su	0.00	**0.987**	0.00	0.0125	0.00	0.00
An	0.0	0.00	**0.80**	0.0666	0.0222	0.111
Di	0.0	0.00	0.0677	**0.83**	0.0338	0.0677
Fe	0.04	0.00	0.00	0.16	**0.72**	0.08
Sa	0.00	0.00	0.178	0.107	0.0714	**0.642**

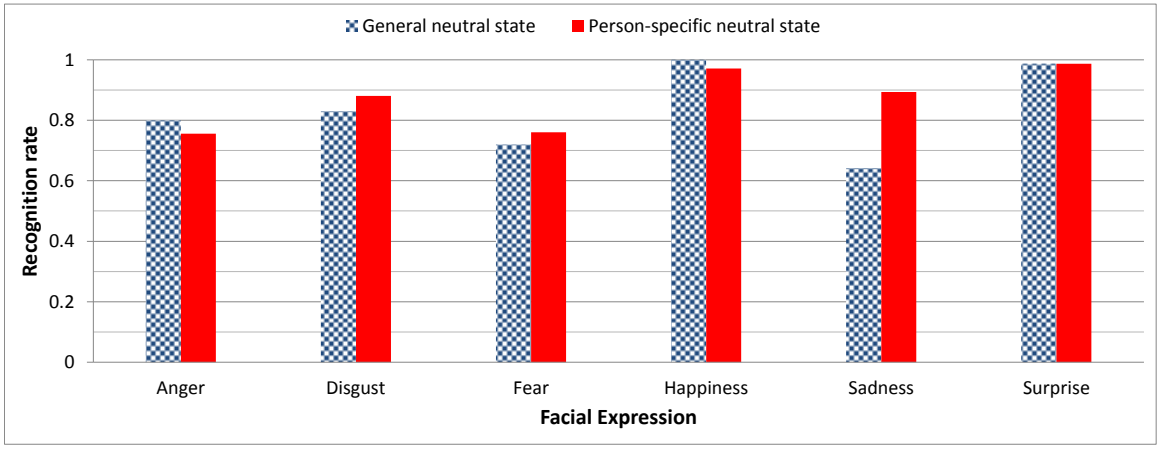


Figure 7.11: Detailed recognition rates for the six facial expressions in both cases: general and person-specific facial expressions. These results are stemmed from applying geometric features extracted from 8 facial points on CK+ database.

Table 7.11: Confusion matrix of 7-class facial expression recognition using the proposed geometric-based approach of 8 facial points exploiting a general neutral model, based on cross-validation evaluation on CK+ database. The first row in each expression represents results using SVM classifier. The other row shows the results using kNN classifier.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.9855**	0.00	0.00	0.00	0.00	0.00	0.0145
	0.9275	0.00	0.00	0.01449	0.04347	0.00	0.01449
Su	0.00	**0.9875**	0.00	0.0125	0.00	0.00	0.00
	0.00	**0.9875**	0.00	0.0125	0.00	0.00	0.00
An	0.00	0.00	**0.6888**	0.02222	0.00	0.06666	0.2222
	0.00	0.00	0.4666	0.08888	0.02222	0.1555	0.2666
Di	0.00	0.00	0.01694	**0.6271**	0.01694	0.01694	0.3220
	0.01694	0	0.08474	0.5593	0.1016	0.01694	0.2203
Fe	0.040	0.00	0.00	0.0400	**0.64**	0.00	0.28
	0.0400	0.00	0.00	0.0800	**0.68**	0.1200	0.08
Sa	0.00	0.00	0.03571	0.00	0.03571	0.3214	**0.6071**
	0.00	0.00	0.00	0.1071	0.1428	0.2857	0.4642
Ne	0.00	0.00507	0.0203	0.04568	0.01015	0.01522	**0.9035**
	0.00	0.01015	0.05076	0.06091	0.03045	0.05583	**0.7918**

with high rate of 90.35% and 79.18% using SVM and kNN, respectively. In summary, I achieved an average recognition rate of 73.63%, 67.12% using SVM and kNN, respectively. These results indicate that SVM classifier outperforms kNN for facial expression recognition. Considering the neutral expression as a separate class implies a lot of confusions with other expressions, especially the subtle ones: fear, anger, and sadness. The evaluation results on BU-4DFE database are summarized in Appendix A.2.2.2.

7.2.2.4 Approach Evaluation with the Developed Facial Point Detector

Locating the eight facial points in the previous two subsections (7.2.2.2,7.2.2.3) involves human intervention either by annotating key frames within each image sequence for CK+ database, or by selecting frames of neutral expression to detect the facial points and track them afterwards for BU-4DFE database. Therefore, these approaches cannot run fully automatically.

Table 7.12: Confusion matrix of 6-class facial expression recognition using the proposed geometric-based approach of eight facial points, those points were detected using a point detector, developed here in Ch. 5. The cross-validation was conducted on Ck+ database. The first row in each expression summarizes the results in the person-specific case, while the other row in the person-independent case.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.9855**	0	0	0	0.0145	0
	0.9710	0	0	0.0290	0	0
Su	0	**0.9878**	0	0	0	0.0122
	0	**0.9878**	0	0	0	0.0122
An	0	0	**0.7556**	0.1333	0.0222	0.0889
	0	0	**0.6444**	0.2444	0	0.1111
Di	0	0	0.0847	**0.8644**	0.0169	0.0339
	0	0	0.1356	**0.8475**	0	0.0169
Fe	0.1600	0.0400	0.0400	0.0400	**0.7200**	0
	0.0800	0	0.0800	0	**0.8400**	0
Sa	0	0	0.2857	0.0714	0.0357	**0.6071**
	0	0	0.2857	0.1429	0.0357	0.5357

Obviously, the performance of a geometry-based approach relies heavily on the point detector accuracy [141]. Expressions of subtle deformations, e.g. sadness and anger, are more exposed to confusions under the point localization error. I dedicated this section to investigate the performance of the proposed method when the facial points are located via our point detector (Sec. 5). Here, I utilize only the 8 points from the detected 49 facial points. The CK+ and BU-4DFE databases were not involved in the training of our point detector. For the neutral frames, I also detected the facial points automatically using our detector, and then utilizing those detections for the feature extraction. The confusion matrices, depicted in Table 7.12, summarize the results obtained by applying LOOCV of the method of 8 facial points (Sec. 7.2.2) on the CK+ database. SVM was employed to assign the extracted features to the corresponding expression.

The comparison of the these results to those in Tables 7.9 and 7.10 is concluded as follows. As the point localization here is less accurate than the annotated data, the average recognition rate drops from 87.48% to 82.01% in the person-specific

Table 7.13: Confusion matrix of 7-class facial expression recognition using the proposed geometric-based approach of eight facial points in person independent mode, those points were detected using a point detector, developed here in Ch. 5. The cross-validation was conducted on Ck+ database.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.9855**	0	0	0	0	0	0.0145
Su	0	**0.9878**	0	0	0	0.0122	0
An	0	0	0.5778	0.1778	0	0.0444	0.2000
Di	0.0169	0	0.0847	**0.7627**	0	0	0.1356
Fe	0.1200	0.0400	0	0	**0.7200**	0	0.1200
Sa	0	0	0.0714	0.0357	0	0.4643	0.4286
Ne	0.0145	0	0.1159	0.1159	0.0145	0.0435	**0.6957**

case and from 83.01% to 80.44% in the general neutral case. Expressions of subtle deformations, sadness and anger, are confused more with each other in comparison to other expressions such as happiness and surprise. The drop in the general neutral case is less than in the person-specific case, since I automatically detected the points in both frames (neutral and apex). Table 7.13 summarizes the case of 7-class, where I automatically detect the neutral state as well. With the developed point detector, I achieved a recognition rate of 74.2%, which is as good as that using the annotated data 73.63%. Sadness, anger, and disgust expressions are confused with the added neutral class, while happiness and surprise are recognized with higher rates. I summarize the evaluations on the BU-4DFE database in Appendix A.2.2.3.

Two facts stemmed from the above evaluations. First, by personalizing the approaches, the recognition rate is improved. Second, our point detector is accurate enough to be used for the expression recognition.

7.2.3 Discussion

I have proposed geometry-based approaches to recognize the six basic expressions, and in some cases along with the neutral state. The proposed methods were evaluated under the absence and existence of prior knowledge of person-specific neutral state. As expected, personalized methods always outperform the general ones. With 49 facial points, I am able to recognize the six expressions with an average

rate of 89.72%, as measured on the CK+ database in person-specific case. This rate drops to 86.52% by removing the personal dependency, to 82.01% by using only 8 points of the 49 points, and to 80.44% by using 8 points and removing the personal dependency as well. These rates correspond to full automatic scenario, where our point detector was employed. The proposed features have shown a great effectiveness, e.g. with 8 facial points using general neutral state I achieved an average recognition rate of 83.01% that is as good as the rate obtained by the approach of [104] (83.15%), in which 68 points are employed in person-specific mode. Although 16% (8/49) of the facial points were employed, the average recognition rate dropped by only 10.82% ($\frac{89.72-80.01}{89.72} \times 100$) in the person-specific case and 7.03% in the person-independent case.

7.3 Joint Facial Expression Recognition and Point Localization

The recognition of the facial expression and the localization of the facial points have been sequentially tackled so far, in which a point detector is trained across facial expressions and employed later for the expression recognition either in a frame or video. The main shortcoming here is that increasing the variations within a training data has a negative effect on the detection accuracy. Another possible solution is to build many point detectors each for one facial expression in analogy to the method used by [114] to recognize the facial expression across poses. Using the latter sequential method, an error in the expression recognition stage would lead to an enormous error in the point detection stage. The optimal solution would be to tackle these two tasks simultaneously rather than sequentially. Exploiting a joint estimation has been proposed earlier for face detection and alignment [29]; pose and face landmarks [91]; face detection, pose, and landmarks [185].

In this section, I introduce a framework that jointly locates eight facial points and recognizes the seven facial expressions (six basic expressions plus the neutral state) on a frame basis. This framework incorporates the state-of-the-art techniques that have been employed to address the two tasks separately, such as the geometry- and appearance- based methods for the facial expression recognition, and the cascade regression and local-based methods for the facial point detection. I adapted the Viterbi algorithm to perform the data fusion from four models; the importance

of each model is taken into account as well.

I consider the eight facial points, depicted in Figure 7.5. After detecting the face via the VJ detector, I ensure the proper and consistent face cropping using the method developed in Sec. 5.1. Next, I propose models to measure the local response of each potential point location under a hypothesis of specific facial expression. Those responses are evaluated further using an appearance-based expression recognizer and a cascade-regression point detector. The final decision about the point location and the facial expression is made by an adapted version of the Viterbi algorithm.

7.3.1 Developed Models for both: Facial Expressions and Points

I evaluate each potential location for each facial point using four models, two trained across the facial expressions and the other two per expression. Configural and shape features are important bases for robust facial analysis approaches [107]. Those features encode the relative locations of the facial components, defined here as the geometry-based features. To this end, the following two models were developed. Let $\mathbf{p}$ denote a potential set of the eight facial points, as numbered in Figure 7.12.

$$\mathbf{p} = \{\mathbf{p}_1, ..., \mathbf{p}_8\}, \quad \mathbf{p}_i \in \mathbb{R}^{2\times 1}. \tag{7.17}$$

The first model was designed to evaluate each $\mathbf{p}$ according to its distance to the expression-specific prior location $\mathbf{pp}$, the mean location across the training data.

$$p(\mathbf{p}|\mathbf{pp}, c) = p(\mathbf{p}|\Phi_{\mathbf{pp}_c}) = \sum_{i=1}^{m} \alpha_i p_i(\mathbf{p}|\phi_i), \tag{7.18}$$

$\phi_i = (\mu_i, \Sigma_i)$, $\Phi_{\mathbf{pp}_c} = (\alpha_1, ..., \alpha_m, \phi_1, ..., \phi_m)$ is the points prior GMM model for the facial expression c, which is estimated via the Expectation Maximization (EM) algorithm. Each p_i is a 16-dimensional multivariate Gaussian distribution given by

$$p_i(\mathbf{p}|\phi_i) = \frac{1}{(2\pi)^{\frac{16}{2}}|\Sigma_i|^{\frac{1}{2}}} \exp\{-\frac{1}{2}(\mathbf{p}-\mu_i)^{\mathrm{T}}\Sigma_i^{-1}(\mathbf{p}-\mu_i)\}, \tag{7.19}$$

$\mu_i \in \mathbb{R}^{16\times 1}$ is the mean vector of the i^{th} subpopulation; where Σ_i is its 16×16 covariance matrix. $\alpha_i \in [0, 1]$ for all i and the α_i's are constrained to sum to one.

Next, a cascade-regression approach was built to locate only the eight facial points. Unlike the SDM used in [170], I exploited here the SVR for the non linear

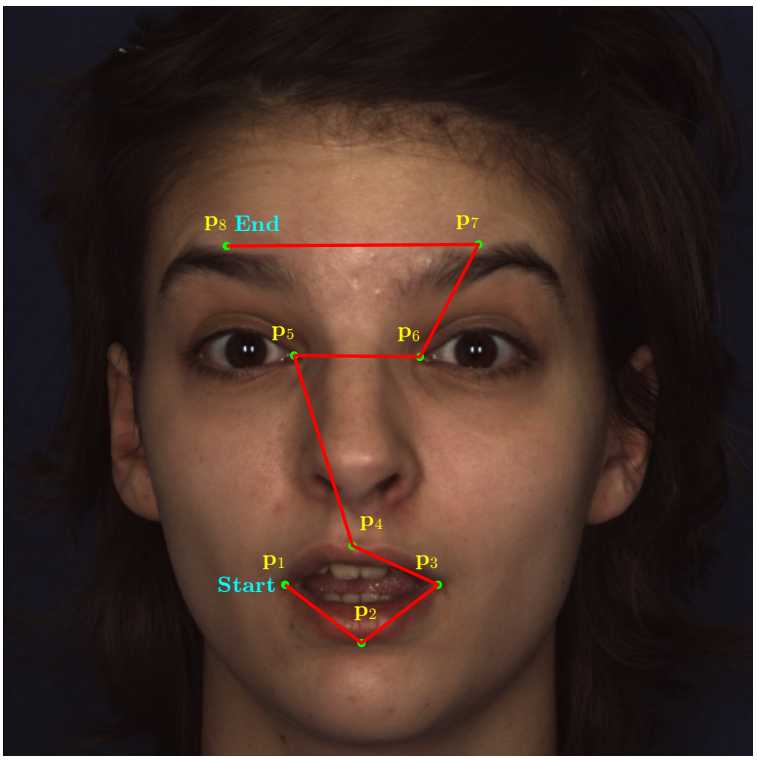


Figure 7.12: The relative location of the eight facial points is measured via two-point models in the depicted sequence.

mapping of the features to the point location. The training was performed across the facial expressions. Each set of points $\mathbf{p}$ was then evaluated with respect to the set $\mathbf{pr}$ that was detected via the cascade-regression approach as follows.

$$p(\mathbf{p}|\mathbf{pr}) = p(\mathbf{p}|\Phi_{\mathbf{pr}}) \tag{7.20}$$

$\Phi_{\mathbf{pr}}$ is a GMM, whose mean is the detected points, and the identity matrix is its covariance.

The facial wrinkles, bulges, and furrows carry significant cues about both the facial expression and the point location, which can be inferred using appearance-based features. To this end, I built the next two models. I constructed a model to evaluate the texture around each potential point $\mathbf{I_p}$ according to its distance to the ground truth location. In particular, HoG features were extracted from patches, each of size 20% of the cropped face size, those patches surround the potential points. This model ψ was trained per expression c using SVR, where the likelihood of each set is calculated as follows.

$$p(\mathbf{p}|\mathbf{I_p}, c) \propto \psi(\mathbf{I_p}, c). \tag{7.21}$$

The fourth model Ψ was built to estimate the probability of a facial expression c given texture features extracted from the entire face patch ($\mathbf{I}$) through SVM classifier [28] as follows.

$$p(c|\mathbf{I}) \propto \Psi(\mathbf{I}, c), \tag{7.22}$$

Throughout this section, HoG descriptor was used to encode the face appearance as it is one of the most effective descriptors for the texture. The cropped face is scaled to a fixed size before applying the aforementioned models. For the fourth model, the face was scaled into 160×160 pixels, and then divided into cells of 20×20 pixels. Next, I constructed an eight-bin orientation histogram for each cell, in which each pixel orientation was weighted by its magnitude. Normalized histograms over block regions were concatenated to form the final feature vector of length 1568. I used a block of 2×2 cells, and a block stride of one cell, same setup as in Sec. 7.1.3.

The final task is to jointly recognize the facial expression (c^*) and locate the facial points ($\mathbf{p}^*$) as follows.

$$\mathbf{p}^*, c^* = \quad \arg\max_{\mathbf{p},c} p^{n_1}(\mathbf{p}|\mathbf{pp}, c) p^{n_2}(\mathbf{p}|\mathbf{pr}) p^{n_3}(\mathbf{p}|\mathbf{I}_\mathbf{p}, c) p^{n_4}(c|\mathbf{I}).$$

$n_1, ..., n_4$ assign the importance of each model in the joint inference. All the parameters (model importance, of SVC, SVR, HoG) were estimated using a grid-search along with cross-validation experiments conducted on the training set aiming more accurate point localization and higher expression recognition rate at reasonable processing and resource cost.

7.3.2 Data Fusion

It is computationally expensive to evaluate all possible combinations of the eight facial points from their potential locations. Seeking more efficient system, I evaluated the potential locations of each facial point independently (not in set). The relative relation of the points was ignored in Eqs. 7.20, and 7.21, where I reformulated them under the assumption of point independence as follows.

$$p(\mathbf{p}|\mathbf{pr}) \approx \prod_{i=1}^{8} p(\mathbf{p}_i|\mathbf{pr}_i) \tag{7.23}$$

$$p(\mathbf{p}|\mathbf{I}_{\mathbf{p}}, c) \approx \prod_{i=1}^{8} p(\mathbf{p}_i|\mathbf{I}_{\mathbf{p}_i}, c). \tag{7.24}$$

The model in Eq. (7.18) was reformulated as well, but in a way preserving the relative location of the facial points.

$$p(\mathbf{p}|\mathbf{pp}, c) \approx \prod_{i=1}^{8} p(\mathbf{p}_i|\mathbf{pp}_i, c) \prod_{k=2}^{8} p(\mathbf{p}_k|\mathbf{p}_{k-1}, c) \tag{7.25}$$

$p(\mathbf{p}_i|\mathbf{pp}_i, c)$ is the likelihood of the potential location ($\mathbf{p}_i$) given the corresponding prior location hypothesized the facial expression is c. $p(\mathbf{p}_i|\mathbf{p}_{i-1}, c)$ evaluates the relative location of $\mathbf{p}_i$ and $\mathbf{p}_{i-1}$ hypothesized the facial expression is c; the sequential order of the points is illustrated in Figure 7.12. All the aforementioned point-wise models are GMMs.

Above I prepared each potential point set to be evaluated in a sequence, in what follows I adapt the Viterbi algorithm to perform this evaluation efficiently. For a hidden Markov model (HMM) of N states, characterized by initial probabilities, stationary transition matrix, the Viterbi approach is used to find the most likely state sequence that produces a given observation sequence of length T, where the computation complexity degrades from $O(TN^T)$ to $O(TN^2)$. Following the same idea, I built seven networks, each for one facial expression. Every model has a sequence of length eight steps, each for one facial point where the points arrangement is depicted in Figure 7.12. Each step contains a variable number of states representing the probable locations of the corresponding point $N_{\mathbf{p}_s}$, e.g. $\mathbf{p}_{si}$ is the potential location i of the point $\mathbf{p}_s$. The evaluation of both the states (point location) and the transition between the consecutive steps was carried out with respect to the aforementioned four models.

At step 1, each potential location is locally evaluated using

$$(p^{n_1}(\mathbf{p}_1|\mathbf{pp}_1, c)p^{n_2}(\mathbf{p}_1|\mathbf{pr}_1)p^{n_3}(\mathbf{p}_1|\mathbf{I}_{\mathbf{p}_1}, c),$$

where the response is stored as a metric value corresponding to this location. Starting from step two, each state bonds with a state from the earlier stage that has the maximum value resulting from multiplying the state metric by $p^{n_1}(\mathbf{p}_i|\mathbf{p}_{i-1}, c)$, $i > 1$. The result of multiplying this maximum value by the local response of the underlying state is considered as its metric value. Each state bonds with only one

from the previous step and varying number from the next step. In the last step, I choose the state with the maximum metric and multiply it by $p^{n_4}(c|\mathbf{I})$ to produce the network response. Finally, I consider the maximum response across the networks our join estimation. The winner network belongs to the recognized facial expression, and recursively from the maximum metric of its last step I get the location of the facial points. This entire fusion process is summarized in Algorithm 4.

7.3.3 Evaluations

I investigated the effectiveness of the proposed method on the two facial expression databases (CK+ and BU-4DFE). Table 7.14 summarizes the evaluations on CK+ database in terms of confusion matrices. First row shows results obtained using the geometric and local models (Ge-Lo) of Eqs. (7.23), (7.24), (7.25). In the second row, you can see the results obtained using the holistic-texture model (Holi-Tex) of Eq. (7.22). Last row was dedicated for the results obtained by them all via Algorithm 4. The average recognition rate of the facial expression via Ge-Lo (67.57%) is lower than that via Holi-Tex (83.71%), as Ge-Lo method is influenced mainly by the point localization error besides the small number of points exploited here. These results agree with an analysis conducted by [141] which states that recognition rates of a geometry-based approach degrades dramatically as the point localization uncertainty increases. Exploiting both geometry- and appearance-based methods, the proposed approach enhances the overall recognition rate by approximately 5.43% (89.14%), where the recognition of the sadness expression has been improved the most. Adding the neutral state as a separate expression class caused the most confusions especially with those of subtle deformations (sadness and anger). The achieved improvement to the holistic approach can be attributed to minimizing the individual variations within the same class using the local-based texture features.

To evaluate the facial point detector, the error of each estimated point is calculated by Eq. (5.10). The obtained results via the joint framework along with results acquired only using the cascade regression model are depicted in Figure 7.13. Clearly shown that the proposed framework leads to an improvement in the points' localization, where the averaged error over all points decreased from 0.0163 to 0.0146

Algorithm 4: The data fusion method. An adapted Viterbi algorithm to jointly locate eight facial points and recognize the corresponding facial expression. $p(\mathbf{p}_{si}|\mathbf{pp}_s, c)$ evaluates the location of the candidate point i for facial point s with respect to expression-specific point prior location, $p(\mathbf{p}_{si}|\mathbf{pr}_s)$ with respect to the estimated location via the cascade regression method, $p(\mathbf{p}_{si}|\mathbf{I}_{\mathbf{p}_s}, c)$ with respect to expression-specific surrounding texture, $p(\mathbf{p}_{si}|\mathbf{p}_{(s-1)k}, c))$ with respect to expression-specific location of candidate point k for facial point $s-1$. $N_{\mathbf{p}_s}$ is the number of the potential points for facial point s.

Data: The face patch $\mathbf{I}$.
Result: $\mathbf{p}^*, c^*$
begin
  for $c \longleftarrow 1$ **to** 7 **do**
    for $i \longleftarrow 1$ **to** $N_{\mathbf{p}_1}$ **do**
      $\mathbf{M}_c[i,1] \longleftarrow p^{n_1}(\mathbf{p}_{1i}|\mathbf{pp}_1, c)p^{n_2}(\mathbf{p}_{1i}|\mathbf{pr}_1)p^{n_3}(\mathbf{p}_{1i}|\mathbf{I}_{\mathbf{p}_1}, c)$
      $\mathbf{Id}_c[i,1] \longleftarrow 0$
    for $s \longleftarrow 2$ **to** 8 **do**
      for $i \longleftarrow 1$ **to** $N_{\mathbf{p}_s}$ **do**
        $\mathbf{M}_c[i,s] \longleftarrow \max_k(\mathbf{M}_c[k,(s-1)]p^{n_1}(\mathbf{p}_{si}|\mathbf{p}_{(s-1)k}, c))$
        $\mathbf{Id}_c[i,s] \longleftarrow \arg\max_k(\mathbf{M}_c[k,(s-1)]p^{n_1}(\mathbf{p}_{si}|\mathbf{p}_{(s-1)k}, c))$
        $\mathbf{M}_c[i,s] \longleftarrow \mathbf{M}_c[i,s] \times p^{n_1}(\mathbf{p}_{si}|\mathbf{pp}_s, c)p^{n_2}(\mathbf{p}_{si}|\mathbf{pr}_s)p^{n_3}(\mathbf{p}_{si}|\mathbf{I}_{\mathbf{p}_s}, c)$
    $\mathbf{Net}_c \longleftarrow \max_k \mathbf{M}_c[k,8] \times p^{n_4}(c|\mathbf{I})$
  $c^* \longleftarrow \arg\max_c \mathbf{Net}_c$
  $z_8 \longleftarrow \arg\max_k \mathbf{M}_{c^*}[k,8]$
  $\mathbf{p}_8^* \longleftarrow \mathbf{p}_{8z_8}$
  for $s \longleftarrow 8$ **to** 2 **do**
    $z_{s-1} \longleftarrow \mathbf{Id}_{c^*}[z_s, s]$
    $\mathbf{p}_{s-1}^* \longleftarrow \mathbf{p}_{s-1z_{s-1}}$

Table 7.14: Confusion matrix of the facial expression recognition, obtained using a cross-validation evaluation conducted on the CK+ database: first row presents the results obtained using the Ge-Lo models (Eq. 7.23,7.24,7.25), second row using Holi-Tex model (Eq. 7.22), third row using all models (the joint frame work). Each row of the confusion matrix represents a ground truth class, and the values in the row correspond to the classification result.

	Predicted						
	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.74**	0	0.09	0.04	0.09	0	0.04
	1.00	0	0	0	0	0	0
	1.00	0	0	0	0	0	0
Su	0	**0.96**	0	0	0	0	0.04
	0	**0.89**	0	0	0	0	0.11
	0	**0.96**	0	0	0	0	0.04
An	0	0	**0.60**	0.20	0.07	0	0.13
	0	0	**0.93**	0	0	0.07	0
	0	0	**1.00**	0	0	0	0
Di	0	0	0.05	**0.65**	0.10	0.05	0.15
	0	0	0	**1.00**	0	0	0
	0	0	0.05	**0.95**	0	0	0
Fe	0	0	0.22	0	**0.67**	0	0.11
	0.11	0	0	0	**0.67**	0	0.22
	0	0	0.11	0	**0.78**	0	0.11
Sa	0	0	0.10	0.10	0	**0.60**	0.20
	0	0	0.10	0	0	0.50	0.40
	0	0	0	0	0	**0.70**	0.30
Ne	0	0.03	0.26	0.10	0.05	0.05	0.51
	0	0	0.10	0.03	0	0	**0.87**
	0	0	0.10	0.00	0	0.05	**0.85**

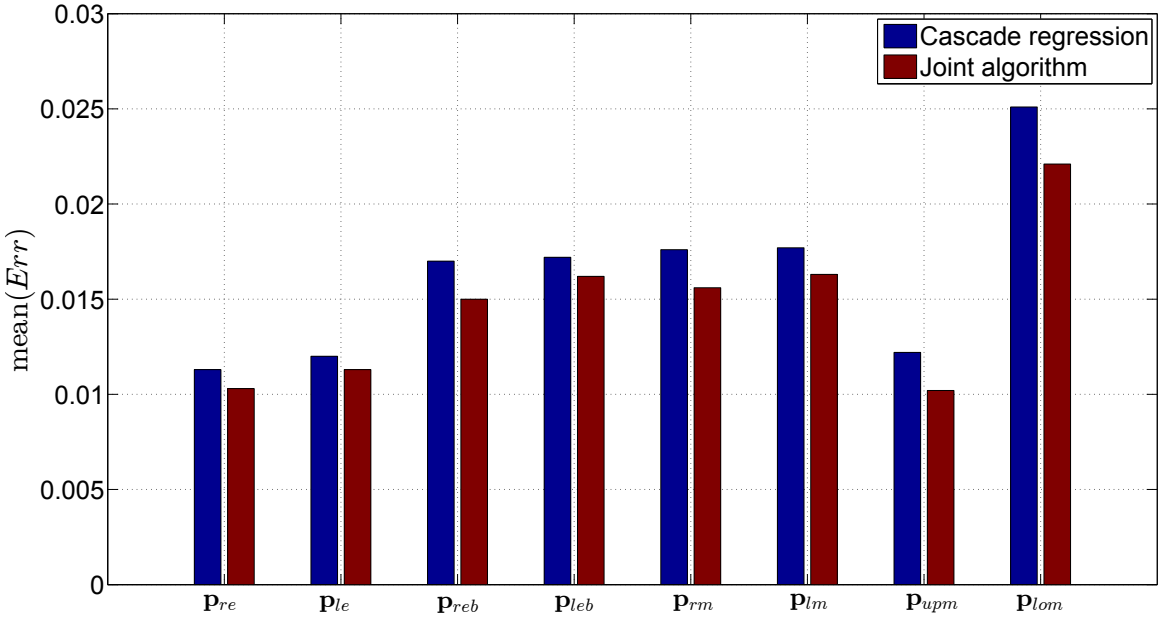


Figure 7.13: The mean of the localization error for each facial point, stemmed from the evaluation conducted on the CK+ database. In blue bars, the results using only the cascade-regression method are presented, and in dark red bars the results using the proposed fusion framework.

(10.4%). This improvement is attributed to the expression-specific models that were exploited in the fusion algorithm.

The evaluations on BU-4DFE database are summarized in Figure 7.14 for the point detection and in Table 7.15 for the expression recognition. Complying with the evaluation on CK+ database, the proposed approach gave a rise in both the expression recognition rate and the point detection accuracy. The proposed approach scored an average recognition rate of 74%, compared to 64.14%, 67.71% using GeLo, Holi-Tex, respectively. Meanwhile, the geometric approach of [141] reported an average expression recognition rate of 68.04% on this database. Additionally, the facial points are localized here with an average error of 0.0129 compared to 0.0142 using the cascade-regression method (improved by $8,91\%$). It is noticeable that the expression recognition rate on this database is lower than that on CK+, which is caused by the higher variation of the expression intensity among the apex frames of the same class in comparison to CK+ database. On the other hand, I achieved a better accuracy for the point localization due to the higher resolution of the images provided by BU-4DFE database.

Table 7.15: Confusion matrix of the facial expression recognition, obtained using a cross-validation evaluation conducted on the BU-4DFE database: first row presents the results obtained using the Ge-Lo models (Eq. 7.23,7.24,7.25), second row using Holi-Tex model (Eq. 7.22), third row using all models. Each row of the confusion matrix represents a ground truth class, and the values in the row correspond to the classification result.

	Predicted						
	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.89**	0.04	0	0	0.07	0	0
	1.00	0	0	0	0	0	0
	1.00	0	0	0	0	0	0
Su	0	**0.79**	0	0.03	0.11	0	0.07
	0	**0.72**	0	0	0.17	0	0.11
	0	**0.86**	0	0.09	0.05	0	0
An	0	0	**0.61**	0.07	0.07	0	0.25
	0	0	0.54	0.17	0	0.17	0.12
	0	0	**0.65**	0.13	0	0.11	0.11
Di	0.05	0	0.05	0.55	0.20	0.15	0
	0.05	0	0.17	0.53	0.10	0.10	0.05
	0.05	0	0.13	0.55	0.11	0.11	0.05
Fe	0.15	0	0.25	0	0.49	0	0.11
	0.10	0	0	0	0.50	0.15	0.25
	0.10	0	0	0	0.57	0.12	0.21
Sa	0	0.06	0.10	0.10	0	0.58	0.16
	0	0	0.10	0	0.05	0.57	0.28
	0	0	0.10	0	0.03	**0.66**	0.21
Ne	0	0.03	0.0	0.10	0.09	0.20	0.58
	0	0	0.04	0	0	0.08	**0.88**
	0	0.01	0.0	0.03	0	0.07	**0.89**

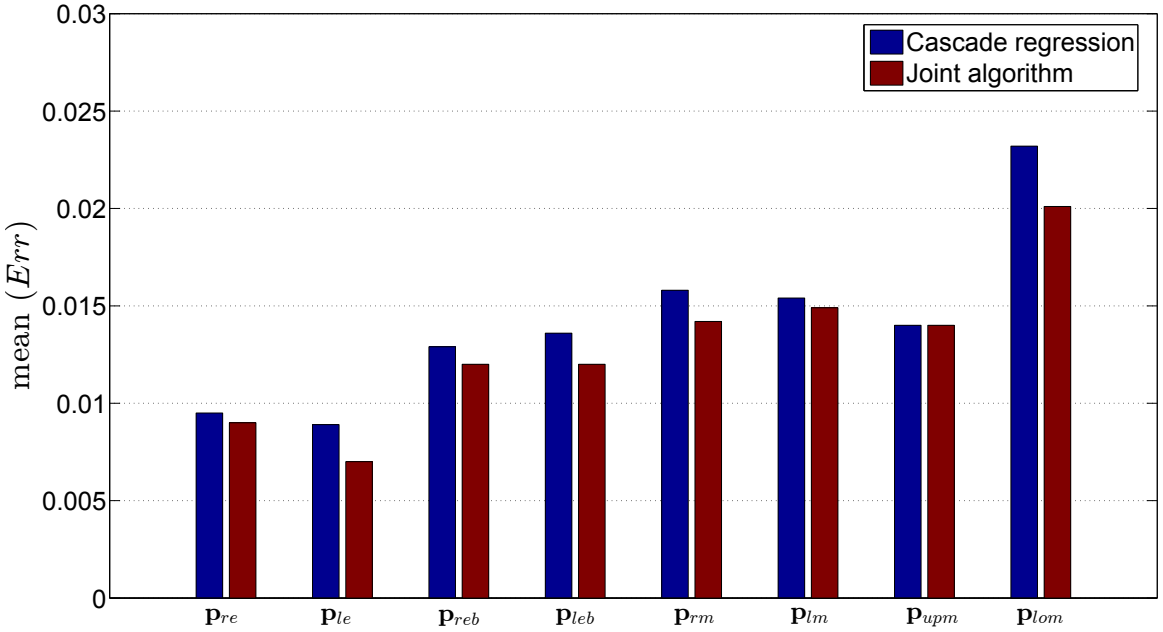


Figure 7.14: The mean of the localization error for each facial point, stemmed from the evaluation on the BU-4DFE database. In blue bars, the results using only the cascade-regression method are presented, and in dark red bars the results using the proposed fusion framework.

7.3.4 Discussion

I have presented a framework for joint facial expression recognition and point localization. For this framework, I considered eight facial points and seven facial expressions (happiness, surprise, anger, disgust, fear, sadness, neutral). The developed method was built on top of state-of-the-art methods: cascade regression for the point detection, appearance- and geometry-based methods for the facial expression recognition. Clearly shown from the conducted evaluations that I successfully enhanced the expression recognition rate by approximately 5%, and the point localization by 9% in comparison to the conventional separated methods.

7.4 Discussion

In this chapter, I have proposed many approaches to automatically recognize seven facial expressions on a frame basis. Our methods start by locating the face within the frame, then refining the cropped patch, next extracting representative features which are mapped finally to a corresponding expression via a machine learning

classifier-based method.

Utilizing the detected 49 facial points in Ch. 5, I developed a geometry-based approach that achieved recognition rates of 89.72% and 86.52% for person-dependent and person-independent scenarios of 6-class, as evaluations conducted on the CK+ database showed. Our point detector was not trained on the CK+ database. In the case of person-independent, the recognition rate dropped to 81.89% when I performed 7-class evaluation by adding the neutral state as a separate category. By conducting those evaluations on the BU-4DFE database, I achieved recognition rates of 83.44%, 76.29%, and 71% for 6-class person-dependent, 6-class person-independent, and 7-class person-independent cases, respectively. It is reasonable to achieve lower rates on BU-4DFE database than on CK+ database as BU-4DFE is more challenging by its inconsistent expression intensity.

Using only 8 out of the 49 facial points, I have proposed an approach that achieved recognition rates of 87.48%, 83.01%, and 73.63% for 6-class person-dependent, 6-class person independent, and 7-class person-independent cases, respectively, on the CK+ database. On the BU-4DFE database, I achieved recognition rates of 83.88%, and 68.04% for 6-class person-dependent, and 7-class person-independent cases, respectively. Although only $16.3\%(\frac{8}{49})$ of the facial points were utilized a maximum drop of 8.26% in the recognition rate was experienced. A practical approach, usually for an industrial purpose, has to be a trade-off accuracy and efficiency. I have highlighted a direction to build an efficient approach for facial expression recognition utilizing only 8 facial points, which are corner and edge points.

An appearance-based approach has been proposed as well. Combinations of one appearance-based feature type and one classifier type were empirically investigated. The best recognition rate was achieved by the employment of HoG features with SVM classifier. I obtained recognition rates of 87.26% and 83.71% for the cases of 6-class, and 7-class, respectively, on CK+ database (here all person-independent). On the BU-4DFE database, I achieved recognition rates of 77.14% and 67.71% for 6-class and 7-class cases, respectively. The drop in the recognition rate for the 7-class case in comparison to 6-class case is reasonable as the new class increases the confusions. BU-4DFE database incorporates samples of each expression of a varied intensity resulting in a lower recognition rate in comparison to CK+ database.

Finally, I have proposed a framework for joint facial expression recognition and

point localization, in which a hybrid of geometry and appearance based features is used for the expression recognition, and a hybrid of cascade-regression and local-based methods is used for the point localization. The approach was configured to recognize the 7 expressions and locate the 8 facial points. The use of it leads to improve the expression recognition rate by at least 5.43% and to improve the localization accuracy by at least 8.91%.

Building a fair comparison with state-of-the-art approaches is difficult due to the different experiment protocols. The approach here is fully automatic, while other approaches utilize a manually face cropping, or point overlying. Many approaches provide video-based decision instead of a frame-based. In many cases, partitioning the database into training and testing differs from each other. The average recognition rates of approaches with almost similar protocol are summarized in Table 7.16, highlighting the superior performance of the proposed approaches.

Table 7.16: The average recognition rates (RR (%)) of approaches that use a similar evaluation protocol to the one used here. # C denotes the number of classes.

Approach	RR (%)	# C	Database	**Note**
Lucey et al. [104]	83.32	7	CK+	they use 68 facial points in person dependent methods, (contempt used instead of neutral)
Zavaschi et al. [176]	71.12	7	CK+	calculated from thier reported confusion matrix using all classifiers
Zavaschi et al. [176]	82.23	7	CK+	calculated from thier reported confusion matrix using Ensemble.
Chew et al. [30]	74.4	7	CK+	calculated from thier reported confusion matrix (contempt instead of neutral).
Zhong et al. [184]	86.31	6	CK+	calculated from their reported confusion matrix (only using Common Patches), the face is optimally cropped.
Zhong et al. [184]	88.25	6	CK+	calculated from their reported confusion matrix (using Common and Specific Patches), the face is optimally cropped.
Amor et al. [8]	81.9	6	BU-4DFE	calculated from their reported confusion matrix using FREE-FORM DEFORMATION HMM CLASSIFIERS, the annotated face is utilized.
Littlewort et al. [98]	81.9	6	BU-4DFE	by applying CERT software on the database.

CHAPTER 8

Conclusions and Future Perspectives

The facial analysis based on a camera has received a lot of attention due to its non-intrusive nature, where accordingly applications ranging from entertainment to serious security systems have been being developed. In this dissertation, I have proposed methods to automatically locate facial points, estimate the head pose, and to recognize seven facial expressions.

To locate the facial points, I have proposed a cascade-regression method, in which **MLP** is exploited for the non-linear mapping of the appearance features to the ground truth location. This method was further enhanced by the use of a guided point initialization instead of the ordinary one with the mean point location, and by performing a feature selection at each iteration. The proposed approach has been comprehensively evaluated in both within and cross- database scenarios. A comparison with state-of-the-are approaches and commercial software packages in terms of accuracy and efficiency was presented. Besides its competitive accuracy, the proposed approach was one of the fastest methods in locating the points, with better generalization capability.

To estimate the head pose of a face depicted in RGBD images, I exploited several appearance- and depth- based features to encode the varying face appearance across head poses. These features were then mapped to the head pose angles (pitch, yaw, roll) via a regression-based method. I further refined the output of the

face detector, exploiting the depth data in some cases, ending in more consistent crops and accordingly more accurate estimation. To encode the facial appearance, I adapted three appearance-based feature types and introduced new depth-based features, where a fair comparison between them in terms of accuracy and efficiency was presented. Superior performance was achieved using the proposed method of a concatenated vector of different feature types. Meanwhile, the newly introduced depth-based features provide competitive results in lower computation time. My approach is qualified to work with ordinary RGB cameras as an RGB-based cropping refinement method was proposed as well. The effectiveness of the proposed method was assessed via conducting within and cross database evaluations, involving comparisons with state-of-the-art approaches. The results highlight the competitiveness of the proposed approach and its better generalization capabilities.

To recognize seven facial expressions, I proposed geometry and appearance -based methods. In the geometry-based method, I made use of the 49 facial points obtained via the point detector that was developed in this thesis as well. Personalized methods outperform the general ones by only 3% on average. A geometry-based method of 8 facial points was proposed as well. Although only 16% of the facial points are used, the drop in the average recognition rate does not exceed 10%. With respect to the appearance-based approach, different configurations of one appearance-based descriptor and one classifier type were investigated. With HoG features and SVM classifier, I achieved the highest recognition rates. In the Appendix B, the proposed geometry and appearance -based approaches were assessed regrading their generalization capability via cross-database evaluations. The geometry-based method generalizes better across the databases, especially when the database of greater variance in the expression intensity was employed for the training.

Finally, I have proposed a framework for joint facial expression recognition and point localization, in which both tasks were advanced in comparison of using them in the ordinary sequence. This framework makes use of the state-of-the-art techniques that have been employed to address the two tasks separately, the geometry- and appearance- based methods for the facial expression recognition, and the cascade regression and local-based methods for the facial point detection. To speed

up the data fusion, Viterbi algorithm was adapted.

The research is a self-evolving process, by its nature. In what follows, a summary of further potential directions within the scope of the presented methods.

- Regarding the **facial point detection** method, the proposed approach was an RGB based, which can be improved by exploiting depth data, consequently extending it to be RGBD based. This option shall improve its robustness for those specific sensors, which are widely spread nowadays.
- Regarding the **head pose estimation** method, there is still a space to improve the approach performance, especially when the head rotates about two axes simultaneously. To this end, a new database should be created and labeled. Using the point location to estimate the pose would enhance the pose estimator robustness. Another further direction is employing the proposed approach as a basis to build an approach for head gesture recognition or to build a pose-invariant approach for the facial expression recognition.
- Regarding the **facial expression recognition** methods, I have proposed frame-based methods. A next logical step is to aggregate these decisions in a clever way to provide a video-based decision.
- I proposed **a framework to joint estimate seven facial expressions and locate eight facial points**. This framework can be extended to involve more facial points and to jointly estimate the head pose as well. With respect to the individual performance of each approach, the efficiency of the framework can be improved.

APPENDIX A

The evaluations of the proposed methods for facial expression recognition on the BU-4DFE database.

This chapter is dedicated to present the detailed results stemmed from performing LOOCV of the proposed methods for the facial expression recognition on the BU-4DFE database. Each evaluation here shares the same setup with the corresponding evaluation conducted on the CK+ database and presented in Ch. 7.

A.1 Appearance-based Method

In what follows, I present the results stemmed from evaluating the proposed appearance based method, depicted in Figure 7.1, on the BU-4DFE database.

A.1.1 Local Binary Pattern Features

The results here are obtained by exploiting the LBP features in the appearance-based algorithm depicted in Figure 7.1. Tables A.1, A.2 report the confusion matrices obtained by the evaluations conducted on BU-4DFE database for the two cases: 6-class and 7 class, respectively. Each column represents samples of the predicted class while each row represents samples of the ground truth class. For a comparison purpose, the results were obtained using four different machine learning algorithms: SVM, NNe, RF, and kNN. Clearly shown in Table A.1, the achieved average recognition rates were $67.85\%, 62.40\%, 53.03\%$, and $43,03\%$ using SVM, NNe, RF, and kNN, respectively. These rates degraded to $62.12\%, 58.08\%, 51.70\%$, and

48.03%, respectively, when I added the neutral state as a 7^{th} category to the classification method (see Table A.2). Here as well, adding neutral category causes a lot of confusions especially with anger and sadness as they are of subtle facial deformations. Less affected were happiness and surprise due to their distinctive and obvious facial deformations. SVM has performed accurately for the facial expression recognition in comparison to NNe, RF, and kNN.

A.1.2 Gabor Filter-based Features

Here I employ the GAB features in the appearance-based algorithm depicted in Figure 7.1. The evaluations of the resulting approach on the BU-4DFE database are summarized via confusion matrices depicted in Tables A.3 and A.4 for 6-class and 7-class cases, respectively. In a good agreement with the evaluations on CK+ database, the recognition rates for happiness and surprise expressions are the highest and not affected by adding the neutral category to the classifier. The recognition rates of the other expressions are below 66%, reflecting the existence of many confusions. With the neutral state, the average recognition rate degraded from 69.04% in Table A.3 to 62.17% in Table A.4.

A.1.3 Histogram of Oriented Gradient Features

Here I employ the HoG features in the appearance-based algorithm depicted in Figure 7.1. The conducted LOOCV are summarized via confusion matrices in Tables A.5 and A.6 for the 6-class and 7-class cases, respectively. In comparison to the evaluation on Ck+ database, the average recognition rate dropped to 77.14% for the 6-class and to 67.71% for the 7-class, as BU-4DFE database is more challenging due to the higher variation of same class samples in terms of intensity. In the 6-class case, happiness and surprise are recognized with high recognition rates of 92.68% and 88.37%, respectively. Relatively low, disgust and fear expressions are recognized with $69.64.68\%$ and 62.71%, respectively. By adding the neutral category to the classifier, happiness and surprise are still recognized with high recognition rates 100% and 72%, respectively, as seen in Table A.6. These two expressions induce distinctive and obvious facial deformations. The recognition rates of the other expressions dropped significantly due to confusions with the neutral state, meanwhile the neutral state is recognized with 88%.

Table A.1: Confusion matrix of 6-class facial expression recognition using LBP features based on evaluation conducted on BU-4DFE database. For each expression, four rows are presented each corresponds to specific machine learning algorithm. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

		Ha	Su	An	Di	Fe	Sa
Ha	SVM	**0.8537**	0	0.0244	0.0122	0.0732	0.0366
	NNe	**0.8659**	0	0.0244	0.0122	0.0610	0.0366
	RF	**0.8902**	0.0244	0.0122	0.0366	0.0244	0.0122
	KNN	**0.7317**	0.1463	0.0244	0.0122	0.0366	0.0488
Su	SVM	0.0115	**0.7471**	0.0345	0	0.1379	0.0690
	NNe	0.0115	**0.7356**	0.0230	0.0230	0.1494	0.0575
	RF	0.0460	0.5632	0.1724	0.0230	0.0345	0.1609
	KNN	0.0115	0.4828	0.0575	0.0575	0.1494	0.2414
An	SVM	0.0423	0.0704	0.5775	0.1268	0.0704	0.1127
	NNe	0.0455	0	0.5758	0.1667	0.0909	0.1212
	RF	0.0845	0.2113	0.4366	0.0986	0.1127	0.0563
	KNN	0.0563	0.2394	0.3521	0.1549	0	0.1972
Di	SVM	0.0727	0.0545	0.1636	**0.6545**	0.0545	0
	NNe	0.0714	0.0536	0.2143	0.5536	0.0536	0.0536
	RF	0.1429	0.1429	0.1607	0.3750	0.1071	0.0714
	KNN	0.0893	0.1429	0.0714	0.4107	0.1071	0.1786
Fe	SVM	0.0980	0.1373	0.0392	0.0588	**0.6667**	0
	NNe	0.1017	0.1525	0.0847	0.0339	0.4576	0.1695
	RF	0.1356	0.0847	0.1695	0.0678	0.4407	0.1017
	KNN	0.0678	0.2034	0.0678	0.0508	0.4237	0.1864
Sa	SVM	0.0317	0.1429	0.1111	0.0635	0.0794	0.5714
	NNe	0.0317	0.0952	0.1270	0.0635	0.1270	0.5556
	RF	0.0635	0.1270	0.1905	0.0317	0.1111	0.4762
	KNN	0.0328	0.1639	0.0984	0.0492	0.1148	0.5410

Table A.2: Confusion matrix of 7-class facial expression recognition using LBP features based on evaluation conducted on BU-4DFE database. For each expression, four rows are presented each corresponds to specific machine learning algorithm. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

		Ha	Su	An	Di	Fe	Sa	Ne
Ha	SVM	**0.8049**	0	0.0122	0.0244	0.0732	0	0.0854
	NNe	**0.8049**	0	0.0244	0.0122	0.0732	0.0366	0.0488
	RF	**0.9146**	0	0.0244	0.0122	0.0244	0.0122	0.0122
	KNN	0.5610	0.1341	0.0976	0.0732	0.0854	0.0122	0.0366
Su	SVM	0.0116	**0.8140**	0.0349	0	0.0581	0.0233	0.0581
	NNe	0.0115	**0.7356**	0	0.0230	0.1149	0.0230	0.0920
	RF	0.0230	0.5747	0.1149	0.0345	0.0575	0.0805	0.1149
	KNN	0.0575	0.5517	0.1034	0.0805	0.0920	0.0690	0.0460
An	SVM	0.0141	0.0282	**0.6056**	0.1268	0.0563	0.0986	0.0704
	NNe	0.0141	0.0282	0.5775	0.1549	0.0563	0.0704	0.0986
	RF	0.0563	0.1690	0.4085	0.0986	0.0986	0.0845	0.0845
	KNN	0.0423	0.1268	0.3944	0.1690	0.0563	0.1831	0.0282
Di	SVM	0.0893	0.0357	0.1786	0.5893	0.0179	0.0536	0.0357
	NNe	0.0909	0.0182	0.2182	0.5455	0	0.0545	0.0727
	RF	0.1250	0.0536	0.1071	0.4107	0.1250	0.0714	0.1071
	KNN	0.1250	0.0893	0.1429	0.3571	0.1071	0.1429	0.0357
Fe	SVM	0.1017	0.1356	0.0508	0.0339	0.4915	0.1017	0.0847
	NNe	0.1000	0.1333	0.0333	0.0500	0.4500	0.1167	0.1167
	RF	0.1356	0.1186	0.1186	0.0508	0.3898	0.0847	0.1017
	KNN	0.1525	0.1017	0.1356	0.1356	0.3390	0.1186	0.0169
Sa	SVM	0	0.0656	0.0984	0	0.1148	0.4918	0.2295
	NNe	0	0.1111	0.1429	0.0317	0.1111	0.3810	0.2222
	RF	0.0794	0.0794	0.1270	0.0159	0.0794	0.4444	0.1746
	KNN	0.0476	0.1429	0.1111	0.0476	0.0159	0.6032	0.0317
Ne	SVM	0.0159	0.0794	0.0794	0.0317	0.0794	0.1587	0.5556
	NNe	0.0159	0.0476	0.0476	0.0794	0.0952	0.1429	0.5714
	RF	0.0159	0.1270	0.1111	0.0794	0.0476	0.1429	0.4762
	KNN	0.0635	0.0635	0.0794	0.0635	0.0476	0.1270	0.5556

Table A.3: Confusion matrix of 6-class facial expression recognition using GAB features based on LOOCV evaluation conducted on BU-4DFE database via SVM. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.8780**	0.0122	0.0122	0.0366	0.0610	0
Su	0.0230	**0.8621**	0.0230	0.0115	0.0460	0.0345
An	0.0141	0.0423	0.5915	0.1690	0.0282	0.1549
Di	0.0536	0.0357	0.0536	**0.6607**	0.1250	0.0714
Fe	0.0781	0.1250	0.0156	0.0938	0.5625	0.1250
Sa	0	0.0476	0.1429	0.0635	0.1587	0.5873

Table A.4: Confusion matrix of 7-class facial expression recognition using GAB features based on LOOCV evaluation conducted on BU-4DFE database via SVM. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.8415**	0.0122	0	0.0366	0.0732	0	0.0366
Su	0.0230	**0.8046**	0.0115	0.0230	0.0690	0.0230	0.0460
An	0.0141	0.0141	0.5775	0.1549	0.0141	0.1268	0.0986
Di	0.0545	0.0364	0.1455	0.5818	0.0727	0.0545	0.0545
Fe	0.0833	0.0667	0	0.1000	0.4833	0.0833	0.1833
Sa	0	0.0159	0.1270	0.0317	0.0794	0.4921	0.2540
Ne	0.0159	0.0635	0.0794	0	0.0952	0.1746	0.5714

Table A.5: Confusion matrix of 6-class facial expression recognition using HOG features based on LOOCV evaluation conducted on BU-4DFE database via SVM. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.9268**	0	0	0.0122	0.0610	0
Su	0	**0.8837**	0.0116	0.0116	0.0930	0
An	0.0141	0	**0.7324**	0.0986	0	0.1549
Di	0.0179	0.0179	0.1250	**0.6964**	0.1071	0.0357
Fe	0.1017	0.0847	0.0339	0.0847	**0.6271**	0.0678
Sa	0	0	0.1429	0.0159	0.0794	**0.7619**

Table A.6: Confusion matrix of 7-class facial expression recognition using HOG features based on LOOCV evaluation conducted on BU-4DFE database via SVM. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**1.00**	0	0	0	0	0	0
Su	0	**0.72**	0	0	0.17	0	0.11
An	0	0	0.54	0.17	0	0.17	0.12
Di	0.05	0	0.17	0.53	0.10	0.10	0.05
Fe	0.10	0	0	0	0.50	0.15	0.25
Sa	0	0	0.10	0	0.05	0.57	0.28
Ne	0	0	0.04	0	0	0.08	**0.88**

Table A.7: Confusion matrix of 6-class facial expression recognition using geometric features extracted from 49 facial points, based on evaluation conducted on BU-4DFE database via SVM. For each expression, two rows are presented. The first row is dedicated for the person-specific scenario, the features are calculated with respect to a prior known person-specific neutral model. The second row is the case where a general neutral model is used. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.9756**	0	0	0	0.0244	0
	0.9634	0	0.0122	0	0.0244	0
Su	0	**0.9302**	0	0.0116	0.0581	0
	0	**0.9302**	0	0.0116	0.0581	0
An	0	0	**0.8310**	0.0423	0	0.1268
	0	0	**0.7746**	0.0563	0.0423	0.1268
Di	0.0179	0	0.1071	**0.7321**	0.0714	0.0714
	0.0179	0	0.0893	**0.7679**	0.0893	0.0357
Fe	0.1017	0.0847	0.0169	0.0508	**0.7119**	0.0339
	0.1017	0.0847	0.0339	0.1017	0.4746	0.2034
Sa	0	0	0.1111	0.0317	0.0317	**0.8254**
	0	0.0159	0.1905	0	0.1270	**0.6667**

A.2 Geometry-based Method

In what follows, I present the results stemmed from evaluating the proposed Geometric-based method on the BU-4DFE database.

A.2.1 A Method of 49 Facial Points

Table A.7 summarizes a similar LOOCV to that in Sec. 7.2.1 but on the BU-4DFE database. Interestingly, I achieved an average recognition rate of 83.44% in the case of person-specific for the 6-class case, which is just less by 6.3% in comparison to the average rate based on LOOCV on CK+. This recognition rate dropped by 7.15% to 76.29% when I employed the general neutral model instead of the person-specific one. This drop is higher than the corresponding drop on CK+ database, which highlights the more variability in the face structure involved in the BU-4DFE database. As expected, the recognition rates of happiness and surprise are

Table A.8: Confusion matrix of 7-class facial expression recognition using geometric features extracted from 49 facial points, based on evaluation conducted on BU-4DFE database via SVM. Here, we infer the neutral state as person-specific neutral state is not available Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.9634**	0	0	0	0.0244	0	0.0122
Su	0	**0.9186**	0	0.0233	0.0581	0	0
An	0	0	**0.6479**	0.0563	0	0.1408	0.1549
Di	0.0179	0	0.0893	**0.7679**	0.0714	0.0179	0.0357
Fe	0.1017	0.0847	0.0339	0.1017	0.4237	0.1356	0.1186
Sa	0	0	0.1429	0	0.1111	0.4921	0.2540
Ne	0	0	0.1341	0	0.0122	0.0976	**0.7561**

not affected, their deformations still obvious. The drop in recognition rate is seen through the confusion between sadness and fear expressions, where 20.34% of the fear samples are recognized as sadness and 20.34% of the sadness samples are recognized as fear. Clearly shown from above that minimizing the individual variation of the face shape through a person-specific normalization of the geometrical features enhances the expression inference.

Table A.8 summarizes the results of the conducted cross-validation in the case of 7-class without prior person-specific information. The recognition rates dropped here as well. The achieved average recognition rate is 71%, where happiness and surprise are perceived with the highest rates 96.34% and 91.86% respectively. Neutral and Disgust are recognized with 75.61% and 76.79% respectively. A lot of confusions were arisen with the neutral, e.g. 15% of anger samples, 11.86% of fear samples, and 25.4% of sadness samples are recognized as neutral.

A.2.2 A Method of 8 Facial Points

Here, the geometry-based approach exploited only eight facial points. In what follows, I present the confusion matrices stemmed from evaluating the proposed approach under two scenarios.

A.2.2.1 Person-specific Neutral State

Table A.9 shows our LOOCV results compared to results obtained by applying the textured-based approach CERT [98] on the BU-4DFE database. For each facial image, CERT provides probability values for eight human expressions (happy, surprise, anger, disgust, fear, sadness, contempt, and neutral). The neutral and contempt expressions are not included in this evaluation. Hence for a meaningful comparison, I decided on the expression from the six expressions with a higher probability to be the recognized one, even if the probability value of contempt or neutral is higher. The proposed approach achieved a high recognition rate for happiness, surprise, anger, and disgust expressions. In contrast with our evaluation on CK+ database (Table 7.9), it achieved lower recognition rate for fear expression. The CERT obtained high recognition rates for happy, sadness, and disgust expression recognition. On the other hand, CERT achieved lower recognition rates for surprise and anger expressions. The higher rate of sadness by CERT illustrates the importance of exploiting texture-based features; however, small confusions between anger and sadness as well as between fear and sadness are also unavoidable, even using texture-based features. The use of geometry-based approaches results in a confusion between happiness and fear. This confusion is less with the help of texture-based features as the results of CERT show. In summary, the proposed approach achieved an average recognition rate of 83.88% compared to 71.68% achieved by CERT, which indicates that geometric features extracted from 8 fiducial points are superior to hundreds features extracted from Gabor filters distributed among the face image.

A.2.2.2 General Neutral State

The results of a similar LOOCV to that in 7.2.2.3 but on the BU-4DFE database are summarized via confusion matrices in Table A.10. Due to their distinctive facial deformations which are easier to be detected, happiness and surprise expressions are recognized with high rates: 88.4%, 93.7% and 85.36%, 86.2% using SVM and kNN classifiers, respectively. Similarly, confusions of subtle expressions with neutral are present. In contrast with the evaluation on the CK+ database, I obtained a lower recognition rate for neutral expression and a higher one for sadness. I achieved average recognition rates of 68.04% and 57.92% using SVM and kNN classifiers,

Table A.9: Confusion matrix of facial expression recognition based on LOOCV evaluation conducted on BU-4DFE database using eight points, SVM, and person-specific neutral model. The first row in each expression represents our results. The other row shows the results obtained by CERT [98].

	Ha	Su	An	Di	Fe	Sa
Ha	**0.926**	0.00	0.006	0.0133	0.0365	0.0182
	0.927	0.0244	0.00	0.0182	0.0182	0.0122
Su	0.0114	**0.959**	0.00	0.00	0.0296	0.00
	0.0115	0.592	0.0283	0.0402	0.207	0.121
An	0.00	0.00	**0.852**	0.0565	0.00	0.0915
	0.00	0.0144	0.57	0.317	0.00	0.0986
Di	0.0357	0.0178	0.0719	**0.803**	0.0449	0.0267
	0.0179	0.00	0.125	**0.732**	0.0269	0.0982
Fe	0.1694	0.0593	0.0087	0.0254	**0.644**	0.0932
	0.0597	0.00	0.0593	0.11	0.576	0.19.5
Sa	0.00	0.00	0.119	0.00	0.032	**0.849**
	0.00	0.00	0.0794	0.0166	0.00	**0.904**

respectively. These rates are lower than that on CK+, which is reasonable due to the higher error in the facial point localization which comes from the point localization method employed on this database. Additionally, this database presents each expression with varying intensity. Once again, SVM classifier outperforms kNN for facial expression recognition.

A.2.2.3 Approach Evaluation with the Developed Facial Point Detector

Similar evaluations to those in Sec. 7.2.2.4 but on the BU-4DFE database are reported via confusion matrices in Tables A.11 and A.12. By comparing those results to those in Tables A.9 and A.10, I achieved comparable average recognition rates. In particular, I achieved average recognition rates of 82.06% for the 6-class person-specific case (compared to 83.88%), of 72.96% for the 6-class general neutral case, and of 68.87% for the 7-class case (compared to 68.04%).

Table A.10: Confusion matrix of 7-class facial expression recognition using geometric features extracted from 8 facial points in the general neutral state case, based on cross-validation evaluation on BU-4DFE database. The first row in each expression represents results using SVM classifier. The other row shows the results using kNN classifier.

	An	Di	Fe	Ha	Sa	Su	Ne
Ha	**0.884**	0.0183	0.0243	0.0365	0.0365	0.00	0.00
	0.8536	0.0243	0.00	0.07317	0.01219	0.03658	0.00
Su	0.0057	**0.937**	0.00	0.0345	0.0114	0.0114	0.00
	0.00	**0.8620**	0.00	0.05747	0.03448	0.00	0.04597
An	0.0140	0.00	**0.627**	0.106	0.0211	0.119	0.112
	0.02816	0.00	0.4225	0.07042	0.02816	0.1690	0.2816
Di	0.0625	0.0267	0.1607	0.598	0.0625	0.0267	0.0625
	0.1250	0.01785	0.1964	0.4642	0.01785	0.03571	0.1428
Fe	0.118	0.0085	0.0762	0.0338	0.559	0.118	0.0847
	0.1525	0.01694	0.03389	0.08474	0.2881	0.1694	0.2542
Sa	0.00	0.00	0.174	0.0158	0.111	0.534	0.1634
	0.00	0.00	0.1904	0.01587	0.1269	0.4444	0.2222
Ne	0.0202	0.0202	0.202	0.0202	0.0 716	0.0418	**0.624**
	0.02	0.02	0.06	0.00	0.12	0.06	**0.72**

Table A.11: Confusion matrix of 6-class facial expression recognition using using the proposed geometric-based approach of eight facial points, those points were detected using a point detector, developed here in Ch. 5. The cross-validation was conducted on BU-4DFE database. The first row in each expression summarizes the results in the person-specific case, while the other row in the person-independent case.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.8902**	0	0	0	0.1098	0
	0.9024	0	0.0244	0	0.0732	0
Su	0	**0.9535**	0.0116	0.0116	0.0116	0.0116
	0	**0.8953**	0	0.0233	0.0814	0
An	0	0	**0.8451**	0.0282	0.0141	0.1127
	0	0	**0.7746**	0.0704	0.0423	0.1127
Di	0.0179	0.0179	0.0893	**0.6964**	0.0893	0.0893
	0	0.0357	0.1250	**0.6786**	0.1250	0.0357
Fe	0.0847	0.0678	0.0169	0.0169	**0.7288**	0.0847
	0.0678	0.0847	0.0678	0.1186	0.4915	0.1695
Sa	0	0	0.0952	0.0159	0.0794	**0.8095**
	0	0.0159	0.1746	0	0.1746	**0.6349**

Table A.12: Confusion matrix of 7-class facial expression recognition using the proposed geometric-based approach of eight facial points in person independent mode, those points were detected using a point detector, developed here in Ch. 5. The cross-validation was conducted on BU-4DFE database..

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.9268**	0	0	0	0.0488	0	0.0244
Su	0	**0.8953**	0	0.0233	0.0581	0	0.0233
An	0	0	**0.6761**	0.0704	0.0141	0.0845	0.1549
Di	0	0.0536	0.1250	**0.6607**	0.1071	0.0357	0.0179
Fe	0.0678	0.0847	0.0508	0.0678	0.4237	0.0847	0.2203
Sa	0	0.0159	0.1429	0	0.0952	0.5556	0.1905
Ne	0	0	0.0976	0.0244	0.0976	0.0976	**0.6829**

APPENDIX B

Cross-database validation of the proposed method for facial expression recognition

In this chapter, I present additional results stemmed from cross-database evaluations that were conducted to asses the generalization capability of the proposed models for the facial expression recognition.

B.1 Appearance-based Method

In what follows, I present the results stemmed from conducting cross-database evaluations of the proposed appearance-based method that is depicted in Figure 7.1.

B.1.1 Local Binary Pattern Features

In Sec. 7.1.1, the employment of SVM classifiers yields the best average recognition rates. Based on that, I only employed SVM classifiers here. Tables B.1, B.2 show the confusion matrices resulting from using CK+ database as a training set and BU-4DFE database as a testing set for 6-class and 7-class cases, respectively. On the other hand, the confusion matrices in Tables B.3, B.4 summarize the evaluation in which BU-4DFE database was employed as a training set and CK+ database as a tesing set for 6-class and 7-class cases, respectively.

The recognition rates were lower than those resulting from LOOCV conducted on each database separately as expected due to many factors. The two databases

Table B.1: Confusion matrix of 6-class facial expression recognition using LBP features based on cross-database evaluation, the model was trained using ck+ and evaluated on BU-4DFE database. SVM was employed as a machine learning method. Each column represents samples of the predicted class, while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.6220**	0.1220	0.1098	0.0366	0.0732	0.0366
Su	0.0230	**0.6782**	0.1839	0.0345	0.0115	0.0690
An	0.0141	0.1972	0.4507	0.2535	0.0141	0.0704
Di	0.1607	0.1429	0.2321	0.3929	0.0357	0.0357
Fe	0.0339	0.3220	0.1017	0.0678	0.4407	0.0339
Sa	0.0317	0.1587	0.0794	0.0794	0.0794	0.5714

vary in many aspects from image quality to expression depiction Moreover, while the apex frames in CK+ database depict the expression in full intensity, the apex frames in BU-4DFE database are of varying intensity (non-annotated intensity).

In the case of using CK+ database as a training set, the average recognition rate drop from 70.91% (testing using LOOCV on CK+) to 52.60% for the 6-class case, and from 64.89% to 46.07% for the 7-class case. Clearly shown that considerable confusions among surprise, fear, and disgust exist due to the poor performance of LBP besides the difference in portraying the expressions in both sets.
In the case of using BU-4DFE database as a training set (Tables B.3, B.4), the average recognition rates drop as well from 67.85% (testing using LOOCV on BU-4DFE) to 49.95%, for the 6-class case, and from 62.12% to 47.70% for the 7-class case. Happiness and surprise expressions have the best recognition rates which can be attributed to their distinctive facial deformations.

B.1.2 Gabor filter-based Features

Here, cross-database evaluations were carried out using the GAB features; SVM classifier was employed here as well due to its better performance as shown in Sec. 7.1.1. Tables B.5 and B.6 summarize the confusion matrices stemmed from exploiting the CK+ database as a training set and BU-4DFE database as a testing set for 6-class and 7-class cases, respectively. The average recognition rates are 43.83% and 42.62% for both cases, which are much lower in comparison with the

Table B.2: Confusion matrix of 7-class facial expression recognition using LBP features based on cross-database evaluation, the model was trained using ck+ and evaluated on BU-4DFE database. SVM was employed as a machine learning method. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.6585**	0.0976	0.0610	0.0244	0.0732	0	0.0854
Su	0.0116	**0.7093**	0.0465	0	0.0116	0.0465	0.1744
An	0	0.1268	0.3239	0.1831	0.0141	0.0704	0.2817
Di	0.1250	0.1071	0.2500	0.3214	0	0.0179	0.1786
Fe	0.0323	0.1290	0.2581	0	0.3387	0.0323	0.2097
Sa	0	0.1429	0.0476	0.0159	0.0317	0.2698	0.4921
Ne	0	0.1270	0.1587	0.0952	0.0159	0	**0.6032**

Table B.3: Confusion matrix of 6-class facial expression recognition using LBP features based on cross-database evaluation, the model was trained using BU-4DFE and evaluated on ck+ database. SVM was employed as a machine learning method. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.6714**	0.0857	0.0429	0.0714	0.0714	0.0571
Su	0.0122	**0.7195**	0.0244	0.0244	0.1463	0.0732
An	0.0444	0.2222	0.2444	0.1556	0.1111	0.2222
Di	0.0172	0.1724	0.2586	0.3103	0.0345	0.2069
Fe	0.1600	0.1200	0.0400	0.0400	0.4800	0.1600
Sa	0	0.2143	0.0357	0.1071	0.0714	0.5714

Table B.4: Confusion matrix of 7-class facial expression recognition using LBP features based on cross-database evaluation, the model was trained using BU-4DFE and evaluated on ck+ database. SVM was employed as a machine learning method. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.7000**	0.0714	0.0571	0.0429	0.0714	0.0429	0.0143
Su	0.0122	**0.6829**	0.0366	0.0244	0.1341	0.0366	0.0732
An	0.0444	0.1778	0.3333	0.1333	0.0889	0.1778	0.0444
Di	0.0169	0.1864	0.2203	0.3559	0.0339	0.1186	0.0678
Fe	0.2400	0.2000	0.0400	0.0800	0.3200	0.1200	0
Sa	0	0.2069	0.0690	0.1724	0.1034	0.3793	0.0690
Ne	0.0050	0.0854	0.1055	0.0553	0.0804	0.1005	0.5678

test on the same database. The decline in the recognition rates can be attributed to the deferences between the two database in terms of image quality and expression depiction. It is clearly to notice that the happinesses and surprise expressions are well recognized across the databases. Similar results were obtained, when the BU-4DFE database was used for training and CK+ database for testing. These results are summarized using confusion matrices in Tables B.7 and B.8 for 6-class and 7-class cases, respectively. Happiness and surprise expressions are identified with higher recognition rates here as well. In the 6-class evaluation, sadness is confused with disgust, disgust with anger. Meanwhile, many other confusions were experienced in the 7-class case.

B.1.3 Histogram of Oriented Gradient Features

In this section, cross-database evaluations are carried out using the HoG features; SVM classifier is employed due to its better performance as shown in Sec. 7.1.1. Tables B.9 and B.10 summarize the confusion matrices stemmed from exploiting the CK+ database as a training set and BU-4DFE database as a testing set for 6-class and 7-class cases, respectively. I achieved average recognition rates of 53.76% and 47.24% for 6-class and 7-class cases, respectively. These rates are lower than the average recognition rates of Tables 7.5 and 7.6, highlighting the differences between the two databases. Clearly seen in Table B.10 that only happiness and surprise expressions are recognized with high rates of 73% and 96.51%, respectively. Fear and

Table B.5: Confusion matrix of 6-class facial expression recognition using GAB features based on cross-database evaluation, the model was trained using ck+ and evaluated on BU-4DFE database. SVM was employed as a machine learning method. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	0.5610	0.1220	0.0976	0.0610	0.1585	0
Su	0.0115	**0.6207**	0.0575	0.0345	0.2759	0
An	0.0563	0.2254	0.3803	0.1549	0.1549	0.0282
Di	0.1071	0.2679	0.1607	0.2857	0.1429	0.0357
Fe	0.0847	0.4068	0.0678	0.0847	0.3220	0.0339
Sa	0.0317	0.1587	0.1111	0.0635	0.1746	0.4603

Table B.6: Confusion matrix of 7-class facial expression recognition using GAB features based on cross-database evaluation, the model was trained using ck+ and evaluated on BU-4DFE database. SVM was employed as a machine learning method. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.6627**	0.0964	0	0.0241	0.0843	0.0120	0.1205
Su	0.0460	0.5402	0	0.0115	0.1149	0.0460	0.2414
An	0.0492	0.1148	0.1967	0.0820	0.0656	0.0820	0.4098
Di	0.0714	0.0893	0.0179	0.3929	0.1607	0.1071	0.1607
Fe	0.1017	0.1864	0.0339	0.0339	0.2542	0.1356	0.2542
Sa	0.1270	0.0635	0	0.0317	0.1587	0.2698	0.3492
Ne	0	0.1270	0	0.0317	0.1746	0	**0.6667**

Table B.7: Confusion matrix of 6-class facial expression recognition using GAB features based on cross-database evaluation, the model was trained using BU-4DFE and evaluated on ck+ database. SVM was employed as a machine learning method. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.6714**	0.1286	0.0714	0.1143	0.0143	0
Su	0	**0.8902**	0.0244	0.0366	0.0366	0.0122
An	0	0.2889	0.3556	0.2667	0	0.0889
Di	0	0.4746	0.0847	0.4237	0	0.0169
Fe	0.1200	0.2400	0.1200	0.3200	0.2000	0
Sa	0	0.3103	0.1379	0.2414	0	0.3103

Table B.8: Confusion matrix of 7-class facial expression recognition using GAB features based on cross-database evaluation, the model was trained using BU-4DFE and evaluated on ck+ database. SVM was employed as a machine learning method. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.6571**	0.1429	0.0714	0.1143	0.0143	0	0
Su	0	**0.8902**	0.0244	0.0366	0.0366	0.0122	0
An	0	0.3111	0.3556	0.2444	0	0.0889	0
Di	0	0.4915	0.0847	0.4068	0	0.0169	0
Fe	0.1200	0.2400	0.0800	0.4000	0.1600	0	0
Sa	0	0.3448	0.1379	0.2414	0	0.2759	0
Ne	0	0.2653	0.1735	0.2755	0.0051	0.0765	0.2041

Table B.9: Confusion matrix of 6-class facial expression recognition using HOG features based on cross-database evaluation; the model was trained using ck+ and evaluated on BU-4DFE database. SVM was employed as a machine learning method. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.6951**	0.0732	0.0610	0.0610	0.0854	0.0244
Su	0	**0.9767**	0.0116	0	0.0116	0
An	0	0.1690	0.4225	0.2394	0	0.1690
Di	0	0.3929	0.0714	0.3214	0.0536	0.1607
Fe	0.0169	0.5932	0.0508	0.0169	0.2542	0.0678
Sa	0	0.2222	0.1587	0.0317	0.0317	0.5556

disgust are confused with surprise in both cases. Incorporating the neutral led to more confusions. Tables B.11 and B.12 summarize the confusion matrices resulting from employing the BU-4DFE database as a training set and CK+ database as a testing set for 6-class and 7-class cases, respectively. As noticed earlier, only happiness and surprise expressions are recognized with high rates. The recognition of anger is better in both cases, while the recognition of the other expressions suffers larger confusions.

B.1.4 Discussion

In the cross-database evaluations, I trained the models using one database and test them on the other one. Generally, testing against another datbase that is captured using different sensor in different environment shall drop the recognition rates. Besides that the variations in expression intensity and depiction between the two database dramatically decline the recognition rates. The best recognition rate (54.17%) was obtained using the HoG features when BU-4DFE database was used for training and CK+ for testing in 6-class case. Interestingly, happiness and surprise expressions are recognized across the two databases with high rates for most configurations.

Table B.10: Confusion matrix of 7-class facial expression recognition using HOG features based on cross-database evaluation, the model was trained using ck+ and evaluated on BU-4DFE database. SVM was employed as a machine learning method. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.7317**	0.0366	0.0122	0.0488	0.0610	0	0.1098
Su	0	**0.9651**	0	0	0.0116	0	0.0233
An	0	0.0986	0.1408	0.2958	0	0.1127	0.3521
Di	0	0.3214	0.0357	0.3393	0.0536	0.1071	0.1429
Fe	0	0.3898	0.0169	0	0.2712	0.0339	0.2881
Sa	0	0.3175	0.0476	0.0317	0	0.2222	0.3810
Ne	0	0.2987	0.0130	0.0260	0.0130	0.0130	**0.6364**

Table B.11: Confusion matrix of 6-class facial expression recognition using HOG features based on cross-database evaluation, the model was trained using BU-4DFE and evaluated on ck+ database. SVM was employed as a machine learning method. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.9429**	0	0.0429	0.0143	0	0
Su	0.0244	**0.8293**	0.0244	0.1098	0.0122	0
An	0.0667	0	**0.8444**	0.0667	0	0.0222
Di	0.0508	0	**0.7458**	0.1864	0.0169	0
Fe	0.5600	0	0.0800	0.1200	0.2400	0
Sa	0.0345	0	0.2759	0.4828	0	0.2069

Table B.12: Confusion matrix of 7-class facial expression recognition using HOG features based on cross-database evaluation, the model was trained using BU-4DFE and evaluated on ck+ database. SVM was employed as a machine learning method. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.9571**	0	0.0143	0.0143	0	0.0143	0
Su	0.0122	**0.7927**	0.0244	0.1220	0.0244	0	0.0244
An	0.1333	0	**0.6667**	0.1778	0	0.0222	0
Di	0.0678	0	0.5932	0.3051	0.0169	0	0.0169
Fe	0.5600	0	0.0400	0.0800	0.2800	0	0.0400
Sa	0.0345	0	0.2759	0.4483	0.0345	0.2069	0
Ne	0.1421	0	0.2741	0.2538	0.0660	0.0406	0.2234

B.2 Geometry-based Method

Martinez [107] state that robust computer vision algorithms for face analysis and recognition should be based on configural and shape features, which are defined as the distance between facial components (mouth, eye, eyebrow, nose, and jaw line). In other words, geometry-based methods are effective for facial analysis tasks. In this dissertation, I exploited geometry-based features to provide a frame-based decision about the expression, hence no spatio-temporal features [130] were exploited. Two scenarios were investigated in Sec. 7.2.1, expression recognition with prior information about person-specific neutral state and without. A geometry-based method ignores the information regarding skin appearance; consequently, it is less susceptible to changes in illumination. Moreover, it is easier to minimize the differences between individuals at the availability of person-specific neutral state.

B.2.1 A Method of 49 Facial Points

In Sec. 7.2.1, I proposed a geometry-based approach to recognize the six basic expressions (happiness, surprise, anger, disgust, fear, and sadness), along with the neutral state in some cases. 96 features were extracted, with a goal of maximizing the inter-class variation and minimizing the intra-class variation. Those features represent the displacement of detected 49 facial points to their corresponding location in either the person-specific or the general neutral model. The displacements

Table B.13: Confusion matrix of 6-class facial expression recognition using geometric features extracted from 49 facial points, based on cross-database evaluation via SVM. The model was trained using ck+ and evaluated on BU-4DFE database. For each expression, two rows are presented. The first row is dedicated for the person-specific scenario, the features were calculated with respect to a prior known person-specific neutral model. The second row is the case where a general neutral model is used. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.8537**	0	0	0.0122	0.1341	0
	0.7927	0.0244	0.0122	0.0122	0.1220	0.0366
Su	0	**0.720**9	0	0	0.2674	0.0116
	0	**0.9651**	0	0.0116	0.0116	0.0116
An	0	0	**0.7324**	0.1690	0.0141	0.0845
	0	0	0.4648	0.2254	0.0141	0.2958
Di	0.1071	0	0.1071	0.4643	0.3036	0.0179
	0.0536	0.0536	0.0714	0.5893	0.0714	0.1607
Fe	0.0678	0.0339	0.0339	0	**0.6441**	0.2203
	0.0678	0.1356	0.0169	0.0169	0.3559	0.4068
Sa	0	0	0.0794	0.0159	0.0476	**0.8571**
	0	0.0159	0.0317	0.0317	0.0317	**0.8889**

are calculated relatively to the face shape; the features were standardized before being forwarded to the SVM classifier.

Table B.13 summarizes expression recognition rates obtained by training a model via the CK+ database and evaluating it on the BU-4DFE database. The first row of each expression is dedicated for the person-dependent scenario, and the second for the person-independent scenario. The achieved average recognition rate is 71.21% in the person-specific neutral case, and 67.61% in the general neutral model. Happiness, surprise, and sadness expressions are recognized with high rates in both cases. More confusions among disgust, fear, and sadness arise, which can be attributed to the inconsistent intensity of the depicted expression in the BU-4DFE database, e.g. 20% of the fear samples in the first case and 40.68% in the second case are recognized as sadness. Contrary to expectations, the recognition rate of the surprise expression raises from 72.09% in person-specific case to 96.51% in the general neutral model. In the former case the individual variation is removed and

Table B.14: Confusion matrix of 6-class facial expression recognition using geometric features extracted from 49 facial points, based on cross-database evaluation via SVM. The model was trained using BU-4DFE and evaluated on ck+ database. For each expression, two rows are presented. The first row is dedicated for the person-specific scenario, the features are calculated with respect to a prior known person-specific neutral model. The second row is the case where a general neutral model is used. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa
Ha	**0.9565**	0	0	0.0290	0.0145	0
	1.0000	0	0	0	0	0
Su	0	**0.9878**	0	0	0	0.0122
	0	**0.9878**	0.0122	0	0	0
An	0	0	**0.9111**	0.0222	0	0.0667
	0	0	**0.8889**	0.0444	0.0444	0.0222
Di	0	0	**0.8814**	0.1017	0	0.0169
	0	0	**0.8644**	0.1356	0	0
Fe	0.0400	0.0800	0	0.1200	**0.7200**	0.0400
	0.2000	0	0	0.0400	**0.7600**	0
Sa	0	0	0.1071	0.0357	0.1071	**0.7500**
	0	0	0.1071	0.1071	0.1786	**0.6071**

Table B.15: Confusion matrix of 7-class facial expression recognition using geometric features extracted from 49 facial points, based on cross-database evaluation via SVM, in person-independent mode. The model was trained using ck+ and evaluated on BU-4DFE database. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**0.7561**	0.0122	0.0122	0.0122	0.0854	0	0.1220
Su	0	**0.9535**	0	0	0.0116	0	0.0349
An	0	0	0.4225	0.1549	0.0141	0.1690	0.2394
Di	0.0714	0.0536	0.0714	0.5357	0.0714	0.1607	0.0357
Fe	0.0508	0.1525	0.0169	0	0.2712	0.2034	0.3051
Sa	0	0.0159	0.0317	0	0.0317	**0.6349**	0.2857
Ne	0	0	0.0488	0	0.0122	0.1220	**0.8171**

Table B.16: Confusion matrix of 7-class facial expression recognition using geometric features extracted from 49 facial points, based on cross-database evaluation via SVM, in person-independent mode. The model is trained using BU-4DFE and evaluated on CK+ database. Each column represents samples of the predicted class while each row represents samples of the ground truth class.

	Ha	Su	An	Di	Fe	Sa	Ne
Ha	**1.0000**	0	0	0	0	0	0
Su	0	**0.9756**	0	0	0.0122	0	0.0122
An	0	0	**0.8222**	0.0444	0.0222	0	0.1111
Di	0	0	**0.8475**	0.1356	0	0	0.0169
Fe	0.0800	0	0	0.0400	**0.8800**	0	0
Sa	0	0	0.1071	0.1429	0.1071	0.5357	0.1071
Ne	0.0290	0	0.2754	0.0145	0.1739	0.0290	0.4783

as the expression intensity varies in the BU-4DFE database, 26% of the samples are identified as fear, which is expressed with similar eyes and eyebrows movement but limited mouth opening.

In Table B.14, the results were obtained by the opposite configuration to the aforementioned one, the model was trained using the BU-4DFE database and evaluated on the CK+ database. Interestingly, happiness, surprise, and anger expressions are recognized with high rates in both cases, more than 95% for happiness, 98% for surprise, and 88% for anger. Sadness and fear expressions are identified with rates ranging between 76% and 60% in both cases. Unlike the other expressions, most of the disgust samples (more than 86%) are identified as anger, which can be attributed to the variations in depicting the expression between the two databases. The variation in expression intensity among individuals in BU-4DFE database makes it a better choice for training as most of the CK+ samples fall in the correct corresponding expression space, except the disgust samples. In general, minimizing the individual variation via extracting geometric features with respect to person-specific neutral state leads to a better generalization capability across databases.

Tables B.15 and B.16 summarize the results of 7-class case, in which no prior information about person-specific was exploited; the neutral state was inferred automatically along with the six basic expressions. The results in Table B.15 were obtained by training a model using the CK+ database and evaluating it on the BU-4DFE database, and a vice versa configuration was used to generate the results in Table B.16. As expected, the average recognition rate drops to 62.73% in the first case (CK+ $\rightarrow$ BU-4DFE), and to 68.96% in the second case (BU-4DFE $\rightarrow$ CK+). In CK+ $\rightarrow$ BU-4DFE case, happiness, surprise, and neutral are recognized with high rates and the rest expressions suffer from significant confusions. In BU-4DFE $\rightarrow$ CK+ case, happiness, surprise, fear, and anger are recognized with magnificent rates, more than 82.22%. As in the 6-class case, disgust is confused with anger. Additionally, the recognition rate of sadness dropped by 10% due to the confusion with the neutral state. 27.54% of the CK+ neutral samples are identified as anger, which is reasonable due to the subtle facial deformations of the anger expression.

B.3 Discussion

By comparing the achieved results via the cross-database evaluations of the geometry based method (of 49 facial points) with those of the appearance-based methods, it is obviously that geometry-based methods have better generalization capability across databases. Apart from depicting the expressions differently or with varying intensity, there is a great contrast between the two databases in terms of image resolution. The appearance-based methods are more susceptible to illumination variations, image resolution deficiency, and image noise. The aforementioned factors have less effect on the geometry-based approaches due to the use of global shape constraints. Moreover, removing the individual dependency in the feature extraction leads always to an improvement in the cross-database expression recognition. BU-4DFE database encompasses more variations in the expressions' intensity; consequently, it provides better results when it is used for training rather than for testing.

Several additional points can be drawn from the presented confusion matrices of the cross-database evaluations, only happiness and surprise expressions are generalized across the two databases using the appearance-based methods, they are recognized with 92.29% and 82.93% in the 6-class case and with 95.71% and 79.27%

in the 7-class case, respectively, as shown in Tables B.11 B.12. Using the geometry-based methods, they are generalized even better with 100% and 98.78% in the 6-class case and with 100% and 97.5% in the 7-class case. Moreover, in Table B.16, the results show that anger and fear expressions are recognized across the databases with high rates as well, 82.22% and 88% respectively. The neutral samples of BU-4DFE database are identified with a rate of 81% as shown in Table B.15, but if I train the model using BU-4DFE database only 47% of the neutral samples of CK+ database are correctly recognized, where 27.5% are identified as anger. On the other hand, 11% of the anger samples are identified as neutral (see Table B.16), which is reasonable due to the subtle facial deformations of the anger expression.

The final message of the cross-database evaluation could be summarized as follows. BU-4DFE database is more suitable for training due to its variation in the expression intensity. Geometric-based methods generalize across databases better than the appearance-based ones. Expressions with obvious facial deformations are recognized always with higher rates in comparison to the expressions of subtle deformations.

Bibliography

[1] Facial muscles, https://www.kenhub.com/en/library/anatomy/the-facial-muscles (accessed: 02- 06- 2016).

[2] ibug - resources.

[3] Project: A companion-technology for cognitive technical systems, http://www.sfb-trr-62.de/ (accessed: 31- 05- 2016).

[4] ABE, S. *Support Vector Machines for Pattern Classification (Advances in Pattern Recognition)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2010.

[5] ALBIOL, A., MONZO, D., MARTIN, A., SASTRE, J., AND ALBIOL, A. Face recognition using hogebgm. *Pattern Recognition Letters 29*, 10 (2008), 1537 – 1543.

[6] ALMUALLIM, H., AND DIETTERICH, T. G. Learning boolean concepts in the presence of many irrelevant features. *Artificial Intelligence 69*, 1 (1994), 279 – 305.

[7] AMBADY, N., AND ROSENTHAL, R. Thin Slices of Expressive behavior as Predictors of Interpersonal Consequences : a Meta-Analysis. *Psychological Bulletin 111*, 2 (1992), 256–274.

[8] AMOR, B. B., DRIRA, H., BERRETTI, S., DAOUDI, M., AND SRIVASTAVA, A. 4-d facial expression recognition by learning geometric deformations. *IEEE Transactions on Cybernetics 44*, 12 (Dec 2014), 2443–2457.

[9] BA, S., AND ODOBEZ, J.-M. Recognizing visual focus of attention from head pose in natural meetings. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 39*, 1 (2009), 16–33.

[10] BALTRUSAITIS, T., MCDUFF, D., BANDA, N., MAHMOUD, M., EL KALIOUBY, R., ROBINSON, P., AND PICARD, R. Real-time inference of mental states from facial expressions and upper body gestures. In *Automatic Face Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on* (March 2011), pp. 909–914.

[11] BALTRUSAITIS, T., ROBINSON, P., AND MORENCY, L. 3d constrained local model for rigid and non-rigid facial tracking. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (June 2012), pp. 2610–2617.

[12] BALTRUSAITIS, T., ROBINSON, P., AND MORENCY, L.-P. Constrained local neural fields for robust facial landmark detection in the wild. In *Computer Vision Workshops (ICCVW), 2013 IEEE International Conference on* (Dec 2013), pp. 354–361.

[13] BARBU, A., SHE, Y., DING, L., AND GRAMAJO, G. Feature selection with annealing for computer vision and big data learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence PP*, 99 (2016), 1–1.

[14] BELHUMEUR, P., JACOBS, D., KRIEGMAN, D., AND KUMAR, N. Localizing parts of faces using a consensus of exemplars. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on* (june 2011), pp. 545 –552.

[15] BOVIK, A. C., CLARK, M., AND GEISLER, W. S. Multichannel texture analysis using localized spatial filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence 12*, 1 (Jan 1990), 55–73.

[16] BRADSKI, G. The OpenCV Library. `http://sourceforge.net/projects/opencvlibrary`.

[17] BRADSKI, G., AND KAEHLER, A. *Learning OpenCV: [computer vision with the OpenCV library]*, 1 ed. O'Reilly, Beijing and and Köln[u.a.], 2008.

[18] BREIMAN, L. Bagging predictors. *Machine Learning 24*, 2 (1996), 123–140.

[19] BREIMAN, L. Random forests. *Machine Learning 45*, 1 (2001), 5–32.

[20] BREIMAN, L., FRIEDMAN, J., OLSHEN, R., AND STONE, C. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

[21] BUDDHARAJU, P., PAVLIDIS, I., AND TSIAMYRTZIS, P. Pose-invariant physiological face recognition in the thermal infrared spectrum. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW '06. Conference on* (New York, USA, June 2006), pp. 53–53.

[22] CALDER, A. J., LAWRENCE, A. D., AND YOUNG, A. W. Neuropsychology of fear and loathing. *Nat Rev Neurosci 2*, 5 (May 2001), 352–363.

[23] CAO, N. T., TON-THAT, A. H., AND CHOI, H.-I. An effective facial expression recognition approach for intelligent game systems. *Int. J. Comput. Vision Robot. 6*, 3 (Jan. 2016), 223–234.

[24] CARCAGN, P., DEL COCO, M., LEO, M., AND DISTANTE, C. Facial expression recognition and histograms of oriented gradients: a comprehensive study. *SpringerPlus 4* (Oct. 2015), 645.

[25] CASTILLO, J., RIVERA, A., AND CHAE, O. Facial expression recognition based on local sign directional pattern. In *Image Processing (ICIP), 2012 19th IEEE International Conference on* (2012), pp. 2613–2616.

[26] CAZZATO, D., LEO, M., AND DISTANTE, C. An investigation on the feasibility of uncalibrated and unconstrained gaze tracking for human assistive applications by using head pose estimation. *Sensors (Basel, Switzerland) 14*, 5 (May 2014), 8363–8379.

[27] CHAN, C.-H., KITTLER, J., AND MESSER, K. *Multi-scale Local Binary Pattern Histograms for Face Recognition*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 809–818.

[28] CHANG, C.-C., AND LIN, C.-J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology 2* (2011), 27:1–27:27. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[29] CHEN, T., YUEN, P., RICHARDSON, M., LIU, G., AND SHE, Z. Detection of psychological stress using a hyperspectral imaging technique. *IEEE Transactions on Affective Computing 5*, 4 (Oct 2014), 391–405.

[30] CHEW, S. W., LUCEY, P., LUCEY, S., SARAGIH, J., COHN, J. F., AND SRIDHARAN, S. Person-independent facial expression detection using constrained local models. In *Face and Gesture 2011* (March 2011), pp. 915–920.

[31] CHOY, C. B., STARK, M., CORBETT-DAVIES, S., AND SAVARESE, S. Enriching object detection with 2d-3d registration and continuous viewpoint estimation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2015), pp. 2512–2520.

[32] COHN, J. F. *Artifical Intelligence for Human Computing: ICMI 2006 and IJCAI 2007 International Workshops, Banff, Canada, November 3, 2006, Hyderabad, India, January 6, 2007, Revised Seleced and Invited Papers*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, ch. Foundations of Human Computing: Facial Expression and Emotion, pp. 1–16.

[33] CORCORAN, P., IANCU, C., CALLALY, F., AND CUCOS, A. Biometric access control for digital media streams in home networks. *IEEE Trans. on Consum. Electron. 53*, 3 (Aug. 2007), 917–925.

[34] CORTES, C., AND VAPNIK, V. Support-vector networks. *Machine Learning 20*, 3 (1995), 273–297.

[35] COVER, T., AND HART, P. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on 13*, 1 (January), 21–27.

[36] CRISTINACCE, D., AND COOTES, T. F. Feature detection and tracking with constrained local models. In *Proc. BMVC* (2006), pp. 95.1–95.10. doi:10.5244/C.20.95.

[37] CRUZ, A., BHANU, B., AND THAKOOR, N. Facial emotion recognition in continuous video. In *Pattern Recognition (ICPR), 2012 21st International Conference on* (Tsukuba, Japan, Nov 2012), pp. 1880–1883.

[38] DAHMANE, A., LARABI, S., DJERABA, C., AND BILASCO, I. Learning symmetrical model for head pose estimation. In *Pattern Recognition (ICPR), 2012 21st International Conference on* (Tsukuba, Japan, November 2012), pp. 3614–3617.

[39] DALAL, N. *Finding People in Images and Videos*. PhD thesis, INRIA Rhone-Alpes, 2006.

[40] DALAL, N., AND TRIGGS, B. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* (San Diego, CA, USA, June 2005), vol. 1, pp. 886–893 vol. 1.

[41] DARWIN, C. *The expression of the emotions in man and animals*. London, John Murray, 1872.

[42] DATTA, S., SEN, D., AND BALASUBRAMANIAN, R. *Integrating Geometric and Textural Features for Facial Emotion Classification Using SVM Frameworks*. Springer Singapore, Singapore, 2017, pp. 619–628.

[43] DE BOULOGNE, G.-B.-A. D. *In Mecanisme de la Physionomie Humaine*. Cambridge University Press, 1862.

[44] DEMPSTER, A. P., LAIRD, N. M., AND RUBIN, D. B. Maximum likelihood from incomplete data via the em algorithm. *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B 39*, 1 (1977), 1–38.

[45] DU, S., TAO, Y., AND MARTINEZ, A. M. Compound facial expressions of emotion. *Proceedings of the National Academy of Sciences of the United States of America 111*, 15 (Mar. 2014), E1454–E1462.

[46] DUDANI, S. A. The distance-weighted k-nearest-neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics SMC-6*, 4 (April 1976), 325–327.

[47] EKMAN, P., AND FRIESEN, W. Facial action coding system: A technique for the measurments of facial movements. *Consulting Psychologists Press* (1978).

[48] EKMAN, P., AND FRIESEN, W. V. *Unmasking the face : a guide to recognizing emotions from facial clues*. Englewood Cliffs, N.J. : Prentice-Hall, 1975. "A Spectrum book".

[49] EKMAN, P., AND FRIESEN, W. V. Measuring facial movement. *Journal of Nonverbal Behavior 1*, 1 (Sept. 1976), 56–75.

[50] EKMAN, P., FRIESEN, W. V., AND ELLSWORTH, P. *Emotion in the Human Face*. Oxford University Press, 1972.

[51] EVERINGHAM, M., SIVIC, J., AND ZISSERMAN, A. Taking the bite out of automated naming of characters in tv video. *Image Vision Comput. 27*, 5 (Apr. 2009), 545–559.

[52] FANELLI, G., WEISE, T., GALL, J., AND GOOL, L. V. Real time head pose estimation from consumer depth cameras. In *Proceedings of the 33rd International Conference on Pattern Recognition* (Berlin, Heidelberg, 2011), DAGM'11, Springer-Verlag, pp. 101–110.

[53] FOSSATI, A., GALL, J., GRABNER, H., REN, X., AND KONOLIGE, K. *Consumer Depth Cameras for Computer Vision*. Springer London, London, 2013.

[54] GHIMIRE, D., LEE, J., LI, Z.-N., AND JEONG, S. Recognition of facial expressions based on salient geometric features and support vector machines. *Multimedia Tools and Applications* (2016), 1–26.

[55] GIAKOUMIS, D., TZOVARAS, D., MOUSTAKAS, K., AND HASSAPIS, G. Automatic recognition of boredom in video games using novel biosignal moment-based features. *IEEE Transactions on Affective Computing 2*, 3 (July 2011), 119–133.

[56] GOURIER, N., HALL, D., AND CROWLEY, J. L. Estimating Face Orientation from Robust Detection of Salient Facial Features. In *Proceedings of Pointing 2004, ICPR, International Workshop on Visual Observation of Deictic Gestures* (2004).

[57] GRIFFIN, H. J., AUNG, M. S. H., ROMERA-PAREDES, B., MCLOUGHLIN, C., MCKEOWN, G., CURRAN, W., AND BIANCHI-BERTHOUZE, N. Perception and automatic recognition of laughter from whole-body motion: Continuous and categorical perspectives. *IEEE Transactions on Affective Computing 6*, 2 (April 2015), 165–178.

[58] GROSS, R., MATTHEWS, I., COHN, J., KANADE, T., AND BAKER, S. Multi-pie. *Image Vision Comput. 28*, 5 (May 2010), 807–813.

[59] GRUEBLER, A., AND SUZUKI, K. Design of a wearable device for reading positive expressions from facial emg signals. *IEEE Transactions on Affective Computing 5*, 3 (July 2014), 227–237.

[60] GUO, G., AND WANG, X. A study on human age estimation under facial expression changes. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (Providence, RI, USA, June 2012), pp. 2547–2553.

[61] GURBUZ, S., OZTOP, E., AND INOUE, N. Model free head pose estimation using stereovision. *Pattern Recognition 45*, 1 (Jan. 2012), 33–42.

[62] GUYON, I., AND ELISSEEFF, A. An introduction to variable and feature selection. *J. Mach. Learn. Res. 3* (Mar. 2003), 1157–1182.

[63] HALL, M. A. *Correlation-based feature selection for machine learning*. PhD thesis, Department of Computer Science, Waikato University, New Zealand, 1999.

[64] HAMMAL, Z., COHN, J. F., AND MESSINGER, D. S. Head movement dynamics during play and perturbed mother-infant interaction. *IEEE Transactions on Affective Computing 6*, 4 (Oct 2015), 361–370.

[65] HAN, J., JI, X., HU, X., GUO, L., AND LIU, T. Arousal recognition using audio-visual features and fmri-based brain response. *IEEE Transactions on Affective Computing 6*, 4 (Oct 2015), 337–347.

[66] HARROLD, N., TAN, C. T., AND ROSSER, D. Towards an expression recognition game to assist the emotional development of children with autism spectrum disorders. In *Proceedings of the Workshop at SIGGRAPH Asia* (New York, NY, USA, 2012), WASA '12, ACM, pp. 33–37.

[67] HASTIE, T., TIBSHIRANI, R., AND FRIEDMAN, J. H. *The Elements of Statistical Learning Data Mining, Data Mining, Inference, and Prediction*, 2nd ed. ed. Springer series in statistics. Springer, New York NY, 2009.

[68] HAYKIN, S. *Neural Networks and Learning Machines (3rd Edition)*, 3 ed. Prentice Hall, Nov. 2008.

[69] HEBB, D. O. *The Organization of Behavior: A Neuropsychological Theory*, new ed ed. Wiley, New York, June 1949.

[70] HOQUE, M. E., MCDUFF, D. J., AND PICARD, R. W. Exploring temporal patterns in classifying frustrated and delighted smiles. *IEEE Transactions on Affective Computing 3*, 3 (July 2012), 323–334.

[71] HSU, C.-W., AND LIN, C.-J. A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks 13*, 2 (Mar 2002), 415–425.

[72] HWANG, M.-C., HA, L. T., KIM, N.-H., PARK, C.-S., AND KO, S.-J. Person identification system for future digital tv with intelligence. *IEEE Trans. on Consum. Electron. 53*, 1 (Feb. 2007), 218–226.

[73] INC., L. luxand facesdk ver. 6.1. www.luxand.com/facesdk/, Dec. 2015.

[74] INC., M. Face++ matlab sdk demo. www.faceplusplus.com, Dec 2013.

[75] JAIN, A. K., AND FARROKHNIA, F. Unsupervised texture segmentation using gabor filters. *Pattern Recognition 24*, 12 (1991), 1167 – 1186.

[76] JARLIER, S., GRANDJEAN, D., DELPLANQUE, S., N'DIAYE, K., CAYEUX, I., VELAZCO, M. I., SANDER, D., VUILLEUMIER, P., AND SCHERER, K. R. Thermal analysis of facial muscles contractions. *IEEE Transactions on Affective Computing 2*, 1 (Jan 2011), 2–9.

[77] JENKE, R., PEER, A., AND BUSS, M. Feature extraction and selection for emotion recognition from eeg. *IEEE Transactions on Affective Computing 5*, 3 (July 2014), 327–339.

[78] JI, Q., ZHU, Z., AND LAN, P. Real-time nonintrusive monitoring and prediction of driver fatigue. *IEEE Transactions on Vehicular Technology 53*, 4 (July 2004), 1052–1068.

[79] JIMENEZ, P., NUEVO, J., AND BERGASA, L. Face pose estimation and tracking using automatic 3d model construction. In *Computer Vision and Pattern Recognition Workshops, 2008. CVPRW '08. IEEE Computer Society Conference on* (Anchorage, AK, USA, June 2008), pp. 1–7.

[80] JIMENEZ-PINTO, J., AND TORRES-TORRITI, M. Optical flow and driver's kinematics analysis for state of alert sensing. *Sensors 13*, 4 (2013), 4225.

[81] KALIOUBY, R. E., AND ROBINSON, P. Real-time inference of complex mental states from facial expressions and head gestures. In *Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04)*

Volume 10 - Volume 10 (Washington, DC, USA, 2004), CVPRW '04, IEEE Computer Society, pp. 154–160.

[82] KAR, N. B., BABU, K. S., AND JENA, S. K. *Face Expression Recognition Using Histograms of Oriented Gradients with Reduced Features*. Springer Singapore, Singapore, 2017, pp. 209–219.

[83] KARG, M., SAMADANI, A. A., GORBET, R., KHNLENZ, K., HOEY, J., AND KULIC, D. Body movements for affective expression: A survey of automatic recognition and generation. *IEEE Transactions on Affective Computing 4*, 4 (Oct 2013), 341–359.

[84] KAZEMI, V., AND SULLIVAN, J. One millisecond face alignment with an ensemble of regression trees. In *2014 IEEE Conference on Computer Vision and Pattern Recognition* (June 2014), pp. 1867–1874.

[85] KING, D. E. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research 10* (2009), 1755–1758.

[86] KOELSTRA, S., PANTIC, M., AND PATRAS, I. A dynamic texture-based approach to recognition of facial actions and their temporal models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 32*, 11 (2010), 1940–1954.

[87] KOLLER, D., AND SAHAMI, M. Toward optimal feature selection. In *In 13th International Conference on Machine Learning* (1995), pp. 284–292.

[88] LANKES, M., RIEGLER, S., WEISS, A., MIRLACHER, T., PIRKER, M., AND TSCHELIGI, M. Facial expressions as game input with different emotional feedback conditions. In *Proceedings of the 2008 International Conference on Advances in Computer Entertainment Technology* (New York, NY, USA, 2008), ACE '08, ACM, pp. 253–256.

[89] LE, V., BRANDT, J., LIN, Z., BOURDEV, L., AND HUANG, T. S. Interactive facial feature localization. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part III* (Berlin, Heidelberg, 2012), ECCV'12, Springer-Verlag, pp. 679–692.

[90] LECUN, Y., BOTTOU, L., ORR, G. B., AND MÜLLER, K. R. *Efficient BackProp*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998, pp. 9–50.

[91] LEE, D., CHUNG, J., AND YOO, C. D. *Computer Vision – ACCV 2014: 12th Asian Conference on Computer Vision, Singapore, Singapore, November 1-5, 2014, Revised Selected Papers, Part IV*. Springer International Publishing, Cham, 2015, ch. Joint Estimation of Pose and Face Landmark, pp. 305–319.

[92] LEE, Y.-H., KIM, C. G., KIM, Y., AND WHANGBO, T. K. Facial landmarks detection using improved active shape model on android platform. *Multimedia Tools and Applications 74*, 20 (2015), 8821–8830.

[93] LI, H., DING, H., HUANG, D., WANG, Y., ZHAO, X., MORVAN, J.-M., AND CHEN, L. An efficient multimodal 2d + 3d feature-based approach to automatic facial expression recognition. *Comput. Vis. Image Underst. 140*, C (Nov. 2015), 83–92.

[94] LIENHART, R., KURANOV, A., AND PISAREVSKY, V. *Empirical Analysis of Detection Cascades of Boosted Classifiers for Rapid Object Detection*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003, pp. 297–304.

[95] LIN, H.-T., LIN, C.-J., AND WENG, R. C. A note on platt's probabilistic outputs for support vector machines. *Machine Learning 68*, 3 (2007), 267–276.

[96] LITTLEWORT, G., BARTLETT, M., FASEL, I., SUSSKIND, J., AND MOVELLAN, J. Dynamics of facial expression extracted automatically from video. In *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW '04. Conference on* (2004), pp. 80–80.

[97] LITTLEWORT, G., BARTLETT, M. S., SALAMANCA, L. P., AND REILLY, J. Automated measurement of children's facial expressions during problem solving tasks. In *FG* (2011), IEEE, pp. 30–35.

[98] LITTLEWORT, G., WHITEHILL, J., WU, T., FASEL, I., FRANK, M., MOVELLAN, J., AND BARTLETT, M. The computer expression recognition toolbox (cert). In *Automatic Face Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on* (march 2011), pp. 298 –305.

[99] LONG, N., GIANOLA, D., ROSA, G., AND WEIGEL, K. Dimension reduction and variable selection for genomic selection: application to predicting milk yield in holsteins. *Journal of Animal Breeding and Genetics 128*, 4 (2011), 247–257.

[100] LÖRINCZ, A., JENI, L. A., SZABÓ, Z., COHN, J. F., AND KANADE, T. Emotional expression classification using time-series kernels. *CoRR abs/1306.1913* (2013).

[101] LOWE, D. G. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision 60*, 2 (Nov. 2004), 91–110.

[102] LU, S., TSECHPENAKIS, G., AND METAXAS, D. N. Blob analysis of the head and hands: A method for deception detection. In *and Emotional State Identification, Hawaii International Conference on System Sciences, Big Island* (2005).

[103] LUCAS, B. D., AND KANADE, T. An iterative image registration technique with an application to stereo vision. pp. 674–679.

[104] LUCEY, P., COHN, J., KANADE, T., SARAGIH, J., AMBADAR, Z., AND MATTHEWS, I. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on* (june 2010), pp. 94 –101.

[105] MA, B., ZHANG, W., SHAN, S., CHEN, X., AND GAO, W. Robust head pose estimation using lgbp. In *Proceedings of the 18th International Conference on Pattern Recognition - Volume 02* (Washington, DC, USA, 2006), ICPR '06, IEEE Computer Society, pp. 512–515.

[106] MANJUNATH, B., SHEKHAR, C., AND CHELLAPPA, R. A new approach to image feature detection with applications. *Pattern Recognition 29*, 4 (1996), 627 – 640.

[107] MARTINEZ, A. Deciphering the face. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on* (june 2011), pp. 7 –12.

[108] MARTINEZ, B., VALSTAR, M., BINEFA, X., AND PANTIC, M. Local evidence aggregation for regression-based facial point detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 35*, 5 (May 2013), 1149–1163.

[109] MASI, I., RAWLS, S., MEDIONI, G., AND NATARAJAN, P. Pose-aware face recognition in the wild. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2016), pp. 4838–4846.

[110] MCCULLOCH, W. S., AND PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics 5*, 4 (1943), 115–133.

[111] MCMURROUGH, C., METSIS, V., KOSMOPOULOS, D., MAGLOGIANNIS, I., AND MAKEDON, F. A dataset for point of gaze detection using head poses and eye images. *Journal on Multimodal User Interfaces 7*, 3 (2013), 207–215.

[112] MILBORROW, S., MORKEL, J., AND NICOLLS, F. The MUCT Landmarked Face Database. *Pattern Recognition Association of South Africa* (2010). `http://www.milbo.org/muct`.

[113] MILBORROW, S., AND NICOLLS, F. Locating facial features with an extended active shape model. In *Proceedings of the 10th European Conference on Computer Vision: Part IV* (Berlin, Heidelberg, 2008), ECCV '08, Springer-Verlag, pp. 504–513.

[114] MOORE, S., AND BOWDEN, R. Local binary patterns for multi-view facial expression recognition. *Comput. Vis. Image Underst. 115*, 4 (Apr. 2011), 541–558.

[115] MORENCY, L.-P., AND DARRELL, T. Head gesture recognition in intelligent interfaces: The role of context in improving recognition. In *Proceedings of the 11th International Conference on Intelligent User Interfaces* (New York, NY, USA, 2006), IUI '06, ACM, pp. 32–38.

[116] MU, Y., YAN, S., ZHOU, B., HUANG, T., AND LIU, Y. Discriminative local binary patterns for human detection in personal album. *2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 00* (2008), 1–8.

[117] MUKHERJEE, S. S., AND ROBERTSON, N. M. Deep head pose: Gaze-direction estimation in multimodal video. *IEEE Transactions on Multimedia 17*, 11 (Nov 2015), 2094–2107.

[118] MUNI, D. P., PAL, N. R., AND DAS, J. Genetic programming for simultaneous feature selection and classifier design. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 36*, 1 (Feb 2006), 106–117.

[119] MURPHY-CHUTORIAN, E., DOSHI, A., AND TRIVEDI, M. Head pose estimation for driver assistance systems: A robust algorithm and experimental

evaluation. In *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE* (Sept 2007), pp. 709–714.

[120] NARDELLI, M., VALENZA, G., GRECO, A., LANATA, A., AND SCILINGO, E. P. Recognizing emotions induced by affective sounds through heart rate variability. *IEEE Transactions on Affective Computing 6*, 4 (Oct 2015), 385–394.

[121] NGUYEN, D. T., ZONG, Z., OGUNBONA, P., AND LI, W. Object detection using non-redundant local binary patterns. In *2010 IEEE International Conference on Image Processing* (Sept 2010), pp. 4609–4612.

[122] NIESE, R., AL-HAMADI, A., FARAG, A., NEUMANN, H., AND MICHAELIS, B. Facial expression recognition based on geometric and optical flow features in colour image sequences. *Computer Vision, IET 6*, 2 (march 2012), 79 –89.

[123] NIESE, R., AL-HAMADI, A., HEUER, M., MICHAELIS, B., AND MATUSZEWSKI, B. Machine vision based recognition of emotions using the circumplex model of affect. In *Multimedia Technology (ICMT), 2011 International Conference on* (July 2011), pp. 6424–6427.

[124] NIESE, R., AL-HAMADI, A., MICHAELIS, B., AND NEUMANN, H. Integration of geometric and dynamic features for facial expression recognition in color image sequences. In *Soft Computing and Pattern Recognition (SoCPaR), 2010 International Conference of* (dec. 2010), pp. 237 –240.

[125] NIESE, R., WERNER, P., AND AL-HAMADI, A. Accurate, fast and robust realtime face pose estimation using kinect camera. In *2013 IEEE International Conference on Systems, Man and Cybernetics - SMC* (Manchester, UK, Oct 2013), pp. 487–490.

[126] OJALA, T., PIETIKÄINEN, M., AND HARWOOD, D. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition 29*, 1 (Jan. 1996), 51–59.

[127] PALM, C., KEYSERS, D., LEHMANN, T., AND SPITZER, K. Gabor filtering of complex hue/saturation images for color texture classification. In *In Int. Conf. on Computer Vision* (2000), pp. 45–49.

[128] PANG, Y., YUAN, Y., LI, X., AND PAN, J. Efficient hog human detection. *Signal Processing 91*, 4 (2011), 773 – 781.

[129] PANNING, A., AL-HAMADI, A., NIESE, R., AND MICHAELIS, B. Facial expression recognition based on haar-like feature detection. *Pattern Recognition and Image Analysis 18* (2008), 447–452.

[130] PANTIC, M., AND PATRAS, I. Dynamics of facial expression: recognition of facial actions and their temporal segments from face profile image sequences. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 36*, 2 (2006), 433–449.

[131] PANTIC, M., PENTLAND, A., NIJHOLT, A., AND HUANG, T. Human computing and machine understanding of human behavior: a survey. In *Proceedings of the 8th international conference on Multimodal interfaces* (New York, NY, USA, 2006), ICMI '06, ACM, pp. 239–248.

[132] PORIA, S., CAMBRIA, E., BAJPAI, R., AND HUSSAIN, A. A review of affective computing: From unimodal analysis to multimodal fusion. *Information Fusion 37* (2017), 98 – 125.

[133] ROSENBLATT, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review* (1958), 65–386.

[134] RUDDARRAJU, R., HARO, A., AND ESSA, I. A. Fast multiple camera head pose tracking. In *In Proceedings, Vision Interface* (Halifax, Canada, June 2003).

[135] RUDOVIC, O., MEMBER, S., PANTIC, M., (YIANNIS PATRAS, I., AND MEMBER, S. Coupled gaussian processes for pose-invariant facial expression recognition. *IEEE Trans. Pattern Anal. Mach. Intell* (2013), 1357–1369.

[136] RUMELHART, D. E., HINTON, G. E., AND WILLIAMS, R. J. Neurocomputing: Foundations of research. MIT Press, Cambridge, MA, USA, 1988, ch. Learning Representations by Back-propagating Errors, pp. 696–699.

[137] RUSSELL, J. A. A circumplex model of affect. *Journal of Personality and Social Psychology 39*, 6 (Dec. 1980), 1161–1178.

[138] RUSSELL, J. A., LEWICKA, M., AND NIIT, T. A Cross-Cultural Study of a Circumplex Model of Affect. *Journal of Personality and Social Psychology 57*, 5 (1989), 848–856.

[139] RUSSELL, J. A., AND MEHRABIAN, A. Evidence for a three-factor theory of emotions. *Journal of Research in Personality 11*, 3 (Sept. 1977), 273–294.

[140] SAEED, A., AL-HAMADI, A., AND GHONEIM, A. Head pose estimation on top of haar-like face detection: A study using the kinect sensor. *Sensors 15*, 9 (2015), 20945–20966.

[141] SAEED, A., AL-HAMADI, A., NIESE, R., AND ELZOBI, M. Frame-based facial expression recognition using geometrical features. *Advances in Human-Computer Interaction 2014*, 1 (2014), 1–13.

[142] SAITOH, T., MORISHITA, K., AND KONISHI, R. Analysis of efficient lip reading method for various languages. In *ICPR* (2008).

[143] SANDBACH, G., ZAFEIRIOU, S., AND PANTIC, M. Local normal binary patterns for 3d facial action unit detection. In *ICIP* (2012), pp. 1813–1816.

[144] SEBE, N., LEW, M. S., SUN, Y., COHEN, I., GEVERS, T., AND HUANG, T. S. Authentic facial expression analysis. *Image Vision Comput. 25*, 12 (Dec. 2007), 1856–1863.

[145] SHAN, C., GONG, S., AND MCOWAN, P. W. Facial expression recognition based on local binary patterns: A comprehensive study. *Image Vision Comput. 27*, 6 (May 2009), 803–816.

[146] SHARMA, B., THOTA, R., VYDYANATHAN, N., AND KALE, A. Towards a robust, real-time face processing system using cuda-enabled gpus. In *High Performance Computing (HiPC), 2009 International Conference on* (Kochi, India, Dec 2009), pp. 368–377.

[147] SHAWE-TAYLOR, J., AND CRISTIANINI, N. *Kernel Methods for Pattern Analysis*. Cambridge University Press, New York, NY, USA, 2004.

[148] SIDNER, C. L., LEE, C., MORENCY, L.-P., AND FORLINES, C. The effect of head-nod recognition in human-robot conversation. In *Proceedings of the 1st ACM SIGCHI/SIGART Conference on Human-robot Interaction* (New York, NY, USA, 2006), HRI '06, ACM, pp. 290–296.

[149] SMITH, B., BRANDT, J., LIN, Z., AND ZHANG, L. Nonparametric context modeling of local appearance for pose- and expression-robust facial landmark localization. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on* (June 2014), pp. 1741–1748.

[150] SOLEYMANI, M., ASGHARI-ESFEDEN, S., FU, Y., AND PANTIC, M. Analysis of eeg signals and facial expressions for continuous emotion detection. *IEEE Transactions on Affective Computing 7*, 1 (Jan 2016), 17–28.

[151] SULEIMAN, A., AND SZE, V. Energy-efficient hog-based object detection at 1080hd 60 fps with multi-scale support. In *2014 IEEE Workshop on Signal Processing Systems (SiPS)* (Oct 2014), pp. 1–6.

[152] SUN, Y., WANG, X., AND TANG, X. Deep convolutional network cascade for facial point detection. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition* (Washington, DC, USA, 2013), CVPR '13, IEEE Computer Society, pp. 3476–3483.

[153] TANER ESKIL, M., AND BENLI, K. S. Facial expression recognition based on anatomy. *Comput. Vis. Image Underst. 119* (Feb. 2014), 1–14.

[154] TU, J., HUANG, T., AND TAO, H. Accurate head pose tracking in low resolution video. In *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on* (Southampton, UK, April 2006), pp. 573–578.

[155] TZIMIROPOULOS, G., AND PANTIC, M. Gauss-newton deformable part models for face alignment in-the-wild. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on* (June 2014), pp. 1851–1858.

[156] UR REHMAN BUTT, W., AND LOMBARDI, L. A survey of automatic lip reading approaches. In *Eighth International Conference on Digital Information Management (ICDIM 2013)* (Sept 2013), pp. 299–302.

[157] VALSTAR, M., MARTINEZ, B., BINEFA, X., AND PANTIC, M. Facial point detection using boosted regression and graph models. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (june 2010), pp. 2729 –2736.

[158] VALSTAR, M., PANTIC, M., AND PATRAS, I. Motion history for facial action detection in video. In *Systems, Man and Cybernetics, 2004 IEEE International Conference on* (2004), vol. 1, pp. 635–640 vol.1.

[159] VIOLA, P., AND JONES, M. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on* (Kauai, Hawaii, USA, December 2001), vol. 1, pp. 511–518.

[160] VURAL, E., CETIN, M., ERCIL, A., LITTLEWORT, G., BARTLETT, M., AND MOVELLAN, J. Drowsy driver detection through facial movement analysis. In *Proceedings of the 2007 IEEE international conference on Human-computer interaction* (Berlin, Heidelberg, 2007), HCI'07, Springer-Verlag, pp. 6–18.

[161] WANG, X., HAN, T. X., AND YAN, S. An hog-lbp human detector with partial occlusion handling. In *2009 IEEE 12th International Conference on Computer Vision* (Sept 2009), pp. 32–39.

[162] WANG, Z., WANG, S., AND JI, Q. Capturing complex spatio-temporal relations among facial muscles for facial expression recognition. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on* (2013), pp. 3422–3429.

[163] WECHSLER, H. *Face recognition: From theory to applications*, vol. vol. 163 of *NATO ASI series. Series F, Computer and systems sciences*. Springer, Berlin and New York, 1998.

[164] WEN, W., LIU, G., CHENG, N., WEI, J., SHANGGUAN, P., AND HUANG, W. Emotion recognition based on multi-variant correlation of physiological signals. *IEEE Transactions on Affective Computing 5*, 2 (April 2014), 126–140.

[165] WENDEMUTH, A., AND BIUNDO, S. A companion technology for cognitive technical systems. In *Cognitive Behavioural Systems*, A. Esposito, A. Esposito, A. Vinciarelli, R. Hoffmann, and V. Mller, Eds., vol. 7403 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2012, pp. 89–103.

[166] WERNER, P., AL-HAMADI, A., NIESE, R., WALTER, S., GRUSS, S., AND HARALD, C. Towards pain monitoring: Facial expression, head pose, a new

database, an automatic system and remaining challenges. In *British Machine Vision Conference (BMVC)* (Bristol, UK, 2013).

[167] WU, J., AND TRIVEDI, M. M. A two-stage head pose estimation framework and evaluation. *Pattern Recognition 41*, 3 (Mar. 2008), 1138–1158.

[168] WU, T., BARTLETT, M., AND MOVELLAN, J. R. Facial expression recognition using gabor motion energy filters. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on* (2010), pp. 42–47.

[169] WU, T.-F., LIN, C.-J., AND WENG, R. C. Probability estimates for multi-class classification by pairwise coupling. *J. Mach. Learn. Res. 5* (Dec. 2004), 975–1005.

[170] XIONG, X., AND DE LA TORRE, F. Supervised descent method and its applications to face alignment. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on* (June 2013), pp. 532–539.

[171] YAN, J., LEI, Z., YI, D., AND LI, S. Learn to combine multiple hypotheses for accurate face alignment. In *Computer Vision Workshops (ICCVW), 2013 IEEE International Conference on* (Dec 2013), pp. 392–396.

[172] YANG, J., LIANG, W., AND JIA, Y. Face pose estimation with combined 2d and 3d hog features. In *Pattern Recognition (ICPR), 2012 21st International Conference on* (Tsukuba, Japan, November 2012), pp. 2492–2495.

[173] YANG, Y., FAIRBAIRN, C., AND COHN, J. F. Detecting depression severity from vocal prosody. *IEEE Transactions on Affective Computing 4*, 2 (April 2013), 142–150.

[174] YIN, L., CHEN, X., SUN, Y., WORM, T., AND REALE, M. A high-resolution 3d dynamic facial expression database. In *Automatic Face Gesture Recognition, 2008. FG '08. 8th IEEE International Conference on* (sept. 2008), pp. 1 –6.

[175] YU, X., HUANG, J., ZHANG, S., YAN, W., AND METAXAS, D. N. Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model. In *2013 IEEE International Conference on Computer Vision* (Dec 2013), pp. 1944–1951.

[176] ZAVASCHI, T. H., JR., A. S. B., OLIVEIRA, L. E., AND KOERICH, A. L. Fusion of feature sets and classifiers for facial expression recognition. *Expert Systems with Applications 40*, 2 (2013), 646 – 655.

[177] ZHANG, B., SHAN, S., CHEN, X., AND GAO, W. Histogram of gabor phase patterns (hgpp): A novel object representation approach for face recognition. *IEEE Transactions on Image Processing 16*, 1 (Jan 2007), 57–68.

[178] ZHANG, L., TJONDRONEGORO, D., AND CHANDRAN, V. Representation of facial expression categories in continuous arousal-valence space: Feature and correlation. *Image Vision Comput. 32*, 12 (2014), 1067–1079.

[179] ZHANG, W., SHAN, S., GAO, W., CHEN, X., AND ZHANG, H. Local gabor binary pattern histogram sequence (lgbphs): a novel non-statistical model for face representation and recognition. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on* (2005), vol. 1, pp. 786–791 Vol. 1.

[180] ZHANG, X., AND GAO, Y. Face recognition across pose: A review. *Pattern Recognition 42*, 11 (Nov. 2009), 2876–2896.

[181] ZHANG, Y., AND JI, Q. 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.

[182] ZHAO, G., AND PIETIKAINEN, M. Dynamic texture recognition using local binary patterns with an application to facial expressions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 29*, 6 (2007), 915–928.

[183] ZHAO, G., AND PIETIKÄINEN, M. Boosted multi-resolution spatiotemporal descriptors for facial expression recognition. *Pattern Recognition Letters 30*, 12 (2009), 1117–1127.

[184] ZHONG, L., LIU, Q., YANG, P., LIU, B., HUANG, J., AND METAXAS, D. N. Learning active facial patches for expression analysis. In *2012 IEEE Conference on Computer Vision and Pattern Recognition* (June 2012), pp. 2562–2569.

[185] ZHU, X., AND RAMANAN, D. Face detection, pose estimation, and landmark localization in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* (2012), pp. 2879–2886.

[186] ZHU, Y., DE SILVA, C., AND KO, C. Using moment invariants and hmm in facial expression recognition. In *Image Analysis and Interpretation, 2000. Proceedings. 4th IEEE Southwest Symposium*, pp. 305–309.

Concise Curriculum Vitae

Name:	Anwar Maresh Qahtan Saeed
Citizenship:	Yemeni
Born in:	Taiz, Yemen
Mail:	P.O. Box 4120, 39016 Magdeburg, Germany
E-mail:	anwar.saeed@ovgu.de

Education

2010 – Present	Pursuing a PhD degree, IIKT, Otto-von-Guericke University, Magdeburg, Germany.
2007 – 2010	Master of Science, Electrical Communications Engineering (ECE), University of Kassel, Germany. (**Graduation grade: 1.2)**
1998 – 2003	Bachelor of Science, Electronics and Electrical Communications, Faculty of Engineering, Cairo University in Egypt. (**Very good with honour degree (82.64 %)**)

Professional Experience

2005 – 2007	Research associate, Taiz University, Taiz, Yemen.
2003 – 2005	Field Engineer, MAM International Corporation, Sana'a, Yemen.

Magdeburg,

Anwar Maresh Qahtan Saeed

Related Publications

Most of the material contained in this dissertation is partly based on the following refereed papers and journals published in a variety of peer-reviewed journals and international conferences.

Journal Publications:

1) A. Saeed, A. Al-Hamadi, and H. Neumann "Facial point localization via neural networks in a cascade regression framework," *Multimedia Tools and Applications*, pp. 1–23, 2017. (**ISI, Impact Factor= 1.331**)

2) A. Al-Hamadi, A. Saeed, R. Niese, S. Handrich, and H. Neumann, "Emotional Trace: Mapping of Facial Expression to Valence-arousal Space," *British Journal of Applied Science and Technology*, vol. 16, no. 6, pp. 114, 2016.

3) A. Saeed, A. Al-Hamadi, and A. Ghoneim, "Head Pose Estimation on Top of Haar-Like Face Detection: A Study Using the Kinect Sensor," *Sensors*, vol. 15, no. 9, pp. 20945–20966, 2015. (**ISI, Impact Factor= 2.033**)

4) A. Saeed, A. Al-Hamadi, R. Niese, and M. Elzobi, "Frame-Based Facial Expression Recognition Using Geometrical Features," *Advances in Human-Computer Interaction*, vol. 2014, no. 1, pp. 1–13, 2014. (**Scopus Indexed**)

5) A. Saeed, A. Al-Hamadi, and M. Heuer, "Multi-modal Fusion Framework with Particle Filter for Speaker Tracking," *International Journal of Future Generation Communication and Networking*, vol. 5, no. 4, pp. 6576, 2012.

Conference Papers:

6) A. Saeed, and A. Al-Hamadi, "A Framework for Joint Facial Expression Recognition and Point Localization," in *23rd International Conference on Pattern Recognition (ICPR 2016)*, Cancun, Mexico, pp. 4125-4130, Dec. 2016.

7) A. Saeed, A. Al-Hamadi, and S. Handrich, "Advancement in the head pose estimation via depth-based face spotting," in *IEEE Symposium Series on Computational Intelligence (SSCI 2016)*, Athens, Greece, pp. 1–6, Dec. 2016.

8) A. Saeed, and A. Al-Hamadi, "Boosted human head pose estimation using kinect camera," in *IEEE International Conference on Image Processing (ICIP 2015)*, Quebec City, QC, Canada, pp. 1752-1756, Sept. 2015.

9) A. Saeed, A. Al-Hamadi, and R. Niese, "Regression-based Head Pose Estimation in 2D Images," in *1st International Symposium on Companion-Technology (ISCT 2015)*, Ulm University, Germany, pp. 161-166, Sept. 2015.

10) M. Elzobi, A. Al-Hamadi, Z. Al Aghbari, L. Dings, and A. Saeed, "Gabor Wavelet Recognition Approach for Off-Line Handwritten Arabic Using Explicit Segmentation," in *5th International Conference on Image Processing and Communications (IPC 2013)*, Bydgoszcz, Poland, September 2013, Advances in Intelligent Systems and Computing, 2014, pp. 245254, Springer-Verlag Berlin/Heidelberg.

11) M. Elzobi, A. Al-Hamadi, L. Dinges, M. Elmezain, and A. Saeed, "A Hidden Markov Model-based Approach with an Adaptive Threshold Model for Offline Arabic Handwriting Recognition," in *International Conference on Document Analysis and Recognition (ICDAR 2013)*, Washington, DC, USA, August 2013, pp. 945–949.

12) A. Saeed, A. Al-Hamadi, and R. Niese, "The effectiveness of using geometrical features for facial expression recognition," in *IEEE International Conference on Cybernetics (CYBCO 2013)*, Lausanne, Switzerland, pp. 122-127, June 2013.

13) A. Saeed, A. Al-Hamadi, and R. Niese, "Neutral-independent geometric features for facial expression recognition," in *12th International Conference on Intelligent Systems Design and Applications (ISDA 2012)*, Kochi, India, pp. 842-846, Nov. 2012.

14) A. Saeed, A. Al-Hamadi, R. Niese, and M. Elzobi, "Effective geometric features for human emotion recognition," in *11th IEEE International Conference on Signal Processing (ICSP 2012)*, Beijing, China, October 2012, vol.3, pp. 623–627.

15) M. Elzobi, A. Al-Hamadi, A. Saeed, and L. Dinges, "Arabic Handwriting Recognition Using Gabor Wavelet Transform and SVM," in *11th IEEE International Conference on Signal Processing (ICSP 2012)*, Beijing, China, October 2012, vol.3, pp. 2164–2158.

16) A. Saeed, A. Al-Hamadi, and M. Heuer, "Speaker Tracking Using Multi-modal Fusion Framework," in *5th International Conference on Image and Signal Processing (ICISP 2012)*, Agadir, Morocco, pp. 539-546, June 2012.

17) A. Saeed, R. Niese, A. Al-Hamadi, and B. Michaelis, "Coping with Hand-Hand Overlapping in Bimanual Movements," in *IEEE International Conference on Signal and Image Processing Applications (ICSIPA 2011)*, Kuala Lumpur, Malaysia, pp. 238-243, Nov. 2011.

18) A. Saeed, R. Niese, A. Al-Hamadi, and A. Panning, "Hand-face-touch Measure: a Cue for Human Behavior Analysis," in *IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS 2011)*, Zhengzhou, China, pp. 605-609, August 2011.

19) A. Saeed, R. Niese, A. Al-Hamadi, and B. Michaelis, ""Solving the Hand-Hand Overlapping for Gesture Application," in *Image Processing and Communications Challenges 3 (IPC 2011)*, Bydgoszcz, Poland, Springer-Verlag pp. 343-350, September 2011.