Mahdi Champour

Multiview Facial Expression Recognition

DOCTORAL THESIS

to achieve the university degree of
Doktor der technischen Wissenschaften

submitted to

Graz University of Technology

Supervisor

Prof. Dr. Horst Bischof
Institute for Computer Graphics and Vision

Prof. Dr. Lap-Fi Yu
University of Massachusetts Boston

Graz, Austria, Feb. 2016

To my love:

Maryam, Kourosh and Ronia

A new challenge everyday
You keep away and delay;
When I act to close the gap
Fate says there is a bigger play.

Hafiz (1325 - 1389 CE)

Abstract

In this thesis, we describe novel approaches for multiview facial expression recognition by developing mapping ideas. The mapping ideas include linear and non-linear transformations, mapping forests, and inpainting. Our goal is to develop techniques for increasing the precision of multiview facial expression recognition methods.

First, we present our linear solution for mapping non-frontal view facial features to the frontal. This method is equipped by pose specific transformations leading to an efficient approach in terms of both accuracy and time complexity.

Second, analysis of our linear solution shows that, although our linear pose-specific mapping is better than the state-of-the-art methods, it cannot satisfy some of the variations due to the non-linear behavior of the multiview facial expression recognition problem. Therefore, we propose using non-linear kernel based mappings for each specific head pose. Our non-linear mappings are driven by training data that adopt themselves with the problem of multiview facial expression recognition accurately. By improving the performance of non-linear mappings, we motivate to employ much more accurate mappings.

Third, inspired by random forests method, we propose the mapping forests method, which performs non-linear mappings in such a way that the mappings are automatically generated. As the non-linearity of the transformations are determined automatically using the forests (trained data), the results are much more accurate even using raw data.

Finally, we employ 3D face models for frontalization. The frontalized faces have some unavailable or occluded regions that we aim to inpaint by means of supervised methods. Our inpainting approach relies on exploiting a guidance face which is selected from training data based on the facial attributes of the input face image.

All the proposed approaches are extensively evaluated with several popular datasets and compared with the state-of-the-art methods, in order to show their practicality and performance. From the experimental results, we show that our approaches outperform the state-of-the-art methods.

Kurzfassung

In dieser Arbeit befassen wir uns mit der automatisierten Erkennung von Gesichtsausdrücken mittels mehrerer Kameraansichten. Dabei beschreiben wir neue Ansätze basierend auf einer Reihe unterschiedlicher Mapping-Ideen. Im Speziellen behandeln wir lineare und nichtlineare Transformationen, Mapping-Forests und Inpainting-Methoden. Unser Ziel ist es, die Genauigkeit bestehender Methoden zur Mimik-Erkennung zu verbessern.

Zunächst führen wir eine lineare Transformation ein, um die Merkmalsvektoren nicht-frontaler Gesichtszüge in eine kanonische, frontal ausgerichtete Ansicht abzubilden. Durch die Verwendung von posen-spezifischen Transformationen können so sehr genaue Ergebnisse erzielt werden. Ein weiter Vorteil dieser Transformationen ist, dass sich diese Abbildungen sehr effizient durchführen lassen und dadurch eine geringe Zeitkomplexität aufweisen.

Obwohl diese linearen Transformationen bereits deutliche Verbesserungen gegenüber dem aktuellen State-of-the-Art erreichen, können damit nicht alle Gesichtsposen gleichermaßen genau in die kanonische Ansicht abgebildet werden. Der Hauptgrund dafür ist, dass die Multiview-Mimik-Erkennung im Allgemeinen ein nicht-lineares Problem darstellt. Um diese Limitierungen zu adressieren, präsentieren wir eine zweite Methode basierend auf nichtlinearen Kernel Mappings. Durch solche adaptiven, posen-spezifischen Transformationen ist es uns möglich, eine exaktere Abbildung auf die kanonische Ansicht durchzuführen.

Als weitere Verbesserung führen wir Mapping-Forests, basierend auf dem Random Forest Framework, ein. Die nichtlinearen Transformationen werden dabei direkt im Trainingsschritt gelernt. Damit ist es uns möglich, die Abbildungen direkt anhand der Eingabebilder ohne explizite Featureextraktion zu bestimmen. Wie aus den Evaluierungen ersichtlich wird, können mit Hilfe dieser Mapping-Forests viel genauere Ergebnisse erzielt werden.

Abschließend zeigen wir, wie 3D Gesichtsmodelle benutzt werden können, um unterschiedliche Gesichtsposen in die kanonische Ansicht zu transformieren. Aufgrund der unterschiedlichen Blickwinkel können jedoch nicht alle Bereiche der kanonischen Frontalansicht nach der Transformation mit Bildinformationen aufgefüllt werden. Inspiriert durch die großen Fortschritte im Bereich des Inpaintings, verwenden wir diese Methode, um die verdeckten Gesichtsbereiche aufzufüllen. Als Schablone dient dabei ein Gesichtsprototyp, der anhand der bestimmten Gesichtsattribute aus einem Datensatz extrahiert wird.

Die Vorteile der gezeigten Verbesserungen und Methoden werden in einer Vielzahl von Experimenten auf öffentlichen Datensätzen gezeigt, um einen fairen Vergleich zu ermöglichen. Die Resultate zeigen, dass unsere Lösungsansätze die Genauigkeit der derzeit besten Methoden übertreffen können.

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used.

The text document uploaded to TUGRAZonline is identical to the present doctoral thesis.

Date

Signature

Acknowledgments

First of all I would like to thank Allah, the most gracious and the most merciful. This work was supported by the Iran Ministry of Science, Research and Technology (MSRT), I would like to thank for the generous scholarship support I received.

I am grateful to my supervisor Professor, Horst Bischof, for his advice, patient guidance and giving me the precious opportunity to work at the prestigious Institute for Computer Graphics and Vision at TU Graz. He is always welcoming for my questions and also gives me complete freedom in choosing my research direction.

I would also like to thank my second supervisor, Professor Lap-Fai (Craig) Yu at the University of Massachusetts Boston, for his valuable comments and advice on my thesis.

I am also grateful to my other colleagues from the ICG: Professor Vincent Lepetit for his awesome help and discussion; Thomas Mauthner who is not only a colleague but also a friend; our collaboration is wonderful and enjoyable; Dr. Jonathan Ventura who always gives me motivation and discusses with me; Horst Possegger and Georg Waltner whom I collaborated closely with; they always helped me during my study in Austria; Dr. Christian Reinbacher, Dr. Samuel Schulter and Dr. Peter Roth for their assistance.

Furthermore, this work would not have been possible without the support of several people from the Iran MSRT and the University of Kerman. I would also like to thank Professor Hassan Moslemi-Naeini, and Professor Mohammad Masoud Javidi. I would like also to thank the Graz University of Technology for awarding two competitive fellowships (Marshall Plan Scholarship and KUWI) to me during my study at Graz from 2011 to 2016.

And finally I would like to thank my love, Maryam, who experienced a lot of gaps and difficulties during our life and living outside of our home country while I was studying. I would also thank my children who missed their father at home for several years. I would also like to thank my grand family: my father, mother, brothers and sisters for being compassionate. Without their support I would never have finished this work.

Contents

List of Figures

List of Tables

1 Introduction

In the last decades the importance of artificial intelligence and especially its subbranches like computer vision has been increased attracted attention. Various problems such as recognition, detection, identification, segmentation, etc. can be currently solved by exploiting intelligent systems and computer vision techniques. They are used in applications like medical image analysis; navigation; traffic controlling; human identification; quality control and hundreds other examples. An important task of computer vision is *recognition* that can analysis images or videos to understand about the objects inside or even discriminant available objects. There are various tasks regarding to the human and human faces handled with computer vision in terms of *recognition.* For instance, face recognition, face parts recognition, facial expression recognition, etc. In general, the problem of *recognition* focuses on deciding weather our goal object/activity is within the scene or no? With such knowledge we can make this ability for the machines that can see, think and decide about the contents. One of the popular and useful analysis on the face content is recognizing the human face expressions. It has several benefits and applications like human computer interface (HCI), child care, aid to meliorate the specific diseases such as Autism, etc.

The problem of facial expression recognition (FER) focuses on the image(s) or video(s) containing human faces to understand what the human expression is. This leads to realize the human emotion which is a significant finding to make the communications between machines and human.

A new challenge and an extended form of facial expression recognition (FER) is multiview facial expression recognition (MFER) which is able to recognize human facial expressions even from non-frontal faces. In fact, it is an unrealistic assumption if we assume that the human faces are always frontal (especially in the wild), therefore, developing new ideas that can improve FER systems to MFER is always desirable. However, this is not a naive task due to the lack of human face parts in non-frontal view. Nevertheless, in this thesis we address this problem by compensating these unavailable parts.

1.1 Motivation

Multiview facial expression recognition (MFER) is currently one of highly active area within computer vision community due to the potential demand. For instance, in human computer interface (HCI) which is an interdisciplinary branch of AI between human science and computer science the computers can discover human emotion by exploiting MFER. In HCI it is a part of a larger system like robots for communicating with human or performing services such as education, etc.

Another significant potential demand on MFER is in psychology and child care, especially with the specific diseases such as Autism. Currently about one percent of the world population have autism spectrum disorder [111] which means about 70 million people, that is equal or greater than the population of the countries like France, or UK. It is a strong motivation for us to explore new ideas for improving systems that can help such people. The goodness is that this conditions can be resolved by recognizing autistic children and treatment them. A prospective solution is to make an interactive environment (e.g. computer applications), such a way that autistic children could interact and train to represent their behavior.

The Games based on the recognized expressions and human emotion is another task in this direction, which can be planned based on the psychological treatments or educational purpose or any other targets. Among the games directory such applications are always requested to follow specific purposes.

1.2 Where the problem of MFER comes from?

There are several applications that can successfully recognize facial expressions (FER). Nevertheless, an important gap within these applications is that they are works only with the frontal faces. This means that if captured faces are not frontal, the system cannot recognize the expressions whilst human can do. In other words, if the current systems are installed in a real world condition (e.g. in a room of kindergarten), it cannot recognize the children behavior most of the time. Unless, the child face become exactly in front of the installed camera(s). In fact, this is insubstantial assumption if we presume that captured data are frontal. On other side, it is very difficult and expensive if we use lots of cameras to detect frontal faces. Therefore, the problem of multiview facial expression recognition comes from that detected faces are not always frontal and current facial expression systems cannot handle facial expressions from different viewpoints.

1.3 MFER challenges and difficulties

MFER is still an open problem due to the various challenges. On one side, there are some general challenges regarding to the FER and on the other side, the challenges related to the arbitrary views that affect on the recognition purposes. Some of these challenges are almost solved but some are still open. We can summarize them into the general categories as:

1) The problem of facial expression recognition has not been solved yet very well. Although, several approaches have been recently introduced that can address the problem of FER with appropriate accuracy (almost better than 95% in different datasets [2, 28, 48]) but still have errors.

2) Privation of desirable multiview data. The early works tackled to the problem of MFER on 2008 by introducing the first multiview facial expression dataset. Recently, several valuable datasets have been provided for the problem of MFER that encourage researchers on this problem.

3) Absence of powerful methods to handle multiview faces; A few works started to address this problem with variant methodologies but it is not solved yet. Therefore, developing efficient approaches towards facial expression recognition in arbitrary views is aimed by this research.

Nevertheless, there are lots of difficulties behind these challenges to recognize facial expressions from arbitrary views. They are including: 1) General difficulties, that do not concern about frontal or non-frontal views such as gender, ethnicity, age, skin tone, lighting conditions, etc. The other group is related to the 2) Non-frontal faces difficulties. For instance, How much and which parts of the face are unavailable? Are unavailable parts including the facial components or no? etc. In the following we mention to these variations:

Gender. As can be seen in Figure 1.1(a) there are differences between male and female faces. These variations are including the facial parts and their properties which can increase the ambiguity of the recognizing process. An appropriate MFER system must be able to address this difficulty and decrease the discrimination due to the gender otherwise these undesirable variations can affect on the expression recognition wrongly.

Ethnicity. There are many ethnicities in the world that they have similar facial structures but almost different details. For instance, the variations between an Asian face and African are as much that our system cannot easily ignore these variations and focus only on the expressions. Therefore, it is reasonable if we see that our classifiers decide wrongly based on these variations instead of expressions. Figure 1.1(b) shows different ethnicities.

Age. Another difficulty that can affects on the expression recognition is age. A supervised method learn often a model to determine the differences and ignore the similarities. For such a good model we need to learn our system with a wide range of age which is almost impossible or very difficult. Three age-variant samples are shown in Figure 1.1(c). Again the variations due to the age can affect on the extracted features. For example, an old person has lots of wrinkle on the face (cheek, forehead) that are not available on the face of a young person, therefore, this is difficult for a computer to realize such variations and decide only based on the expressions.

Skin tone. A wide range of variations on the face is related to the skin tone, it is originally because of ethnicity, human features or due to the capturing system (e.g.

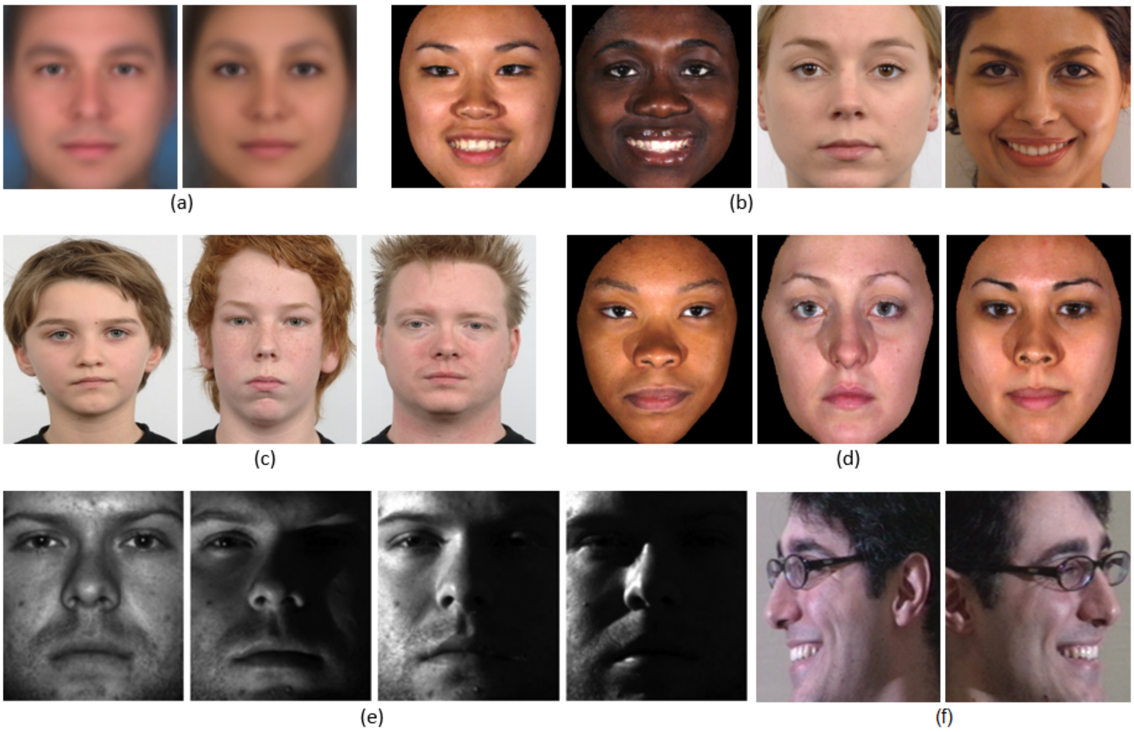


Figure 1.1: Multiview facial expression recognition difficulties. Variations on (a) Gender, (b) Skin tune, (c) Age, (d) Ethnicity, (e) Illumination and (f) Head pose.

camera intrinsic). These variations make our expression recognition much more difficult, therefore, MFER system must be able to decide using the information that they are stable with variant skin tone otherwise a vast range of skin color, absolutely affect on the extracted features. Three samples with different skin tone are illustrated in Figure 1.1(d).

Lighting conditions. Another complicated challenge on the facial analysis is lighting conditions. In fact, the facial features are strongly depend on the lighting conditions such a way a poor lighting system almost fails with extracting correct features, therefore, these features may lead to the wrong recognition. Figure 1.1(e) shows a face with different lighting conditions.

Head pose An important challenging variation on the face analysis is head pose. It affects on almost all of the face algorithms and applications due to the changes on the face structure. For instance, on the problem of face detection, an usual hypothesis is browsing for two eyes, a nose, a mouth, etc. whilst if the input face is non-frontal, our simple hypothesis will be failed due to the unavailable regions (e.g. part of mouth or an eye). Considerable samples on beside faces are illustrated on Figure 1.1(f).

Which parts of the face are unavailable? This is a substantial challenge on non-frontal facial analysis. If unavailable parts are including the facial components especially important components (e.g. mouth), we can expect a significant fails with the recognition. Therefore, it is valuable if we can provide the capability of compensating such regions. In

this thesis we focus on approaches to compensate facial parts missed due to the head pose.
How much of the face is unavailable? Some parts of the face are not visible in a non-frontal faces because of the head pose. It is serious that how much is the size of invisible parts. If the size is small, we may compensate them by exploiting neighbors or facial symmetric features, otherwise we may need to provide more complicated approaches. More discussion is to be found in chapter 4.

In this thesis, we focus on proposing approaches that can resolve the problem of handling non-frontal faces, while our approach are almost able to reach other kind of the difficulties. For instance, by employing HOG features [19] we can dispel the lighting variations due to the HOG robustness on the illumination conditions.

1.4 Main Contribution

By the current research, we provide several approaches based on mapping ideas for multiview facial expression recognition. While an efficient FER system needs to the whole of the face we have to provide unavailable parts of non-frontal face image. We have done it by mapping approaches which perform this ability to estimate correspondence facial features in frontal view. They are more efficient for purpose of expressions recognition.

To this end, we propose 1) Linear mappings on raw, basic and sparse features, including general and pose specific transformations; 2) Non-linear kernel-based mapping that is an extended version of linear mapping; 3) Mapping Forests, which is a very efficient approach for providing non-linear mappings such a way that mapping parameters have been determined automatically by exploiting forests; and finally 4) mapping raw data based on the 3D face model where we generate a frontal face from input non-frontal using available regions and propose inpainting for unavailable parts. By providing our approaches, our system can perform multiview facial expression recognition.

1.5 Outline

Multiview facial expression recognition (MFER) is an active problem in computer vision. It is an extended form of facial expression recognition (FER) that addresses to recognize expressions even from non-frontal face images. We arrest attention MFER by proposing mapping approaches to provide frontal faces/features as their processing has currently appropriate outcomes in terms of facial expression recognition.

This thesis is organized as follows: Section 2 presents a literature review on the related works and the problem background, it contains also with introducing on the state-of-the-art. In chapter 3, we have proposed our system pipeline including problem statement on our MFER system. Moreover, our MFER system is described in details where it is containing the face detection, feature extraction, feature dimension reduction, feature encoding and frontal facial expression recognition. Chapter 4 is focused on mapping approaches including our contributions by variant transformations. Finally, our experimental results and comparisons are presented in chapter 5 and chapter 6 is specified for summary and conclusion.

2 Related Work and Background

Contents

This thesis aims for multiview facial expression recognition. In this chapter, we introduce the related concepts and provide an overview on the state-of-the-art regarding the frontal and multiview facial expression recognition. We first introduce the notations and review briefly on the problem of face detection. The history of facial expression recognition will be reviewed in subsection 2.3. We then present the concepts of automatic facial expression recognition including specific properties, advantages and weaknesses in section 2.4. Multiview facial expression recognition is reviewed then in section 2.5. Moreover, random forests, regression forest and 3D morphable model are other topics that we used during our research, therefore, we review their concepts at the end of this chapter.

2.1 Notation and Conventions

Before starting the literature review, we introduce the mathematical notations which are used throughout the thesis. Scalar values are represented by italic fonts, e.g. x or c_i. Matrices and vectors are depicted in bold font, e.g. $\mathbf{M}$ or $\mathbf{v}$. Vector spaces are depicted in double-lined upper case letters, e.g. $\mathbb{R}^3$ or $\mathbb{Z}^3$. Functions, mapping between different

Entity	**Notation**
Scalar	a, c_i
Vector in 2D	$\mathbf{v} = (x, y)^{\mathrm{T}}$
Vector in 3D	$\mathbf{X} = (x, y, z)^{\mathrm{T}}$
Matrix	$\mathbf{M} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$
Vector Space	$\mathbb{R}^3$
Mapping Function	$\mathcal{P} : \mathbb{R}^3 \rightarrow \mathbb{R}^2$

Table 2.1: List of notations used in this thesis.

spaces are given in upper case calligraphic letters, e.g. $\mathcal{P}$ or $\mathcal{H}$. An overview over the notation is given in Table 2.1.

2.2 Face Detection

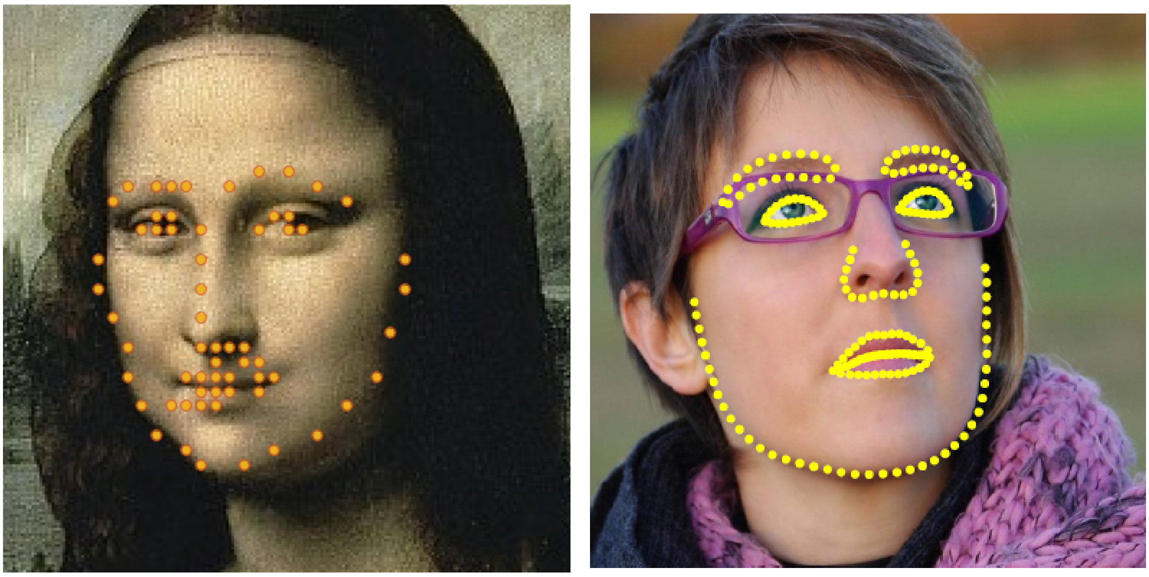
For many years the problem of face detection was one of the hot topics within computer vision community. The first real time attempt proposed by Viola and Jones [124] with two important characteristics:

1. Real time: It was the first framework capable to detect the face in real time.
2. Robust: True positive rate is high and the false positive rate is always very low.

The Viola and Jones framework has mainly 4 stages:

1. Haar features selection
2. Creating integral image
3. Adaboost-based training algorithm
4. Cascaded classifiers

All of these stages are reasons for above effective characteristics, for instance the idea of integral image increases the speed of feature evaluations significantly. Moreover, the Haar features are very fast and effective features for this purpose. Today the problem of face detection is focused on further challenges by extending Viola & Jones face detector. Detecting human faces from arbitrary head pose is one of the open challenges in this direction. Nevertheless, a few approaches recently achieved high precision on face detection even from non-frontal face images. For instance, [59] and [144] are two approaches that provide face detection even from non-frontal head pose by estimating the facial landmarks. Therefore, we used both of them (alternatively) for our multiview facial expression recognition system.



(a) By Zhu and Ramanan

(b) By Kazemi and Sullivan

Figure 2.1: Response of employed face landmarks detector (even from non-frontal samples).

[59] proposed a cascade of regressors with ensemble of regression trees that is not only very efficient in terms of time complexity but also improved the landmark detection error rate. On the other hand, [144] is based on a mixtures of trees with a shared pool of facial parts; every facial landmark modeled as a part and an effective tree-structured model proposed for global mixture to capture topological changes due to the viewpoint. Figure 2.1 illustrates the response of both employed face (landmarks) detectors.

2.3 Facial Expression Recognition, the History

Human facial expressions are efficient visible non-verbal communicating of the affective states. Facial expressions occur due to the facial muscles movements beneath the skin. These movements compose human emotions that transfer individual states [98]. Facial Action Coding System (FACS) which was the first system for classifying human facial muscles movement based on the face appearance proposed by Paul Ekman [26] to study the human facial expression.

Ekman focused on the facial muscles and categorized them based on each specific expressions to generate an automatic system for recognizing facial expressions: “A new method of describing facial movement based on an anatomical analysis of facial action” said Paul Ekman about FACS [128]. Nevertheless, expressions have overlaps with each other on some of the muscles. Table 2.2 illustrates some action units based on the Ekman and his colleagues’ researches [16].

Ekman and Friesen [26] introduced six basic expressions including: anger (AN), disgust

(DI), fear (FE), happy (HA), sadness (SA) and surprise (SU) which are in contrast of neutral (NE) that are related to the action units. For instance, Table 2.3 shows action units that represents these six basic facial expressions [32].

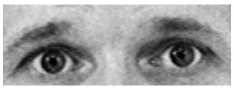
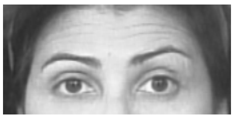
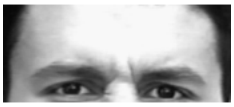
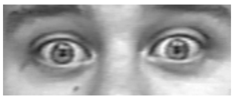
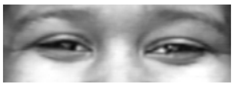
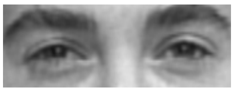
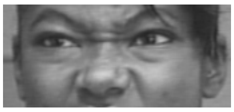
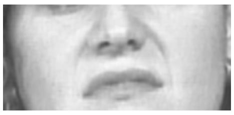
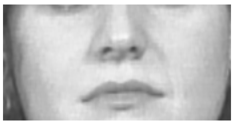
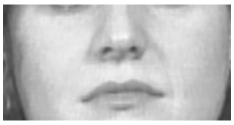
AU#	*Description*	*Facialmuscle*	*Exampleimage*
1	*InnerBrowRaiser*	*Frontalis, parsmedialis*	
2	*OuterBrowRaiser*	*Frontalis, parslateralis*	
4	*BrowLowerer*	*Corrugatorsupercilii,*	
5	*UpperLidRaiser*	*Levatorpalpebraesuperioris*	
6	*CheekRaiser*	*Orbicularisoculi, parsorbitalis*	
7	*LidTightener*	*Orbicularisoculi, parspalpebralis*	
9	*NoseWrinkler*	*Levatorlabiisuperiorisalaquaenasi*	
10	*UpperLipRaiser*	*Levatorlabiisuperioris*	
11	*NasolabialDeepener*	*Zygomaticusminor*	
12	*LipCornerPuller*	*Zygomaticusmajor*	

Table 2.2: Ten Action Units (AUs) introduced by Cohen et al.

As can be seen in the Table 2.3, there are units that influence on more than one emotion, therefore, this is the reason that automatic facial expression recognition is a complex task with ambiguities. In other word, to define the problem: Facial expression recognition is a challenging problem due to the complex emotions and similar muscles movement for different expressions. The goal is providing systems that can be able to recognize facial expression automatically even from non-frontal viewpoints. In the following we first review the concepts that address the problem of facial expression recognition in frontal view.

$Emotion$	$ActionUnits$
$Anger$	$4 + 5 + 7 + 23$
$Contempt$	$R12A + R14A$
$Disgust$	$9 + 15 + 16$
$Fear$	$1 + 2 + 4 + 5 + 7 + 20 + 26$
$Happiness$	$6 + 12$
$Sadness$	$1 + 4 + 15$
$Surprise$	$1 + 2 + 5B + 26$

Table 2.3: Six basic facial expressions + 'Contempt' and related action units.

2.4 Facial Expression Recognition

The problem of facial expression recognition has been analyzed with various approaches in the last three decades. An overview on the most of the approaches shows that they are based on the learning methods that trained features (raw, low level or encoded features) and recognized the expressions using different classifiers. Therefore, a general schema for the proposed solutions illustrated in Figure 2.2.

In general, the problem of facial expression recognition can be categorized into three categories: 1) Geometric approaches, 2) Appearance-based approaches which can again categorize into several variant sub-categories and 3) Hybrid methods. In the following we briefly introduce them:

2.4.1 Geometric Approaches

Geometric models are approaches that use geometric information of the face, including landmarks, shape or facial action units (AUs). For instance, a geometric-based approach included regression-based of different mapping functions of geometric features proposed by Rudovic et al. [95] or the work of Hu et al. [41] which used 2D facial points to calculate the geometric 2D displacement of facial features between expressions and neural state at the corresponding angles. Moreover, InteraFace (IF) [21] is another specific effort on facial expression recognition which works based on the action units and not only recognize the expression from images but also from videos. The advantage of geometric approaches is their accuracy on the facial parts variations (e.g. facial landmarks movements, shape alignment, etc.).

2.4.2 Variant Descriptors and Classifiers

In contrast to the geometric-based approaches there are approaches that use feature descriptors to represent facial features. They have been categorized in appearance-based

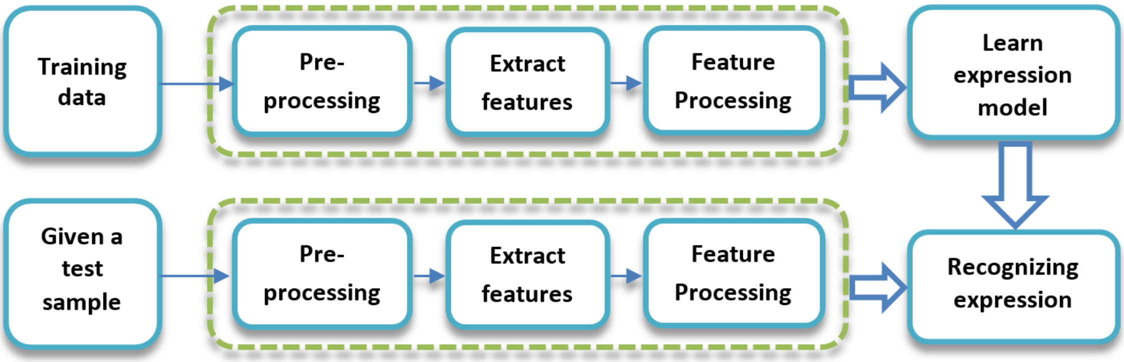


Figure 2.2: Overall structure in most of the facial expression recognition frameworks.

approaches. Multiple efforts on facial expression recognition with variant descriptors are reported. However, it is not very clear that which descriptor is the most efficient due to the different protocols, setup system and mixed contributions but there are a few works that provided comparisons between the several feature descriptors. For instance, Hu et al. [40] provided a comparison between three feature descriptors LBP, HOG and SIFT. They showed that SIFT has better accuracy than other descriptors however, this improvement is negligible. Another study by Hesse et al. [38] evaluated SIFT, LBP and DCT that extracted from around of facial landmarks. Their results show that DCT features yield better performance than SIFT and LBP. An interesting point by means of computer vision feature descriptors is proposing new descriptors, for instance, the following descriptors are introduced for facial expression recognition: Local Directional Pattern (LDP) [50], Local Gabor Binary Pattern (LGBP) [85], Pyramid Histogram of Orientation Gradients (PHOG) [61], Pyramid Local Phase Quantization (PLPQ) [126], Weber Local Descriptor (WLD) [135], Local Directional Pattern Variance (LDPv) [56], Gradient directional pattern (GDP) [2], Local Monotonic Pattern (LMP) [84], Local Gabor Directional Pattern (LGDP) [49] Local Gabor Binary Patterns (LGBP) [3], etc.

They are descriptors that proposed first time for the problem of facial expression recognition. Nevertheless, the most popular descriptors for facial expression recognition are SIFT, HOG, Gabor and LBP. Figure 2.3 shows an example of extracted HOG features. The similar story is for employing variant classifiers. Almost of the approaches for the facial expression recognition are supervised methods, they learned via training data and used a classifier for final decision on the test sample. Therefore, a wide range of classifiers are used for this purpose. For instance, the following are the most popular used classifiers: K-Nearest neighborhood (KNN) [135], [49], [115] Linear Discriminant Analysis (LDA) [47], [105] Support Vector Machine (SVM) linear/kernel based [126], [3], [50], [2] AdaBoost / Gentle Boost classification [6], [69] Supervised Soft Vector Quantization (SSYQ) [117] Neural Network Classifiers [112], [133] Bayes Classifiers [4] [60] Maximum Likelihood Classifiers [88], etc. A comparison of classifiers for facial expression recognition provided by [61], [6].

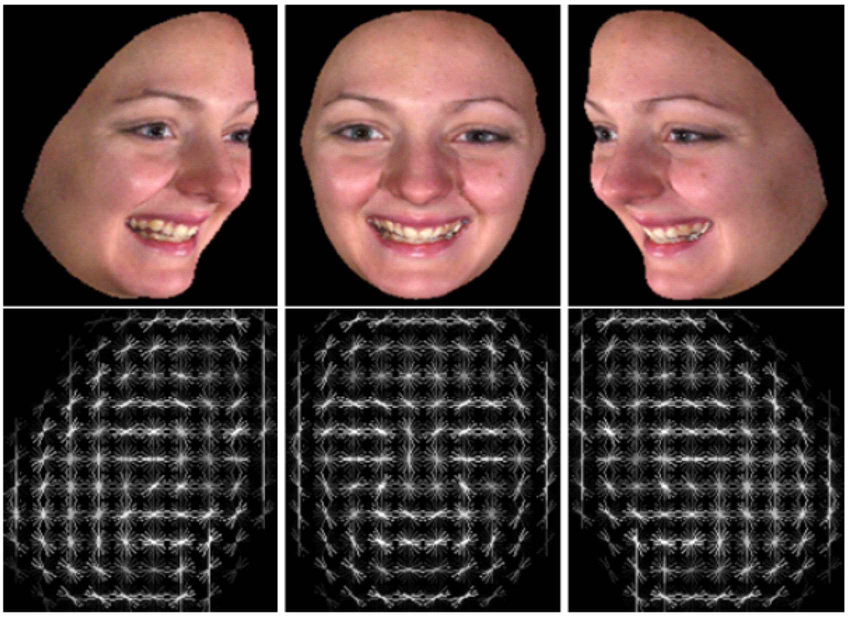


Figure 2.3: HOG features: a popular descriptor for facial expression recognition.

2.4.3 Encoding Approaches

Encoding approaches are recently successful ideas for facial expression recognition systems. There are multiple works that encode data aiming to improve the accuracy. In general, almost of the encoded approaches have a pre-processing step that encode data into a new feature space (e.g. sparse coding), this new space(s) has often properties that help to improve the recognition process. The important motivation for encoding approaches is that the new mapped space is easier, smaller or more discriminant than before for recognition purpose (e.g. decreasing the outliers). For instance, approaches based on principle component analysis (PCA) that decrease the dimensionality of the features space (subsequently, decrease the outliers) to make an easier analytic space for classifier. Several employed encoding approaches are: Principle Component Analysis (PCA) [12], [131] Canonical Correlation Analysis (CCA)/Kernel CCA [33], [46], [101] Sparse Coding [116], [30] Regression Transformation [53], [95], etc.

2.4.4 3D Models

Recently, with increasing the availability of devices capable to capture high resolution 3D objects, the approaches have been introduced for recognizing facial expressions from 3D faces [20, 47, 102, 105]. 3D morphable model (3DMM) [10] is one of the prominent approaches for reconstructing 3D surface from 2D facial images that is a statistical representation of both texture and shape of human face. It has been shown that 3DMM is an efficient approach for extracting 3D facial surface and texture from images [102]. How-

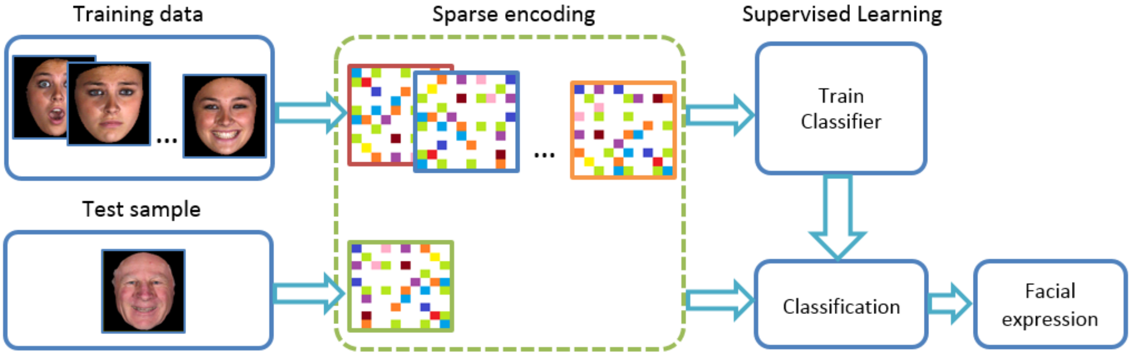


Figure 2.4: Basic (sparse) encoding framework.

ever, it is very sensitive to the occlusion and depends on initialization. Another successful category relies on mapping models, for instance, Huang et al. [47] provided a 3D mapping approach including: displacement mapping and point-to-surface mapping that reconstruct the facial 3D surface. While their point-to-surface mapping is a regional transform, it is capable to preserve the facial deformations which is the most important advantage of the 3D facial expression recognition approaches.

2.4.5 Video-based Systems

In another aspect, facial expression recognition has been investigated in terms of video-sequences. A prominent methodology relies on employing spatio-temporal descriptor which is able to exploit variation on sequence of frames. For instance, Fan and Tjhajadi [27] proposed spatio-temporal domain that provides 3D facial features, then integrated them with dense optical flow and provided a descriptor that is capable to extract both spatial and dynamic motion of facial expression. Another category on video-based facial expression recognition system is approaches that utilized from depth information. However, the result of depth-based approaches is not as well as traditional methods that use 3D mesh but depth-based approaches do not need high resolution data which is required for 3D mash approaches. Therefore, there are attempts on employing depth information for video-based facial expression recognition that do not need high resolution data. For instance, Shao et al. [106] provided a similar approaches on low-resolution videos. Also, the method of Zia Uddin [123] on depth video that is an appearance-based approach and used local directional pattern (LDP) categorized with the same category. Nevertheless, a wide range of video-based approaches benefit from temporal variations on facial parts which means they make decision on the features that originally generated from landmarks displacement [113].

2.4.6 Hybrid Models (Geometric and Appearance)

Without any doubt, the multimodal approach that exploits both geometric and appearance based information is one of the best methods for facial expression recognition. Although, the appearance based approach try to explore the facial expression variation but the other appearance features (e.g. skin tone) unexpectedly affect on expression recognition. In other words, while we expect the same expression from two different subjects with similar emotion, the result of appearance based recognition is different due to the variation on skin tone. On the other hand, geometric approaches suffer from the lack of facial appearance data. Therefore, it motivates to exploit hybrid models. For instance, the approaches that use both 3D shape and texture are state-of-the-art. Very recently publication by Hubin Li et al. [72] combined local texture and shape description; they benefit from information around of landmarks on both 2D and 3D data. Several other works with the same approach used the differences of local texture information and the displacement of geometric facial points between neural and expressive facial expression images [13, 99]. Figure 2.5 illustrates 3D shape and detected landmarks on 'neural' and 'happiness' expression with their textures.

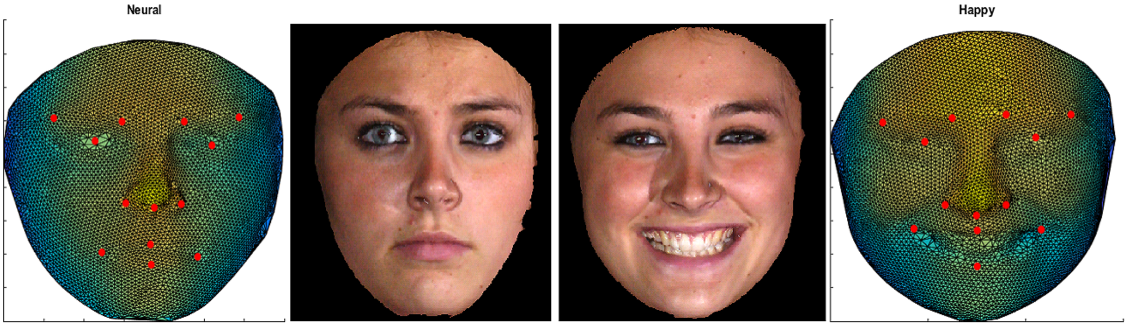


Figure 2.5: 3D Shapes and textures on 'neural' and 'happiness' with landmarks.

2.4.7 Other Approaches

A very recently approach to address the problem of facial expression recognition is based on Deep Learning. As there are several parameters (e.g. classifier, feature descriptor, feature selection, etc.) trough the solutions, it is always desirable to perform an automatic methodology to balance these parameters. One of the most important advantages of deep learning frameworks is that the highly complex features (parameters) can be learned according to the relative importance of the training data. This capability of deep learning leads to create multimodal methods and balances the parameters. Wei Zhang et al. [134] and Ping Liu et al. [77] introduced similar approaches where the former learned a joint representation of texture and landmarks modality of facial images, as mentioned before, the most successful approaches are methods that benefits from both appearance and geometric information. Also, Ping liu [77] proposed Boosted Deep Belief Network to perform feature learning, feature selection and classifier construction. They showed that their deep learning based approach handled a complex system very well.

2.4.8 Discussion on the FER techniques

Review on the FER techniques show that it is an interesting problem which has been well studied with various techniques. The works are studied based on geometric information presume that geometric information is much important to discover the expressions. Whilst, appearance-based approaches believe that the texture information has more influence for recognizing expressions. Recently studies used both geometric and appearance based information. Their motivation is based on that the facial expressions affect on both facial parts geometry and textures. For instance, for a happy face the variations on the mouth geometry are very clear while the pixels details around of the mouth have been affected with such action. Nevertheless, other techniques (e.g. encoding approaches) shows that processing on the raw data only is not enough efficient. It is because of outliers and complication of raw data domain space that do not allow to the classifiers that discriminant data very well. Therefore, proposed ideas transform raw/feature space into the more efficient space. The new spaces are almost simpler and easier than before that can improve the classification process. This simplification process can be done by accessibility to the more information, for example, depth information in 3D based approaches that is a reasonable idea can help to the FER techniques to have better recognition. However, this increasing information can rise the processing complexity but they are very useful information (depth) for expression recognition purpose.

While the problem of FER is analyzed and evaluated with variant techniques, another view to this problem is if we use dynamic information during the time. In fact, a sequence of frames can be very useful for expression recognition. Almost of these methods benefit from geometric information and use variations on the landmarks or facial key parts.

In general, all of the FER techniques try to compute and discriminant features which are serious to capture the facial parts variations. Two important research items among the introduced FER systems are: 1) Extracting efficient features or augment the features from the face, 2) Providing a good discrimination strategy. By improving one or two of these items a FER system can be improved.

2.5 Multiview Facial Expression Recognition (MFER)

After three decades from beginning to address the problem of facial expression recognition (FER), today there are several approaches that can provide appropriate results for the problem of FER. Therefore, other new challenges are introduced based on the importance of them in real applications. One of the most important related challenges is the capability of recognizing facial expression from non-frontal faces that made new keywords and studies trough the computer vision community. The terminology of 'non-frontal' facial expression recognition is very close and equivalent to the words of 'multiview', 'arbitrary view', 'pose free' or 'view invariant' facial expression recognition. All of these phrases refer to the processing of recognizing facial expression where the input face image must not be taken from frontal view like the samples in Figure 2.6. Nevertheless, there are several approaches with variant methodologies to address multiview facial expression recognition. By the following we review the top related publications in details.

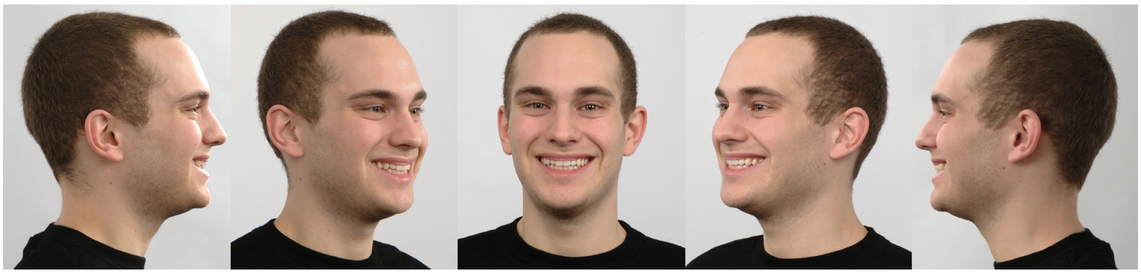


Figure 2.6: Non-frontal samples for facial expression recognition.

2.5.1 MFER baseline

One of the first attempts regarding to the facial expression recognition in non-frontal view proposed by Hu et al. [41], they motivated to research on a new unexplored research of non-frontal facial expression recognition with accessibility to a 3D facial expression dataset (BU3DFE).

They answered to the question that weather non-frontal facial expression analysis can achieve the same or better performance than frontal? To this end, they used the information of face landmarks (eyes, eyebrows, mouth, etc.) and geometric displacement in compare with the neutral faces. They detected Landmarks manually and 2D displacement computed between emotional and neutral expressions of the same person at different view angles.

They employed various classifiers (Linear and Quadratic Bayes Normal Classifier, Parzen Classifier and Linear SVM) to classify the normalized displacement vectors and showed that SVM outperforms other classifiers. In that work, authors showed that their geometric model for non-frontal facial expression recognition can achieve the performance of the same problem in frontal face. Although, today the performance of the frontal facial analysis is very efficient and the problem of frontal facial expression recognition is almost solved but Hu's results comprehensively showed that non-frontal faces have more or less similar information to the frontal. Nevertheless, we have to mention that their geometric displacement is manually selected where non-frontal landmark detection is still an open question in computer vision.

Moreover, the same authors in another work [40] evaluated an appearance based model and showed the influence of different feature descriptors and dimension reduction methods, they provided a comparison between them with a basic view-dependent method. Authors employed HOG, LBP and SIFT as three popular feature descriptors to extract and describe the facial features. They also analyzed PCA, LDA and LPP dimension reduction methods with nearest neighbor classifier. They showed that a combination of SIFT descriptor with LPP dimension reduction performs the best in comparison with other possibilities.

A very basic comparison of two baseline works of Hu et al. [40, 41] shows that their appearance based approach has better performance than geometric approach.

2.5.2 Appearance-based MFER Approaches

The problem of non-frontal facial expression recognition followed by other researchers with variant methodologies, the works that benefit from appearance information. Most of these works developed the discrimination process, for instance, [141] and [142] developed non-frontal facial expression recognition system by means of Bayes theory. They introduced the problem of facial expression recognition as an optimization problem that minimized the upper bound of estimated two-class Bayes error and then extended it for multiclass purpose whilst they employed SIFT features. They reduced the dimension of features by means of Bayes Discriminant Analysis (BDA) and Gaussian Mixture Model (GMM). Nevertheless, as the appearance based methods have large scale feature vectors, most of them need to the dimension reduction. Principle Component Analysis (PCA) and Bayes Discriminant Analysis (BDA) are two popular algorithms for this purpose [142], [131].

The earlier works on multiview facial expression recognition started based on the geometric approach but the most progress is done based on the appearance based methods. For instance, Moore and Bowden [85] proposed an appearance based multiview facial expression recognition. Their method relies on pose dependent facial expression recognition. While Hu et al [40] showed before that non-frontal facial expression has similar performance as frontal, Moore and Bowden proposed their method that simply outperformed significantly the baseline. Their method relies on classifying non-frontal facial samples into the several categories based on the head pose then employed classifier for each specific category individually. Moore's method [85] considers originally two regressors for:

1. Head pose classification

2. Facial expression recognition

Of course their method increases the number of classifications, nevertheless, it enhances not only the recognition rate but also decreases the time complexity of the classifications due to the smaller training data during the learning. Therefore, categorizing the problem of multiview facial expression recognition based on the head pose into the several smaller problems that has same head pose constraint could be an efficient idea that improve the performance of the solutions. Moore and Bowden used the same idea with other contributions in [86]. They introduced also a new feature descriptor namely Local Gabor Binary Patterns (LGBP) which inspired from LBP and it is an extended version of Gabor filters with advantages of multi-resolution and multi-orientation decomposition of facial images.

Variant descriptors have been investigated via other appearance based approaches. SIFT, LBP and DCT are evaluated descriptors by [38]. They also exploit head pose-dependent MFER that used AAM model for detecting facial landmarks and the appearance features around of detected landmarks. An important achievements by Hesse et al. was that concatenating shape coordinate with appearance features even with the simplest way, can improve the final recognition rate. It motivates other researchers to benefit from both geometric and appearance based information.

2.5.3 Feature Encoding-based Approaches for MFER

Increasing using appearance based approaches leads to explore encoding models. Similar to the facial expression recognition from frontal view, encoding features have advantages in MFER (e.g. decreasing dimension, change spaces for simpler discrimination, etc.).

An efficient encoding approach is the idea based on Canonical Correlation Analysis (CCA). It is by seeking the weighted linear composite (basis vectors) for variables such that the shared correlation between two sets is maximized. CCA is the best way for understanding how the variables in two sets are related to each other? Also it can be used as a measuring the relationship between two multidimensional sets. To find the weighted linear composite (basis vectors) we assume two sets of **x**, **y** where:

$$\mathbf{x} = \langle x_1, x_2, \ldots, x_n \rangle$$
$$\mathbf{y} = \langle y_1, y_2, \ldots, y_n \rangle$$

The goal is finding two sets of **u**, **v** with the maximum correlation:

$$\mathbf{u} = \left\langle w_x^1 x_1, w_x^2 x_2, \ldots, w_x^n x_n \right\rangle = \langle u_1, u_2, \ldots, u_n \rangle$$
$$\mathbf{v} = \left\langle w_y^1 y_1, w_y^2 y_2, \ldots, w_y^n y_n \right\rangle = \langle v_1, v_2, \ldots, v_n \rangle$$

where finding the basis vectors is as:

$$\rho = \frac{E\left[\mathbf{uv}\right]}{\sqrt{E\left[\mathbf{u}^2\right] E\left[\mathbf{v}^2\right]}} \tag{2.1}$$

$$\rho = \frac{E\left[x^T w_x y^T w_y\right]}{\sqrt{E\left[(x^T w_x x^T w_x)\right] E\left[(y^T w_y y^T w_y)\right]}} \tag{2.2}$$

$$\rho = max_{w_x,w_y} \frac{E\left[w_x^T x y^T w_y\right]}{\sqrt{E\left[(w_x^T x x^T w_x)\right] E\left[(w_y^T y y^T w_y)\right]}} \tag{2.3}$$

$$\rho = max_{w_x,w_y} \frac{E\left[w_x^T C_{xy} w_y\right]}{\sqrt{E\left[(w_x^T C_{xx} w_x)\right] E\left[(w_y^T C_{yy} w_y)\right]}} \tag{2.4}$$

Solving with these constraints:

$$\begin{aligned} w_x^T C_{xx} w_x &= 1 \\ w_y^T C_{yy} w_y &= 1 \end{aligned}$$

Therefore:

$$\begin{aligned} C_{xx}^{-1} C_{xy} C_{yy}^{-1} C_{yx} w_x &= \rho^2 w_x \\ C_{yy}^{-1} C_{yx} C_{xx}^{-1} C_{xy} w_y &= \rho^2 w_y \end{aligned}$$

And

$$\begin{aligned} C_{xy} w_y &= \rho \lambda_x C_{xx} w_x \\ C_{yx} w_x &= \rho \lambda_y C_{yy} w_y \end{aligned}$$

Therefore:

$$\lambda_x = \lambda_y^{-1} = \sqrt{\frac{w_y^T C_{yy} w_y}{w_x^T C_{xx} w_x}} \tag{2.5}$$

The computed basis vectors minimize the correlation between two sets **x**, **y**. An extension of CCA for multiple sets (multiset) is employed by [45] where they proposed the idea of discriminative neighbor preserving embedding to obtain the discriminating information from facial expression images. Employed multiset canonical correlation analysis (MCCA) maximizes the correlation of the same facial expressions among the all head poses. Authors proposed facial expression recognition in arbitrary view by maximizing the intrinsic discrimination between same expressions.

Another prominent encoding approach is encoding into the sparse representation. It is not only an efficient approach for compacting the representation of observed data but also it can be useful to explore semantic information. Sparse coding is by choosing codebooks (atoms) to generate a dictionary such a way that reconstructed data from the dictionary has minimum error. Where the maximum number of non-zero coefficients must be less than Γ which is a constant threshold. Sparse coding has more flexibility than Principal Component Analysis (PCA) because it does not impose that the codebooks are orthogonal. Finding a reconstructive dictionary given the training features X by minimizing:

$$\arg\min_{S} = \|X - DS\|_2^2 \quad s.t. \; \|s_i\|_0 \leq \Gamma \tag{2.6}$$

Where D is the dictionary and S the matrix of encoding coefficients. The sparse coding and its advantages is reviewed and discussed in details by the following chapters. Moreover, MFER has been evaluated with sparse coding where [116] used SIFT features and encoded extracted features into the sparse coding by means of a generic dictionary, they created the dictionary from all trained features as an offline processing. Comparison of the experimental results shows that sparse coding is one of the efficient approach for multiview facial expression recognition. An overall structure of encoding approach using sparse coding is illustrated in Figure 2.7.

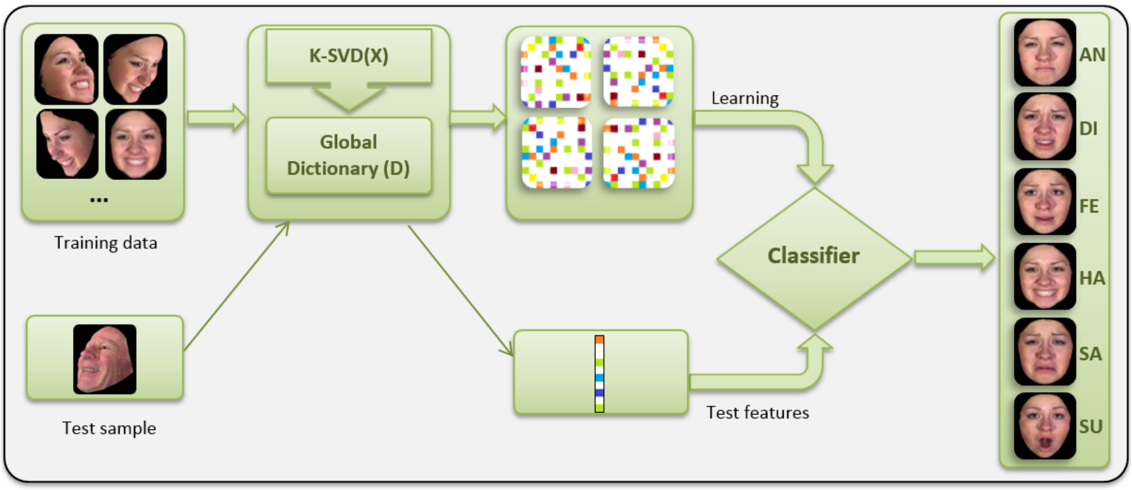


Figure 2.7: Overall structure of encoding approach using generic sparse coding for MFER.

2.5.4 MFER using 3D Face Knowledge

To the best of our knowledge the problem of multiview facial expression recognition using 3D face knowledge is a very new research direction. While the problem of multiview facial expression recognition suffers from unavailable face parts due to the non-frontal head pose, it is a possible direction to perform 3D face model using prior knowledge even from a single face image. However, recently [83] introduced a 3D based reconstruction approach aiming for facial expression recognition, nonetheless, authors used frontal face image to reconstruct non-frontal samples. Clearly it is an unrealistic assumption to have always frontal face. On the other hand, there are attempts on exploring 3D face reconstruction [22, 25] and 3D face recognition [73, 81] that could be very useful for this purpose. Yet, the problem of multiview facial expression recognition from 3D knowledge is much harder because it needs to preserve the expression information. As part of our research, we provide multiview facial expression recognition using 3D face information that is introduced in details by the following chapters.

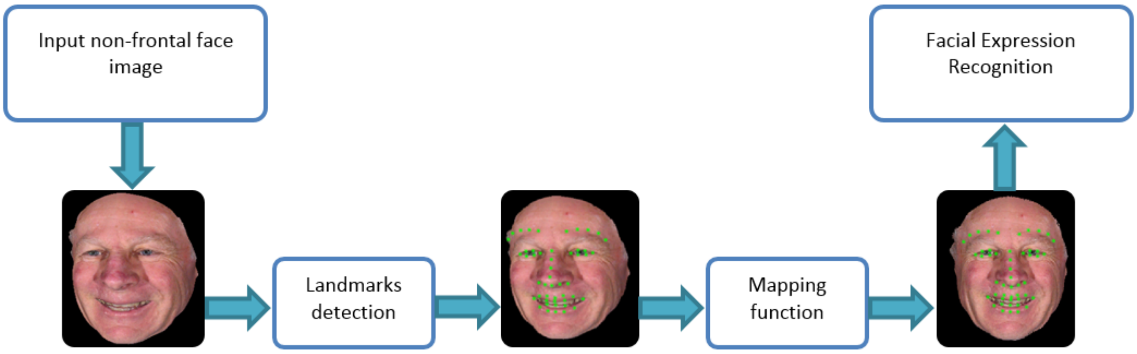


Figure 2.8: Overall structure of mapping approaches using landmarks for MFER.

2.5.5 Mapping Approaches

Several efforts have focused on multiview facial expression recognition using mapping approaches. The earlier attempts, address MFER problem by mapping 2D landmarks points of facial expressions in non-frontal view to the correspondence in the frontal. Most of the works on mapping approaches are based on the linear or non-linear regression. For instance, [94] proposed an idea based on the Gaussian process regression or [95] that used linear regression, support vector regression and relevance vector regression on mapping landmarks from non-frontal view to the frontal. The regression transformation performs the best approximation for the landmarks geometrical mapping.

[96], [93] used similarly regression based model to map previously classified head poses data into the frontal. The former introduced Laplacian dynamic ordinal regression and the later proposed coupled scaled Gaussian process regression model for head pose normalization. They showed that the regression transformation is an efficient idea for mapping the information. Although, all of these efforts focused on geometric information and employed landmarks to estimate frontal geometry from non-frontal face images. The frontalized landmarks are approximations with negligible errors but still efficient in recognition purposes where it has been shown that mapping approaches are state-of-the-art on MFER. Therefore, it is reasonable to extend our methods through the mapping approaches.

The questions are: How perform a frontalized face/features (which is easier for FER) from non-frontal? How we can compensate unavailable facial parts? And such results can improve the current state-of-the-art?

A combination of mapping models and machine learning let us to compensate unavailable facial parts by means of training data where they can learn the mapping functions between non-frontal and frontal facial training data. The same mapping functions can then be used for compensating unavailable facial information during the test. In this way, there are attempts by means of mapping functions such as regression with appearance based information. For instance, [51], [52] and [53] are our approaches that used mapping models based on linear and non-linear regression where they map extracted appearance-based features (e.g. HOG, LBP or sparse representation). Review on the literature shows that mapping model is the most efficient approaches with high performance among the MFER techniques. Figure 2.8 illustrates an overall structure of mapping models based on mapping geometric information (landmarks). We also provide more details on regression-based mapping approaches in chapter 4.

2.6 Decision Forests

Decision forests or random decision forests are an ensemble learning method including multitude of randomly decision trees which considered as weakly learners combined to create a strong learner. It is including several classifiers from subsets of training data and make a combined decision with maximum votes. Random decision forests is a highly efficient structure for many computer vision applications. It has been since used in numerous classification or regression tasks [24, 58, 66, 67, 103, 104].

During the random forests algorithm, each tree of the forests is grown using a bootstrap sample (Φ_i) of training data and the best split is chosen random sample of k variables instead of all variables at each node. A classifier consisting all bootstraps can be present as $h(\mathbf{x}, \Phi_i), i = 1 : N$ where h provides a vote for feature vector $\mathbf{x}$ by means of each independent bootstrap sample Φ_i, $i = 1 : N$. Therefore, a tree $f_t(\mathbf{x}) : \mathcal{X} \longrightarrow \mathcal{Y}$ classifies sample $\mathbf{x} \in \mathcal{X} \subseteq \mathcal{R}^n$ through the depth of trees to the leaves and the final vote is by considering the votes of all trees among the forests.

An example of binary decision tree is demonstrated in Figure 2.9 that input image shows a human face. Splitting function ψ at node j has two possibilities (as it is a binary tree) that are shown as left and right or 0, 1. Answering to the splitting function at each node determines the direction to the left or right, finally, at the leaves a set of questions is followed that decide about a fact (input image is human face or no?). These splitting functions represent a task like weak learners, for example, in Figure 2.9 a splitting function is: ‘Is there a pattern at top like human eyes?’ which is seeking for a mandatory part on the face, clearly if answering to such question is not ‘True’ the input image can not match as a human face. A collection of these questions (splitting functions) make a strong classifier that can efficiently decide about the input image by means of decision tree.

As the decision forests are supervised learning method, there is training stage that learn how trees decide about the input samples. By the learning process we aim to find splitting function parameters θ^* such a way that maximizes the information gain I_j at node j:

$$\theta^* = \underset{\theta}{\operatorname{argmax}}\, I_j \tag{2.7}$$

with

$$I_j = I(X_j, X_j^L, X_j^R, \theta_j) \tag{2.8}$$

where X_j is a set of training data at node j and X_j^L and X_j^R are two subsets of X_j that sent to the left and right branches of node j respectively depend on the parameter of θ_j. The information gain is:

$$G_j = H(X_j) - (\frac{\left|H(X_j^L)\right|}{|H(X_j)|}H(X_j^L) + \frac{\left|H(X_j^R)\right|}{|H(X_j)|}H(X_j^R)) \tag{2.9}$$

where $H(p_i)$ is the entropy as:

$$H(p_i) = \sum_i -p_i \log p_i \tag{2.10}$$

In the particular example of Figure 2.9 a pattern like human eyes is asked at the first depth that whether input training sample has such pattern or no? If yes, the input sample sent to the subset at right otherwise to the left. Therefore, in the second level of the tree we have two subsets with a simple knowledge about availability of eyes-like pattern on the input image. In other word, X_1 which is the training data at root is divided into two subset of X_1^L and X_1^R that X_1^R contains samples with eyes-like pattern. During the test, an input sample answer to the same questions trough the depth of the tree and the leaf performs a distribution over the class C in case of classification tasks. In the following we have reviewed random forests and its extension of regression forests.

2.6.1 Random Forests

Random forest is an ensemble of randomized trees, each tree is built, trained and tested independently from other trees. The training data for each tree generated from subsampling of the original data. During the training, each node splits the training data into the subsets using splitting functions:

$$\psi(\mathbf{x}, \theta) = \begin{cases} 0 & \text{if } r_\theta(x) < 0 \\ 1 & \text{otherwise} \end{cases} \tag{2.11}$$

Where θ defines the response function $r_\theta(\mathbf{x})$. There are many different kind of response function $r(.)$ which has been used in different tasks [66, 110]. For instance, $r_\theta(\mathbf{x}) = \mathbf{x}[\theta_1] - \theta_{th}$ used in [103]. They defined operator [.] for selecting one dimension of x such that $\theta \in [1..len(\mathbf{x})]$ and θ_{th} as an arbitrary threshold.

Moreover, injecting the randomness during learning is an important stage for random forests. There are two common approaches, the first which is the most popular relies on randomly sampling the training data (i.e. bagging) and the second method is optimizing the randomized nodes [17, 109]. Figure 2.10 demonstrates the randomness process in random forests.

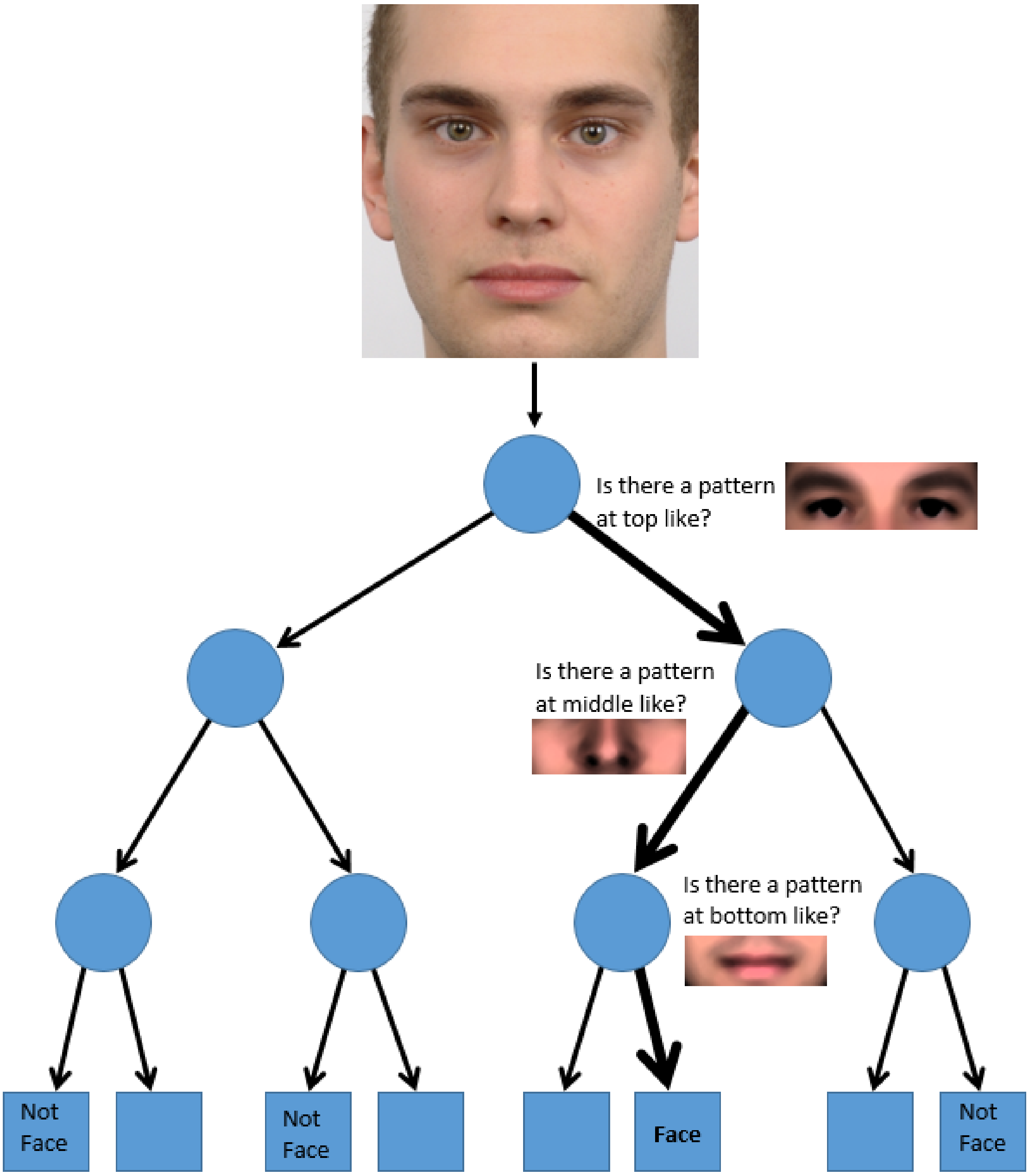


Figure 2.9: Illustration of a decision tree which is used to figure out whether input image is a human face or no?

Therefore, each tree of the forests is as $f_t(\mathbf{x}) : \mathcal{X} \longrightarrow \mathcal{Y}$ and for the forest $F = \{f_1, \ldots, f_T\}$ where T is the number of trees. The probability of class example k in the case of classification task is:

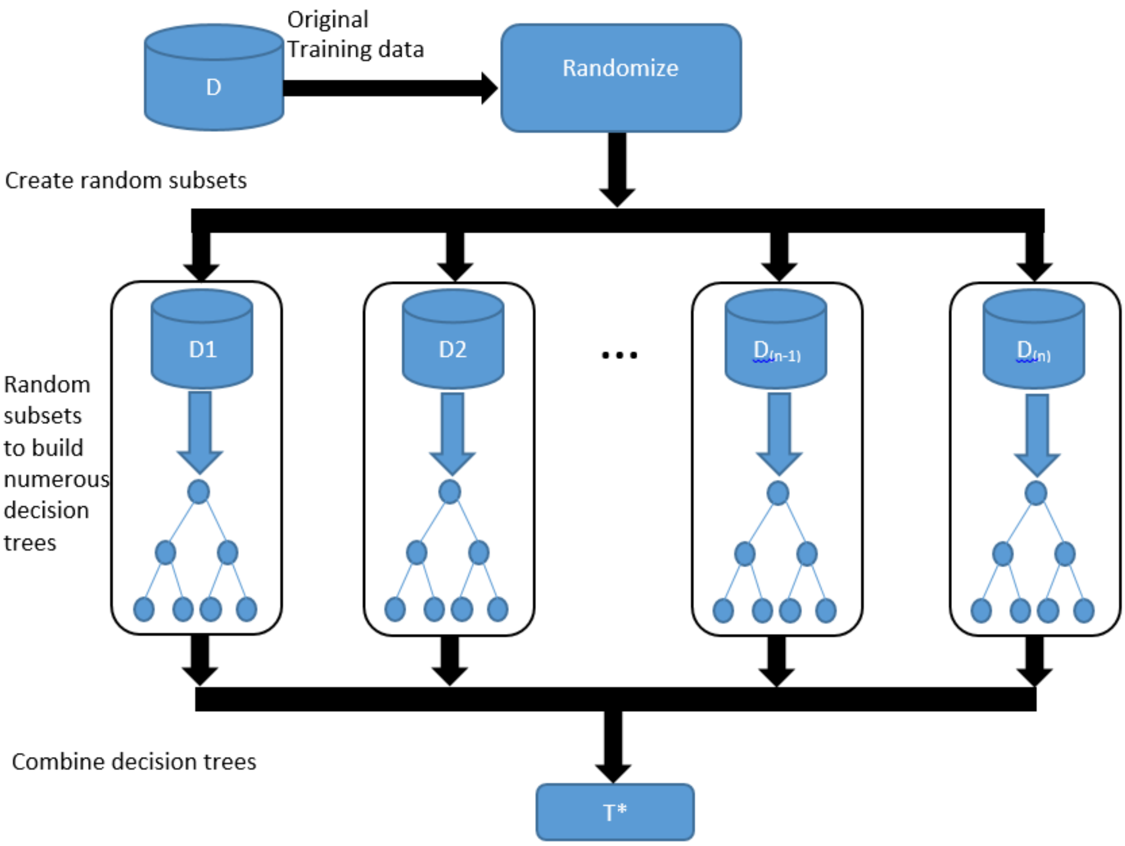


Figure 2.10: Randomness stage in random forests.

$$p(k|x) = \frac{1}{T}\sum_{t=1}^{T} p_t(k|x) \tag{2.12}$$

$$C(x) = \arg\min_{k\in\mathcal{Y}} p(k|x) \tag{2.13}$$

Where $p_t(k|x)$ is density of class label k estimated by t^{th} tree and $C(x)$ the final class label.

2.6.2 Regression Forests

Regression forest is a set of randomly trained regression trees where a regression tree splits a complex non-linear regression problem into several smaller problems which are more easily to solve [17]. Regression forest has been used widely for non-linear regression of a pair of explanatory and dependent variables. The main difference between random and regression forests is the continues nature of regression forest which is able to work with continues data, therefore, training and test labels are continuous. In regression forest, we can use different kind of objective functions (e.g. linear, polynomial, probabilistic, etc.)

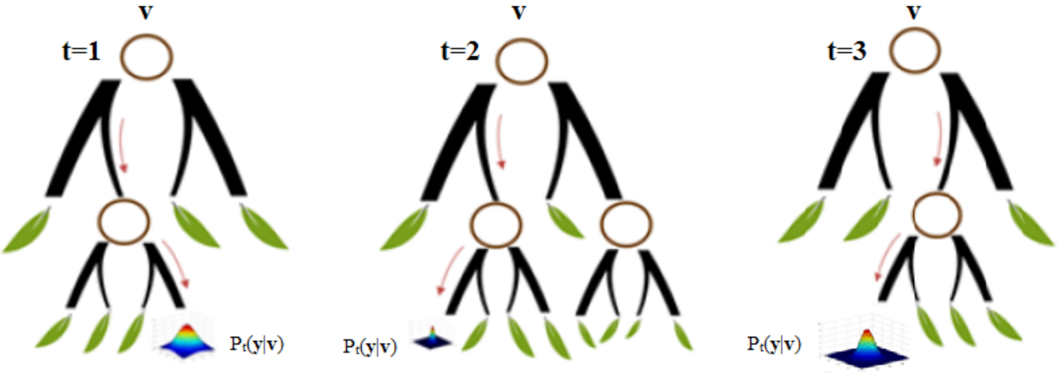


Figure 2.11: The overall structure of regression forests.

for a subspace of input data. For instance, a general polynomial function for input data x is:

$$y(x) = \sum_{i=0}^{n} w_i x^i \tag{2.14}$$

We can perform an optimize value n using Taylor series, nevertheless, equation 2.14 is a general form of polynomial solution which could be a linear solution while n=1. Again similar to the classification the regression forest output is the average of all T trees:

$$p(y|v) = \frac{1}{T}\sum_{t=1}^{T} p_t(y|v) \tag{2.15}$$

Where y is continuous input data and $p_t(y|v)$ is density probability estimation function over an input point v. Consequently, we define the mapping function f similar to w_i in equation 2.14 that depends on the data X_{NF} :

$$\hat{X}_{Fr} = f(X_{NF})X_{NF} \tag{2.16}$$

Where the notation X_{NF} denoted to a set of non-frontal facial features and X_{Fr} to correspondence frontal features. Note that throughout the rest of the chapter, we use the same notations. Moreover, $x\{i\}$ in both X_{NF} and X_{Fr} is a feature vector of i^{th} same subject but in different viewpoint. With respect to equation 2.16, we need to find the function $f(.)$ using training data and then extend it for test samples. In the following, we explain how our MF approach can efficiently provide the mapping function $f(.)$ using decision forests.

2.7 Frontalization

Very recently publication [145] has introduced mapping approach for frontalizing facial images from non-frontal view similar to our approaches. It employed 3D face model and works based on the idea of 3D morphable model which introduced first by Volker Blanz [9]. In the following we review this approach.

2.7.1 3D Morphable Model

3D Morphable Model is one of the most successful approach for pose invariant 3D facial analysis that was first introduced by Blanz and Vetter [9]. The morphable face model is based on a linear combination of 3D scan faces which contain both shape and texture that describe a realistic human face:

$$S = \begin{pmatrix} x_1 & x_2 & \dots & x_n \\ y_1 & y_2 & \dots & y_n \\ z_1 & z_2 & \dots & z_n \end{pmatrix}, \quad T = \begin{pmatrix} r_1 & r_2 & \dots & r_n \\ g_1 & g_2 & \dots & g_n \\ b_1 & b_2 & \dots & b_n \end{pmatrix} \tag{2.17}$$

That we can make a combinations of linear shape S_i and texture T_i to produce a new face sample:

$$S = \sum_{i=1}^{m} a_i S_i, \quad T = \sum_{i=1}^{m} b_i T_i. \tag{2.18}$$

whilst:

$$\sum_{i=1}^{m} a_i = 1, \quad \sum_{i=1}^{m} b_i = 1. \tag{2.19}$$

Where S_i and T_i are shape and texture vectors respectively and sum of the coefficients a_i and b_i must be equal to one, aiming to avoid changes in overall brightness in texture and size in shape. The morphable model can be described using principle component analysis (PCA) that performs basis representation on orthogonal coordinate system to decorrelate texture and shape vectors:

$$S_{model} = \overline{S} + \sum_{i=1}^{m-1} \alpha_i s_i, \quad T_{model} = \overline{T} + \sum_{i=1}^{m-1} \beta_i t_i. \tag{2.20}$$

Where m is the number of faces, $\overline{S}$ and $\overline{T}$ are mean of shape and texture respectively S_{model} and T_{model} are shape and texture models and s_i, t_i are eigenvectors of covariance matrix shape and texture. $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_{m-1})$ and $(\beta = \beta_1, \beta_2, \dots, \beta_{m-1})$ are shape and texture coefficients that determine the importance of each specific shape or texture.

For instance, the application of PCA on shape is as the following: We first compute and subtract averaged shape vectors $\overline{S}$ from all shape vectors and represent data in matrix A

then make the covariance matrix C from A and compute the singular value decomposition (SVD) [121] to perform the eigenvectors:

$$\overline{S} = \frac{1}{M} \sum_{i=1}^{M} S_i, \tag{2.21}$$

$$a_i = S_i - \overline{S}, \tag{2.22}$$

$$A = [a_1, a_2, \ldots, a_M] = U \Sigma V, \tag{2.23}$$

$$C = \frac{1}{M} A A^T = \frac{1}{M} U \Sigma^2 V^T, \tag{2.24}$$

That there are M column-based orthogonal eigenvectors in matrix U and eigenvalues are $\sigma_i^2 = \lambda_i^2 / M$ where the λ_i are deceasing diagonal elements of matrix Σ. There is also another advantage for representing shape and texture using PCA that is dimensionality reduction [122] which is by using k eigenvectors of matrix U as $X \approx [u_1, u_2, \ldots, u_k]$ instead of N dimensions of input data $X = [x_1, x_2, \ldots, x_N]$ while $k << N$.

For a new input sample, 3DMM estimates the parameters of mapping such a way that a weak projection of the model maximally match with the input test sample by:

$$I_{fitted} = f(\overline{S} + \sum_{i=1}^{m-1} \alpha_i s_i), \tag{2.25}$$

Where I_{fitted} is the fitted 3D face model to the input image by means of $f = (\gamma, r, \tau)$ which is mapping function including: scale (γ), rotation (r) and 3D translation (τ). After mapping 3D face model over the input face, a 2D image is generated using weak perspective projection. The generated image and input sample will be used to find the 3D parameters by minimizing the Euclidean distance between them that is done using Gauss-Newton method over the pixels intensity:

$$E = \sum_{i,j} \| I_{inp}(i,j) - I_{gen}(i,j) \|^2 \tag{2.26}$$

3DMM determines the parameters α_i and β_i in equation 2.20 by minimizing the energy function E in equation 2.26 that provide appropriate reconstruction of input sample in 3D face structure.

3D morphable model which is based on the synthesizing of the facial images requires to solve two problems simultaneously [138]:

- The model must be able to synthesize all face images for covering the possible variations.

- Solving an optimization problem due to the fitting the model to a novel face image.

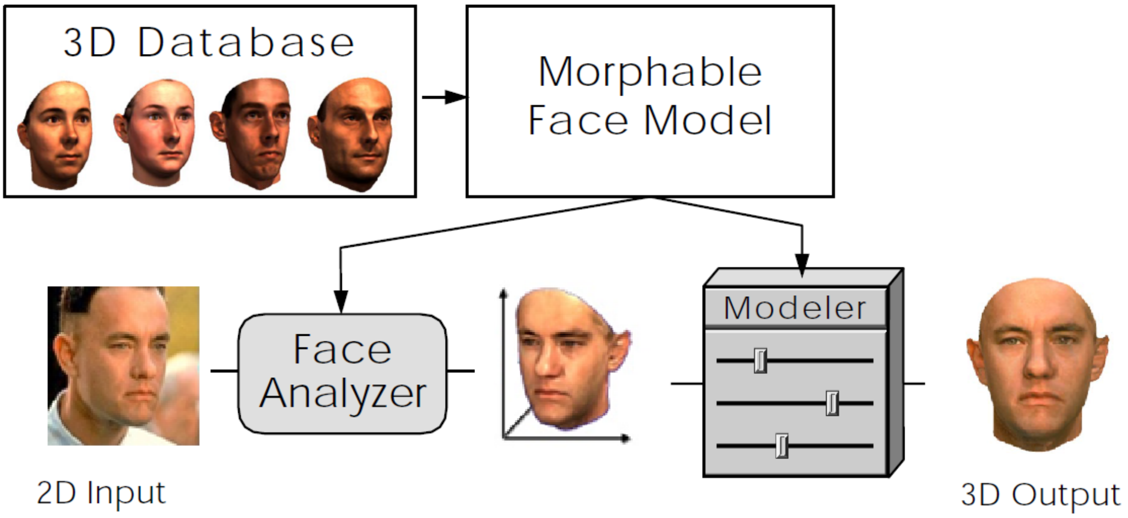


Figure 2.12: 3D morphable model, overall structure.

Nevertheless, as 3DMM uses the pixels intensity during the algorithm for fitting the parameters, it is seriously sensitive on expressions, occlusion and complicated illumination. Moreover, it suffers from local minimum problem due to the using Gauss-Newton optimization, therefore it is an expensive process in terms of time complexity. These side-effects motivate numerous methods to explore new approaches to address these problems by exploiting landmarks, edges, pixels, etc. [145, 146]. In general, the most important challenge of the 3D morphable model is discovering the correspondences between 3D and 2D points.

2.7.2 Frontalization using 3D Face Information

In contrast to the 3D morphable model, there is a frontalization approach [37] that synthesizes frontal facing views to provide a frontal face from non-frontal view. Unlike to the 3DMM that attempted to approximate the 3D facial shapes, frontalization approach transforms faces viewed from unconstrained viewpoints to constrained froward facing poses. Frontalization uses a single 3D surface as an approximation to the shape of all input faces. An important achievement by frontalization approach is its capability of working on unconstrained images while the recognition rate on such samples (In the Wild [44]) is significantly improved in compare with the related works [43, 92].

Early frontalization [36] approximates a 3D surface of the face to generate a new view. It uses accurate localization of facial feature points for alignment. However, it does not guarantee for such accurate alignment. Very recently approach of frontalization uses instead a single 3D reference face to produce the frontal view.

The overall pipeline for the frontalization is shown in Figure 2.13 which is as the following: First, it detects the face using the popular face detectors [124] as the input

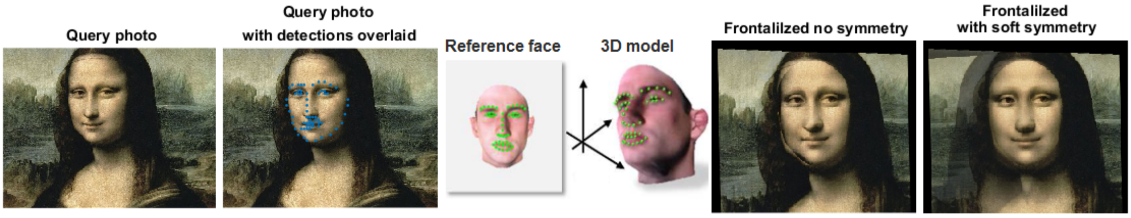


Figure 2.13: Frontalization overview, a 3D based method to provide frontal face from non-frontal.

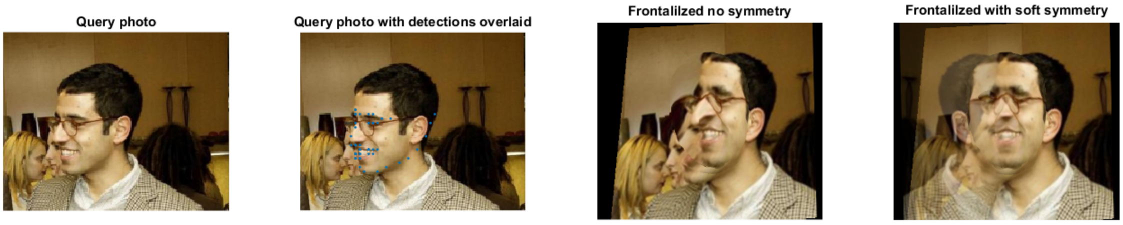

Query photo
Query photo with detections overlaid
Frontalilzed no symmetry
Frontalilzed with soft symmetry

Figure 2.14: Frontalization fails, the frontalization on the faces which are not near to frontal almost fail.

image is unconstrained, then, the face is aligned, cropped and rescaled into the standard coordinate system. By the next step, facial landmarks is recognized using the state-of-the-art [59, 144] and aligned with the textured 3D model of a generic reference face. The frontalized face can be obtained by back projecting from input image to the reference coordinate system using 3D surface model. The final frontal face is generated by inpainting the unavailable facial parts due to the head pose using the symmetric information. During the frontalization, it needs to compute the 2D-3D correspondence between input face points and the correspondence points on 3D surface that is known as *Posit*. It is as:

$$M_P = C_P[R_P t_P] \tag{2.27}$$

Where M_P is the reference mapping matrix, C_P is intrinsic matrix and R_P, t_P are rotation matrix and translation vector respectively. Frontalization benefits from the mapping matrix M_P to provide a correspondences between two points of p_i, P_i where p_i is a landmark on 2D input face and P_i is the correspondence of p_i on the 3D surface that $(p_i, P_i) = (x_i, y_i, X_i, Y_i, Z_i)$. Frontalization directly matches the landmark points from non-frontal input sample to the landmarks in frontal rendered 3D surface and find appropriate transformation, it uses then this transformation during the back projection.

Although the results of frontalization seems as well as the ground truth but there are two major problems with this approach: First, frontalization works well with the samples near to the frontal but it fails with the head pose of very non-frontal faces. Second, there is no solution for asymmetric facial parts. As it uses the symmetric information to compensate unavailable parts due to the head pose, if invisible parts contain asymmetric region (e.g. mouth, forehead), there is no way to compensate them. The first challenge is shown in Figure 2.14 and the second problem can be seen with the results of the authors in [37]. Nevertheless, there are also other small problems that could be ignored with the current solutions (e.g. unusual objects in symmetric, lighting, etc.). A few of these problems are addressed by the authors with the same work.

2.8 Intrinsic Image Decomposition (IID)

The intrinsic image decomposition (IID) is the problem of separating image into the intrinsic components that make easier and more efficient several computer vision tasks (e.g. face analysis). The most common approaches [14, 71, 108] decompose images into two *reflectance* (albedo) and *shading* components that they have advantages in compare with the *rgb* space. The most important advantage is that separated components can be used more efficiently on image manipulation. For example, learning symmetric information from reflectance is more reasonable than shading.

Although, IID capable to operate image editing such as re-texturing and relighting, it is efficient for inpainting images by inpainting each component separately. There are several kind of IID approaches: 1) Methods that use image gradients to either reflectance or shading changes, this method is employed in several works by thresholding image gradients [64]. To preserve the global consistency, 2) There are methods that use non-local constrained between non-adjacent pixels [107, 137]. Also, 3) Methods that benefit from generic or specific priors including illumination environment or face priors [5, 71].

We will use IID in our inpainting approach in chapter 4 for decomposing *rgb* images into the reflectance (albedo) and shading components. Therefore, by assuming two input images I_1 and I_2 for decomposing into their intrinsic components as:

$$\begin{aligned} I_1 &= R_{I_1} \cdot S_{I_1}, && (2.28) \\ I_2 &= R_{I_2} \cdot S_{I_2}, && (2.29) \end{aligned}$$

where R_{I_1}, S_{I_1}, R_{I_2}, S_{I_2} are the reflectance and shading images of the input image I_1 and I_2, respectively. Based on the decomposition results, we match the skin tone and shading level between the input image 1 and 2 by:

$$\begin{aligned} R'_{I_2}(c) &= \frac{M(R_{I_1})}{M(R_{I_2})} R_{I_2},\ c \in \{r, g, b\} && (2.30) \\ S'_{I_2} &= \frac{M(S_{I_1})}{M(S_{I_2})} S_{I_2}, && (2.31) \end{aligned}$$

where $M(\cdot)$ is the median operator.

2.9 Discussion

Literature review on the problem of facial expression recognition shows that after 37 years from beginning of this problem, there are multiple approaches that proposed valuable results. Accuracy of the current FER methods is now more than 96% on the popular datasets (e.g. CK+) with high performance which means that the problem of FER is almost solved. Nevertheless, there are still questions without clear answers, for instance:

On Methodology:

1. - Which methodology is the best (geometric based, appearance based, hybrid, multimodal, etc.)

2. - The capabilities of the methods (real time, robustness on challenges e.g. noise, occlusions, etc.)

On Classifier:

1. - Which classifier is the best while it is also depend on our training data

2. - The feasibility of the classifiers based on problem statement (speed, reliability, etc.)

On Features:

1. - Geometric or appearance-based features? or fusion of them?

2. - Method of data fusion if use both geometric and appearance features

3. - Features properties (e.g. size, use dimension reduction or no?)

On Data:

1. - There are several datasets, which one is the best?

2. - How much is a dataset close to the real world data?

3. - What is the best settings for the data?

On validation:

1. - Which validation is the best for facial expression recognition?

2. - How much the validations are far from the truth?

Although, One or some of these questions have been responded by the current approaches but there is not unique answer for these questions. On the other hand, new challenges on facial expression recognition such as MFER is introduced.

Multiview facial expression recognition (MFER) has been analyzed in last years and there are a few approaches that addressed this problem, nevertheless, the overall performance and the accuracy is far from the current facial expression recognition (FER) performance. Because the most serious challenge among the multiview facial expression recognition is that some of the facial data which are necessary for the expression recognition are not available due to the head pose. Therefore, a meaningful way is providing an approximation from unavailable facial parts. To this end an idea is employing regression transformation with geometric or appearance-based features.

During this thesis we address these important questions: How perform a frontalized face/features from non-frontal? What is the best solution for compensating unavailable facial features? Does the frontalized features better than the state-of-the-art on expression recognition?

3

Our System Pipeline

Contents

In this chapter, we first propose our MFER system pipeline and then introduce its components. The goal of MFER is to decide which facial expression is happening while the input face image must not be needed to be a frontal face image. As the current FER systems have appropriate performance on the frontal facial expression recognition, we employ a simple yet efficient idea for frontal FER and focus on frontalizing non-frontal face/features. We assume that the current FER performance can be achievable if we perform desirable mapping from non-frontal data into the frontal. The frontalization of the faces/features can be either on geometric or appearance based information. Nevertheless, our system pipeline is including preparation, features extraction, feature analysis, mapping transformation, and facial expression recognition.

In this chapter we introduce and discuss on the whole of the system except the mapping transformation. We explain about the best features, employed dimension reduction, classification and other details regarding to the MFER system in the following of this chapter. The mapping ideas to handle non-frontal facial images are proposed independently in the chapter 4.

3.1 Our System Pipeline

Before introducing our system pipeline it is necessary to clarify that our MFER system is based on a supervised learning approach which means that we have both training and test processes. To learn our system we employed training data and their correspondence expressions, we estimate then the best expression label during the test based on the similarity of the test sample with the training data. The training data and classifier has important tasks during the supervised methods. The overall structure of an automatic facial expression recognition (FER) illustrated in Figure 2.2. Assuming that we need to provide frontal view, we need an extra pre-processing step for transforming non-frontal faces into the frontal view to perform a general idea for similar system using non-frontal samples. Therefore, we add an extra step of '*mapping process*'. To this end, we need also estimating the head pose which is necessary for our mapping process. We propose desirable mappings from variant viewpoints to the frontal view based on the estimated head pose. Figure 3.1 illustrates general structure of our MFER system. This is an efficient solution for the problem of MFER where it first extracts the facial features and decide about the head pose using a supervised learning algorithm. By discovering the head pose it learns then pose specific mappings for each viewpoints. The mappings can be useful then for an unseen test sample. In general there are two important reasons that made this system efficient: the first is that the current idea is enough fast and efficient in terms of time complexity due to the splitting the problem into the several smaller problems. The second which is again due to simplification of the problem is the performance of the system that is better than relevant methods. As we proposed several ideas for mapping process, they are explained and discussed in chapter 4, but in the following we describe our system components.

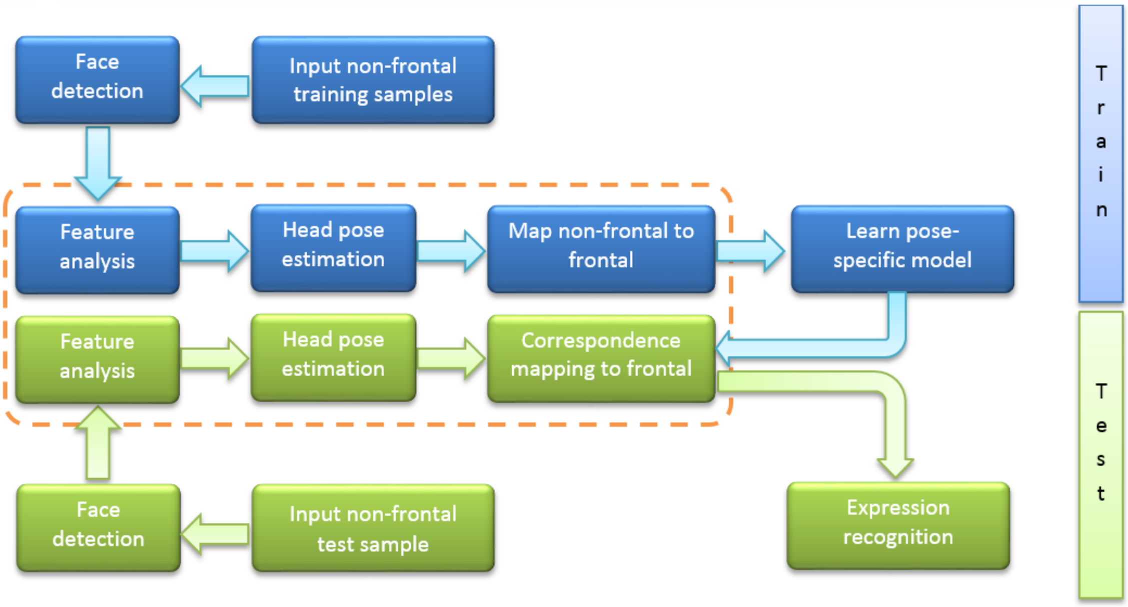
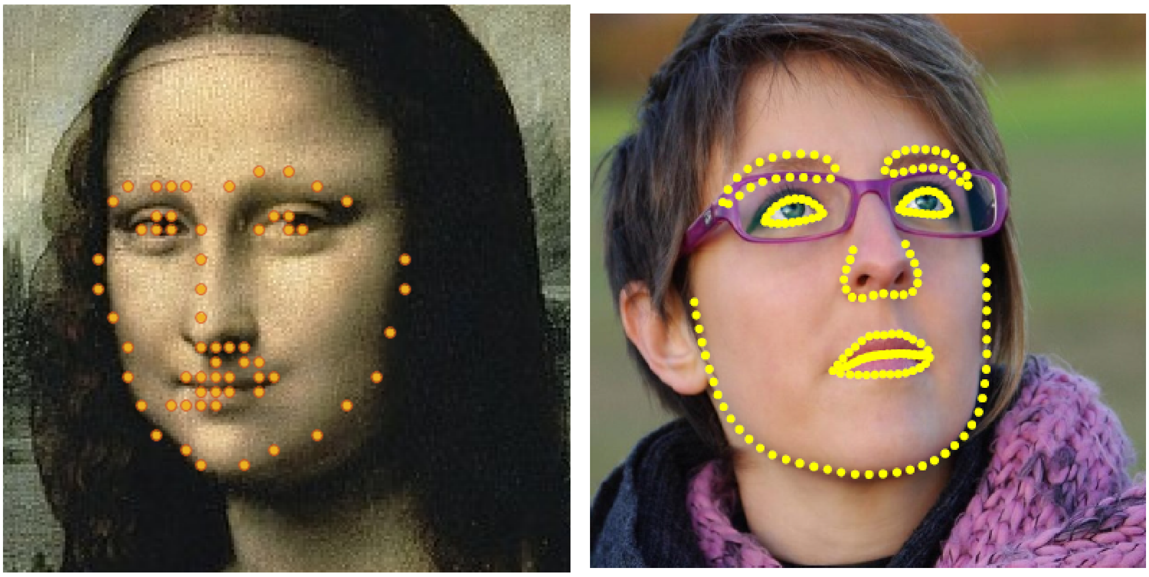

Face detection
Input non-frontal training samples
Feature analysis
Head pose estimation
Map non-frontal to frontal
Learn pose-specific model
Feature analysis
Head pose estimation
Correspondence mapping to frontal
Face detection
Input non-frontal test sample
Expression recognition
Train
Test

Figure 3.1: General structure of the proposed MFER system.

Our system pipeline includes some secondary components that are individually specific problems in computer vision. For instance, *Face detection* which is itself a common problem in computer vision and reviewed on previous chapter. Other components are feature extraction, feature dimension reduction, feature encoding, head pose classification



(a) By Zhu and Ramanan

(b) By Kazemi and Sullivan

Figure 3.2: Response of employed face landmarks detector (even from non-frontal samples).

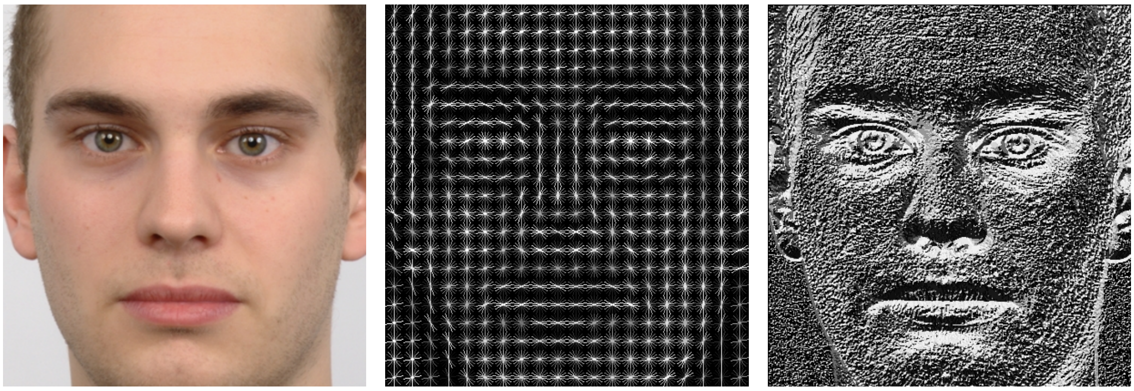
and classifier that described in the following sections.

3.2 Our MFER System Components

Performing an efficient MFER system needs to provide first a framework that is able to recognize facial expression from frontal or near to frontal fruitfully. The literature review on automatic facial expression recognition systems, however cannot determine the best system clearly but we can approximate the best components for a successful system. To this end, our system includes: 1) face detection, 2) feature analysis which is itself consist of triple feature extraction, feature dimension reduction and feature encoding, 3) head pose estimation, 4) facial mapping and 5) classification. Facial mapping is postponed to the next chapter and other components are described in the following.

3.2.1 Face Detection

Our system pipeline includes face detection which is itself an important and almost solved problem in computer vision. As reviewed in chapter 2, the Viola and Jones's method is an efficient technique for the classic face detection problem. We also mentioned that there are recently introduced efficient approaches for face detection even from non-frontal face images. We employed both [59] and [144] alternatively for our system. They perform a region of interest (ROI) including the face that we want to re-generate its frontal. Example of face detection by means of these two methods is shows in Figure 3.2.



(a) Original image **(b)** HOG **(c)** LBP

Figure 3.3: The responses of HOG and LBP feature descriptors on a test image.

3.2.2 Feature Extraction

Feature extraction has a key rule in most of the computer vision applications. As can be seen by the literature review in chapter 2 there are several feature descriptors either for general purpose or for specific facial expression target. On the other hand, we found that both geometric and appearance based methods are useful during feature analysis. Therefore, a concatenation of two feature descriptors of HOG [19] and LBP [89] has been employed.

Histogram of oriented gradients (HOG) is a gradient-based descriptor which is stable on illumination variation. It can describe local object appearance and shape by the distribution of edge directions or intensity gradients, this is by counting the occurrences of gradient orientation in localized portions of an image. Since HOG operates on local cells, it is invariant to geometric and photometric transformation but not to orientation. Moreover, it is a fast descriptor in comparison to the SIFT and LDP (Local Directional Pattern) descriptors due to the simple computation.

On the other hand, Local binary patterns (LBP) is a common texture-based descriptor which is used widely in face analysis. It is a powerful feature descriptor for texture analysis which has been shown that a combination of LBP and HOG improves the detection performance considerably [127]. LBP considers 8 bit to describe each pixel where it evaluates 8 neighbors around of each pixel and set '1' if their values are grater than the center pixel, otherwise set '0'. It computes then the histogram over the cells; also extended versions of LBP considers the normalization of histogram. As success of LBP to describe appearance-based features, there are several descriptors that is originally driven from LBP basic idea.

In our experiments, the extracted features are defined as feature vectors for every facial image and benefit from both HOG and LBP by concatenating them into a combined feature vector. The responses of HOG and LBP is visualized in Figure 3.3.

Moreover, to analysis the impact of the feature descriptors we evaluated HOG, LBP and both of them together on the same conditions (e.g. similar dataset, algorithm, etc.). The aim of this evaluation was to discover the best information (e.g. texture or gradient) for frontal FER. These analysis is illustrated in Table 3.1, Table 3.2 and Table 3.3 which are the results of basic FER on frontal faces. These evaluations show that the concatenating of the both descriptors has the best performance due to the employing both texture and gradient information. We also find the best parameters for such descriptors (e.g. descriptors cell size).

Cell size	**Dimension**	**FER accuracy**
15	6045	75.0 %
20	3410	75.0 %
25	2232	76.6 %
30	1519	72.5 %

Table 3.1: Evaluation of HOG features cell size and its facial expression results on frontal faces.

Cell size	**Dimension**	**FER accuracy**
15	10556	82.5 %
20	6380	83.3 %
25	3248	83.3 %
30	2436	79.1 %

Table 3.2: Evaluation of LBP features cell size and its facial expression results on frontal faces.

Cell size	**Dimension**	**FER accuracy**
15	16601	86.6 %
20	9790	87.0 %
25	5480	89.2 %
30	3955	85.0 %

Table 3.3: Evaluation of different cell size on concatenation of HOG and LBP features and its facial expression result for frontal faces.

3.2.3 Features Dimension Reduction

In many problems, the measured data like extracted features in our work are expensive in terms of the dimensionality. Although, extracted features are always useful but most of the computer vision applications suffer from the high dimensionality of the feature vectors. Consequently, a combination of feature descriptors makes it much more complicated.

Therefore, dimension reduction is an appropriate solution to decrease not only the feature dimensions but also aid to reduce probable outliers. Principle Component Analysis (PCA) and its extensions (e.g. kernel PCA or dual PCA) [39] are well known techniques that produce compact encoded data from high dimensional feature vectors. It is by providing a sequence of principle components that perform the best linear approximation of high dimensional data where the variance of the data in the low-dimensional representation is maximized [55]. It can be represented by finding a transformation as:

$$T = XW \tag{3.1}$$

Where X, W and T are matrices of input vectors, basis vectors and transformations respectively that maps feature vector x_i from an original space of p variables to a new space of q variables which are uncorrelated over the dataset [7]. Nevertheless, all of the principle components must not be needed for approximating of the input data X, therefore, selecting l components can preserve the information that we need in our work to represent the data, thus truncated transformation is:

$$T_l = XW_l \tag{3.2}$$

PCA learns a linear transformation of l orthogonal bases of matrix W and total squared reconstruction error will be:

$$E = \left\| TW^T - T_l W_l^T \right\|_2^2 \tag{3.3}$$

In some parts of our work, we employed PCA for reducing the extracted features as it is a popular and successful linear method for dimension reduction. However, a non-linear PCA can be achievable by means of kernel trick. PCA compacts our combined features from 5480 dimensions (2232 dimensions are computed by HOG and 3248 dimensions by LBP) to 200 dimensions. Reduced features are used then for mapping ideas which are explained in chapter 4. By the following, we describe other method for dimension reduction that has also other benefits.

3.2.4 Features Encoding

Feature encoding makes efficient and easier analysis in most of the time by changing the domain space to another space. Sparse Coding (SC) is a recently popular and successful encoding approach that either decreases the dimensionality of the data or reduces outliers. Therefore, we employed SC instead of PCA in our last works due to the possibility of simplification of data for machine learning purposes and ignoring outliers. Sparse representation approximates an input vector by a sparse linear combination of basis (codebooks or atoms) based on a compact dictionary D. It aims to find a set of basis vectors S_i such a way that each input vector $\mathbf{x}$ can be represented with a linear combination of these basis vectors.

$$x = \sum_{i=1}^{k} w_i S_i \tag{3.4}$$

The advantage of sparse coding in comparison with PCA is that while PCA learns a complete set of basis vectors, sparse coding wishes to learn an over-complete set of basis vectors to represent input vector $\mathbf{x}$. The other advantage is that sparse coding has more flexibility than PCA and it does not impose that the basis (codebooks) are orthogonal. Although, over-complete basis lead to better capability of capturing patterns, but the coefficients w_i are no longer uniquely determined by input vector $\mathbf{x}$ [125].

A set of basis vectors represent a dictionary (D) where each feature vector can be reconstructed using this dictionary. Creating such dictionary (set of basis vectors) that minimizes the error rate between reconstructed vectors and ground truth, can be done by means of K-SVD [1] which is a learning algorithm for this purpose. Therefore, we are interested in finding a reconstructive dictionary given the training features X by minimizing:

$$\|X - DS\|_2^2 \quad s.t. \, \|s_i\|_0 \leq \Gamma \tag{3.5}$$

Where $D \in \mathrm{I\!R}^{(n \times k)}$ is the dictionary, each column representing a codebook vector while it is contain k bases (atoms, codebooks) and $S \in \mathrm{I\!R}^{(k \times N)}$ the matrix of encoding coefficients. Γ is the sparsity constraint factor, defining the maximum number of non-zero coefficients per sample. Given a fixed dictionary (D), K-SVD [1] and orthogonal matching pursuit (OMP) [120] can represent (solving) the coding of a new input sample. K-SVD benefits from singular value decomposition for the dictionary for sparse representation where it is a generalization of k-means clustering. Moreover, OMP is originally an iterative algorithm and very expensive in terms of time complexity that is an extended version of matching pursuit (MP) which is a greedy solution. OMP has better result than MP but computationally more expensive, therefore, improving alternative solutions that can improve the running cost of OMP is always desirable. Our literature review on previous research [132] shows that the sparsity constraint must not be needed during the reconstruction. Therefore, given a sparse codebook create by Eq. 3.5, we can reformulate the solution of finding best encoding S by replacing $l0-norm$ for the coefficients, which could be called Ridge Regression [119] as:

$$\|X_i - DS\|_2^2 + \lambda \, \|S\|_2 \tag{3.6}$$

Where D is created global dictionary and X_i is a set of input data. However, it eliminates the rules leading to sparsity but we are using $l2-norm$ because of two clear reasons: first, to avoid over fitting during the regression; second, to stabilize projections specially when we know there are collinearity between data. Moreover, as mentioned before, the sparsity constraint must not be needed during the reconstruction. The parameter λ also allows us to detract the singularity problem. Eq. 3.6 is a ridge regression model and the solution is given by least square solution that is more described in chapter 4.

3.2.5 Machine Learning

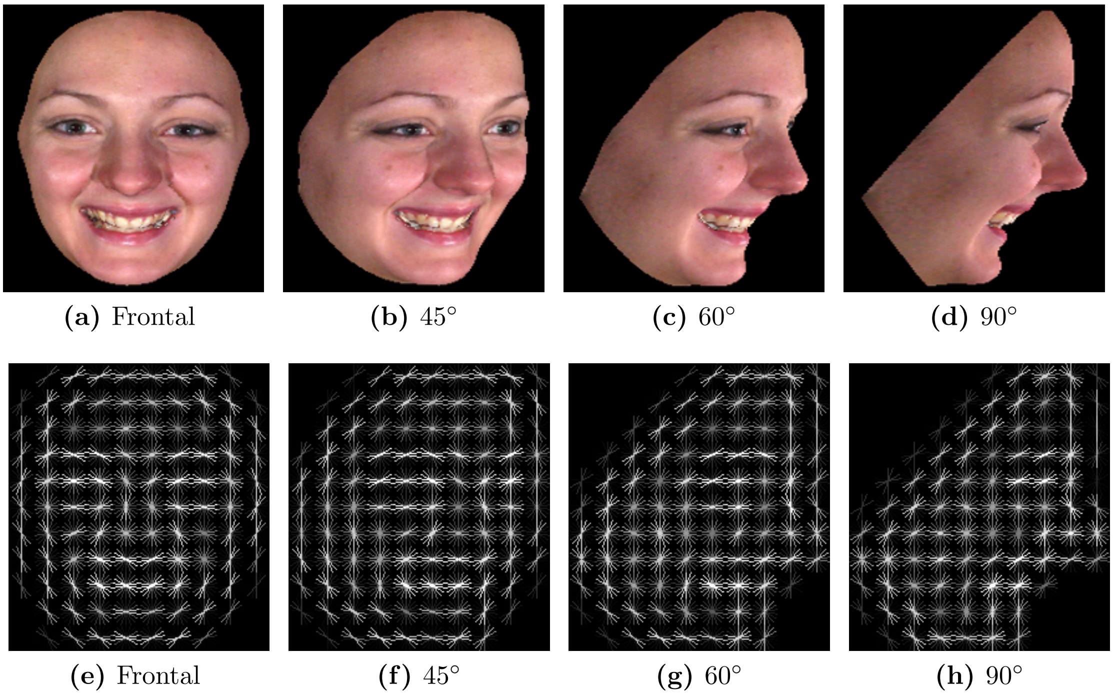
Machine learning is a branch of computer science and artificial intelligence. It is a "Field of study that gives computers the ability to learn without being explicitly programmed", (said Arthur Samuel, 1959) that can be done by learning from data and make a decision for a new unseen sample. The major task of machine learning is exploring algorithms to provide such learning and prediction. Therefore, learning algorithms prepare trained model f that deduces from training observations to an unknown sample. In general, machine learning tasks can be classified into three typical categories based on the learning system. They are **supervised**, **unsupervised** and **reinforcement learning** approaches. Moreover, a popular subcategory is **semi-supervised learning** which is between supervised and unsupervised learning approaches. A supervised learning approach is a model that learns a general rule from input samples and their desired output labels. While $X = \{(x_1, y_1), (x_2, y_2), \dots (x_N, y_N)\}$ is a set of training data, supervised learning finds a function f where $f : X \longrightarrow Y$. In contrast, unsupervised learning approaches are algorithms that learn from training data $X = \{(x_1), (x_2), \dots (x_N)\}$ but no given labels, it is by exploring hidden patterns from training data. The third extreme case of learning approaches is reinforcement learning which exploits teaching feedback from environment that weather the predicted class label is correct or no aiming to improve the performance. The most important feature of reinforcement learning is its interacting with an environment that can make a measurement for evaluating the predictions validity.

In addition, semi-supervised method is originally a kind of supervised method that uses also from unlabeled data for training. Numerous works with semi-supervised approach [70] and [100] showed that unlabeled data when used in conjunction with a small number of labeled data, can produce appropriate improvements on learning accuracy.

Our work relies on exploiting supervised approach whilst we also evaluated an unsupervised method for head pose estimation which is explained in the following.

3.2.6 Head Pose Estimation

As part of our pipeline system, we perform to categorize input faces based on the head pose. We will discuss on mapping approaches from categorized input face into the frontal by the next chapter. Nevertheless, the problem of MFER has two major ambiguities (except from ethnicity, personality, skin tone, etc. which are same in MFER): First, facial expression and second, multiview (variant head pose) face. We are looking to address the first by reducing the ambiguity of the second variation. This means, it is always desirable if we can recognize the head pose efficiently. Therefore, categorizing data into the several smaller groups has advantages. For instance, learning mapping models from



(a) Frontal **(b)** 45° **(c)** 60° **(d)** 90°

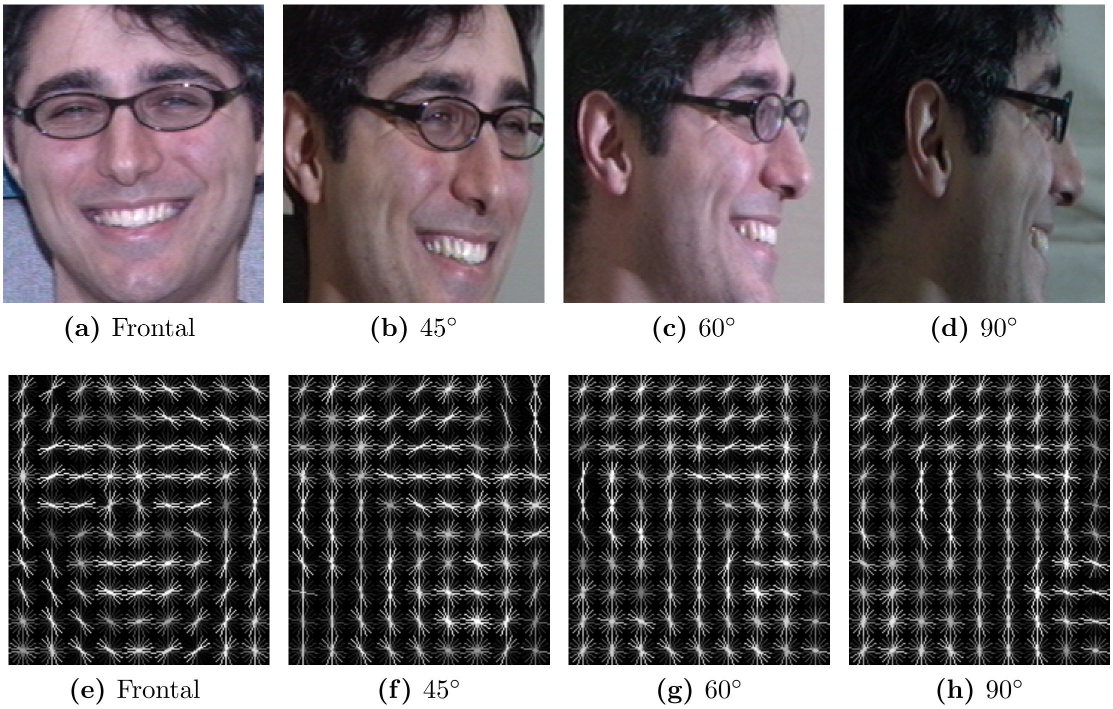
(e) Frontal **(f)** 45° **(g)** 60° **(h)** 90°

Figure 3.4: HOG features for supervised head pose recognition (BU3DFE dataset).

non-frontal samples in a specific head pose which has more homogenous data is much more efficient than a lot of samples in variant head poses. Moreover, the final classification on facial expression for a small subset in an specific head pose is faster than the classification through all viewpoints samples. However, this needs several classifications (based on the number of viewpoints) than a global classification. Nevertheless, our experiments in chapter 5 show that splitting data into the smaller categories has better performance. To this end, two kind of supervised and unsupervised splitting approaches can be employed. In the following, both methods are explained.

3.2.6.1 Unsupervised Splitting Data using K-means

A most popular method to separate data into the smaller groups with increasing the correlation of each subset members is by exploiting k-means clustering which is an unsupervised technique. K-means clusters data iteratively into the k categories based on the nearest neighbors (the similarity). We aimed to split data into the smaller groups where decrease the dissimilarity of head pose but the result showed that there is no improving on facial expression recognition because of no constraints to enforce clustering based on the head pose. Therefore, some of the clusters have been categorized based on the other variations (e.g. expressions, gender, etc.) which are not our goal. The results of k-means bring up that we need a categorizing technique capable to exploit head pose variations, hence, a



(a) Frontal (b) 45° (c) 60° (d) 90°

(e) Frontal (f) 45° (g) 60° (h) 90°

Figure 3.5: HOG features for supervised head pose recognition (Multi-PIE dataset).

supervised classification evaluated for this purpose.

3.2.6.2 Supervised Splitting Data by means of SVM

While supervised learning methods are very efficient approaches for classification, they can be used for categorizing data according to the training labels. A basic supervised classification model contains test and training data with correspondences labels. A model can be learned from training data and then can be used for estimating a new unseen test sample. Therefore, it is possible to re-label the training data according to the head pose. This means, the training samples from similar viewpoints have the same labels, we learn then our head pose system with training data and new labels based on the viewpoints. Therefore, our model estimates for relevant label which is a possible viewpoint for each input test sample.

To this end, a straightforward procedure has been applied that receives concatenation of employed both HOG and LBP features and re-label them according to the viewpoints. In addition, linear SVM [15] is employed for classification purpose. The supervised SVM classification achieves high performance for head pose categorizing due to the using hyperplanes in a high- or infinite-dimensional space. We achieved expectable results on several multiview datasets such a way that the categorized data with supervised methods (e.g. using SVM) has better precision than unsupervised techniques (e.g. clustering using k-means). Therefore, we employed supervised SVM classification with re-labeling based on the viewpoints to split multiview facial expression data into the several smaller groups in our system pipeline.

3.2.7 Supervised Facial Expression Recognition

Our multiview facial expression recognition system contains supervised classification for discriminating the expressions. There are several classifiers that can discriminant the extracted features to recognize the expressions while they have almost some advantages and a few disadvantage. Most popular supervised classifiers are Support Vector Machines (SVM), Adaboost, Random forests, Deep learning and etc. Without any doubt, the SVM is a straightforward and well known classifier that can be used efficiently in discrimination tasks. The last part of our pipeline system rely on recognition of facial expression, to this end, we employed a linear SVM [15] for this purpose. As our research thesis focused on addressing multiview facial expression recognition, we assume that the state-of-the-art on FER can be achievable. Therefore, we only define uncomplicated system for classification using basic SVM and the same setup in whole of MFER system. We compare our results in both FER and MFER systems based on such classification settings.

3.2.7.1 Hierarchical Multiclass SVMs

An idea to improve the classification accuracy on multiclass classification is hierarchical strategy. This idea relies on categorizing the easiest classes and ignore them from the original pool, then repeat it up to categorizes all classes. There are six basic expressions in facial expression recognition problem, therefore, our hierarchical classification categorizes two classes with maximum discrimination in each level and pass the remained classes to the next level. According to the six expression classes and recognizing two classes in each level, we have a three level hierarchical classification as shown in Figure 3.6. However, finding the most discriminant classes needs to compute the SVM probability estimations in each level, which means that the idea of hierarchical classification is more expensive in terms of complexity. Nevertheless, in our experiments, the hierarchical SVM classification improves the accuracy of frontal facial expression recognition.

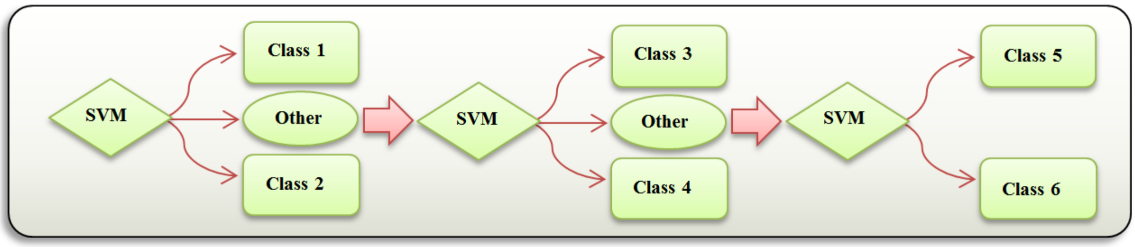


Figure 3.6: Three levels hierarchical SVM classification.

3.3 Discussion

We explained our system pipeline including: face detection, features extraction, features dimension reduction, features encoding, mapping approach and classification. For a big system like our MFER, we need to benefit from several smaller components. In our work, the components were themselves previously open problems in computer vision, some of them are still open as they are not completely solved. Although, they have efficient performance (e.g. face detection) near to solved. Nevertheless, we have introduced our setup system even by means of one or two components which are not completely solved. We evaluated our contributions using such components but these components have been analyzed for the best parameters (e.g. classification). Our evaluations on the same settings show that the concatenation of HOG and LBP has the best precision for our problem. Moreover, both PCA and Sparse representation can be used for dimension reduction that we used PCA for some of our early works and sparse coding for newer publications as sparse coding has other benefits such as robustness on outliers. Both of these processes have been done on the detected face parts using the state-of-the-art approaches. We finally employed a simple yet efficient classification for expression recognition. By exploiting such system pipeline we focus on mapping approaches in the following chapter.

4
Facial Mappings

Contents

The problem of multiview facial expression recognition or briefly MFER and other similar problems (e.g. multiview face recognition) suffer from processing of non-frontal samples. However, it seems that the imperative facial information is available in these problems for the related processing where human can understand and discriminant from even half of the face. But, machines (e.g. cameras, facial systems) do not understand as well as human. Therefore, the lack of the capability of understanding such samples could be comes from current facial analysis techniques. Focusing on frontal face analysis techniques (e.g. face detection [124]) shows that our assumption on disability of frontal approach to understand and analysis of non-frontal faces even with the imperative information is correct. Because these approaches assumed that the whole of the face is available (whilst the whole of the face is not really required due to the symmetric structure of the face). Nevertheless, that approaches account on all of the facial parts. For instance, a successful human face detector assumes that a face has two dark parts on top (eyes), a bright section at center (nose) and a curve at bottom which is almost dark (mouth)! Therefore, a face that does not satisfy these assumptions cannot be detected. In contrast, any other objects that fulfill these assumptions will detected as a face mistakenly. For this reasons, new approaches were striving to provide invariant approaches (e.g. head pose invariant face detectors [144]) almost try to handle the basic assumptions (use the information like eyes, nose, mouth, etc.) such a way that first, consider these parts as a positive signs which are useful for detection and second, do not make a strict constraint on the position of these objects (e.g. part based analysis approaches [31]). That means, such methods assume that these parts should be available but it is possible that the position of one or some of them is moved from our expectation or even missed. Therefore, the possible directions for our research are:

1) Extending the related assumptions for facial expression recognition to provide all possible hypotheses of multiview facial expression recognition, or,

2) Exploring transformations from non-frontal faces to the frontal, in a way that the mapped faces are understandable with current frontal face analysis approaches.

As there are many head pose possibilities (unlimited in real world), we aimed to select the second direction which means we are seeking for approaches that map faces (face features) from arbitrary head pose to the frontal and use then current available frontal facial expression recognition system to express the expressions. To this end, we introduce our proposed approaches in the following of this chapter.

4.1 Linear Regression Mapping

Linear regression is one the most popular statistical techniques for data analyzing. It is the study of relationship between two sets of independent (e.g. X) and dependent (e.g. Y) variables. Finding a linear relation between two sets of X and Y can be represented using linear regression as:

$$\arg\min_{T} \|Y - TX\| \quad , \tag{4.1}$$

Where transformation T can actually be computed in closed-form using the well-known formula:

$$T = Y(X^T X)^{-1} X^T \quad . \tag{4.2}$$

That T is the best linear transformation which maps X into the Y. This can be exploited for our specific problem if we assume that there is at least a relation between two sets of frontal and non-frontal faces. Therefore, let X be a set of vectorized features in size of $(q \times N)$ from N aligned non-frontal faces that is extracted and concatenated by appearance-based features HOG and LBP from the faces. X_θ is a subset of non-frontal features and X_{0^c} is the correspondence frontal. A global linear regression between non-frontal and frontal facial features is an adopted form of equation 4.2 as:

$$\arg\min_{T} \|X_\theta - TX_{0^c}\| \quad , \tag{4.3}$$

Where

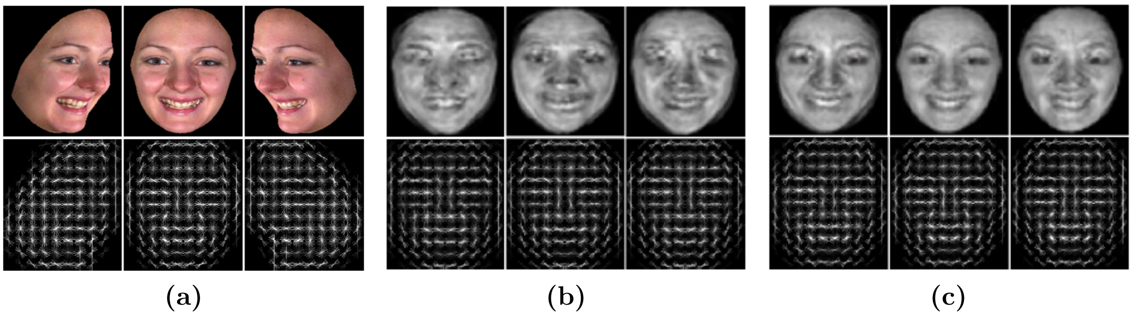


Figure 4.1: Linear mapping of facial features to the frontal. The first row is the analysis on the raw features and the second row visualizes the correspondences on HOG features. (a) Original input faces and their features in three variant viewpoints, (b) The results by means of a linear global mapping, (c) The results by means of a pose specific linear mapping.

$$T = X_{0^c}(X_\theta^T X_\theta)^{-1} X_\theta^T \quad , \tag{4.4}$$

Global transformation T maps non-frontal features into the frontal where the transformed features have the minimum differences from correspondence frontal features.

It has been influenced from all non-frontal features which means that T can map facial features from head pose to either left or right. Clearly, exploring such transformation that can map facial features from all viewpoints to frontal has more error than an specific head pose into the frontal. On the other hand, the problem of multiview facial expression recognition is a nonlinear problem due to the ethnicity, gender, expressions, skin tone, and other variations. Therefore, we proposed to use a specific mapping function for each viewpoint that means we perform M transformations where $M \in \mathbb{N}$ is the number of arbitrary viewpoints. As the variation of independent and dependent variables will decrease for each subset due to the similar head pose, linear regression can perform more efficient transformations. Therefore, we propose a limited form of global linear regression as pose-specific mapping model which transform non-frontal features from each arbitrary viewpoints individually to the frontal.

Given $X_{\theta_i} = [I_1^{\theta_i}, I_2^{\theta_i}, \ldots, I_N^{\theta_i}]$ is a matrix of size $(q \times N)$, and refers to the N vectorized facial features denoted by $I_k^{\theta_i} \in \mathrm{IR}^{(q\times 1)}$. Note that I_k^0 and $I_k^{\theta_i}$ are vectorized features of the k^{th} facial expression image of the training data from the same person in different poses. Based on this, we define pairwise sets of training data, X_0 and X_{θ_i}, where the former is the set of frontal and latter is a set of correspondence non-frontal features. The transformations T_θ can be estimated during an offline stage using the same formula as before:

$$T_{\theta_i} = X_{0^c}(X_{\theta_i}^T X_{\theta_i})^{-1} X_{\theta_i}^T \quad i = 1, 2, \ldots, M \quad , \tag{4.5}$$

Where X_{θ_i} is the matrix of the feature vectors for the training images with the head

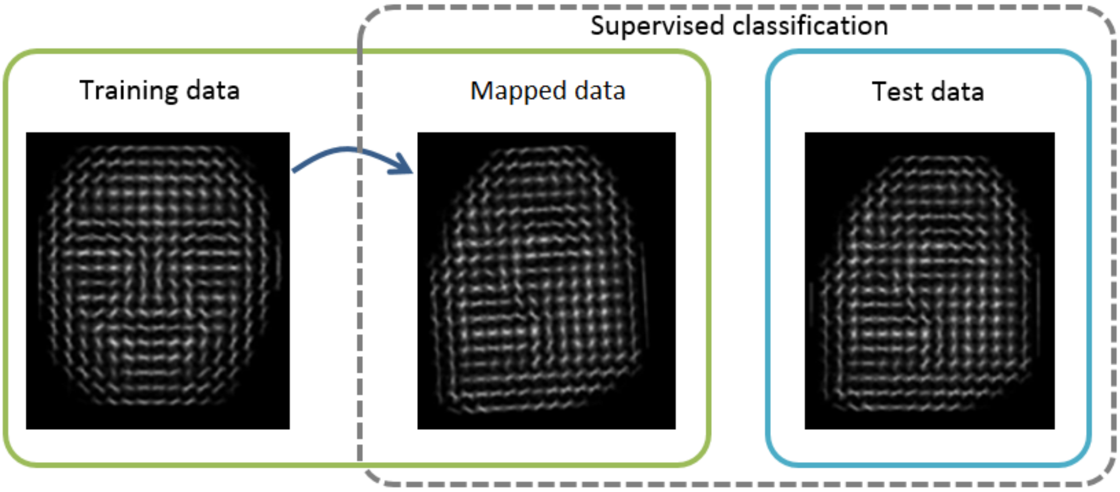


Figure 4.2: Inverse mapping with linear regression. Map frontal training data into the correspondence non-frontal then employ transformed data for training the classifier.

is under pose θ_i. Given a feature vector x_{θ_i} computed for an image of a face seen under pose θ_i, we can predict the corresponding feature vector $\tilde{x}_{\theta_i \to 0}$ as if the face was under a frontal view in the image by computing:

$$\tilde{x}_{\theta_i \to 0} = T_{\theta_i} x_{\theta_i} \quad .$$

Figure 4.1 illustrates a visualization of global and pose specific linear regression transformation. As can be seen, the transformed features by pose specific mapping looks much more reasonable.

4.1.1 Inverse Mapping with Linear Regression

The idea of pose specific mapping can reasonably transforms non-frontal facial features into the frontal. Nevertheless, such mappings are erroneous due to the linear regression. The linear regression transformation provides the best linear approximation between two sets of dependent and independent data, therefore, mapping error is computable from equation 4.5 as:

$$\widehat{X}_0 = T_{\theta_i} X_{\theta_i} \quad , \tag{4.6}$$

$$E_{\theta_i} = X_0 - \widehat{X}_0 \tag{4.7}$$

This error rate shows that our proposed linear transformation loses some information from test data during the run, therefore, missed data will affects on the mapping and facial expression recognition. Consequently, an alternative idea that address this problem is using inverse mapping. It is instead by transforming training data which means we use similar mapping as introduced in equation 4.5 but for mapping frontal data to correspondence non-frontal. This idea has two advantages but two side effects; the advantages are: 1) Transforming frontal training data into the non-frontal could be done as an offline processing which increases the speed time during the test, 2) The test data does not loose any information because we map training data, therefore, the maximum available information will be considered for expression recognition. In other side, inverse mapping has not the properties of simple direct mapping, for instance, direct mapping approaches that transform non-frontal facial features into the frontal are almost stable with number of training data because the facial features in each arbitrary viewpoints will be mapped to the frontal. The second side effect is that the inverse mapping still looses some information but during the training where it makes our solution yet complication. Figure 4.2 demonstrates an overall idea of inverse mapping.

4.2 Sparse Coding Regression Mapping

As introduced by encoding approaches in 3.2.4, sparse coding is an effective representation which reduces not only the dimensionality of facial features but also decreases the outliers. Reducing the dimensions of facial features augments the speed of our system and improves the memory usage. we are interested in finding a reconstructive dictionary given the training features X by minimizing:

$$\|X - DS\|_2^2 \quad s.t. \ \|s_i\|_0 \leq \Gamma \tag{4.8}$$

Where $D \in \mathbb{R}^{(q \times s)}$ is the dictionary of codebooks, each column representing a codebook vector, and $S \in \mathbb{R}^{(s \times N)}$ the matrix of encoding coefficients. Γ is the sparsity constraint factor, defining the maximum number of non-zero coefficients per sample. We apply K-SVD [1] as dictionary learning algorithm and orthogonal matching pursuit (OMP) [120] as an efficient way for solving the coding of new test samples, given a fixed dictionary.

We use then the sparse representation of facial features for learning a linear model using regression approximation, to this end, each feature vector has been represented in sparse representation using OMP and the aid of dictionary. During the test, we have similar procedure which is that each input feature vector must be expressed as a vector of sparse representation (using K-SVD and dictionary) then map to the new space by means of learned model. Hence, equation 4.4 can be rewrite as:

$$T^S = S_{0^c}(S_\theta^T S_\theta)^{-1} S_\theta^T \quad , \tag{4.9}$$

Where S_θ is the matrix of the feature vectors for the training images in sparse representation where the head is under pose θ and S_{0^c} is a representation of sparse code for correspondence frontal features. T^S is a general (global) transformation of sparse features from non-frontal to frontal viewpoint.

Consequently, we found that again M partial projections which approximate linear

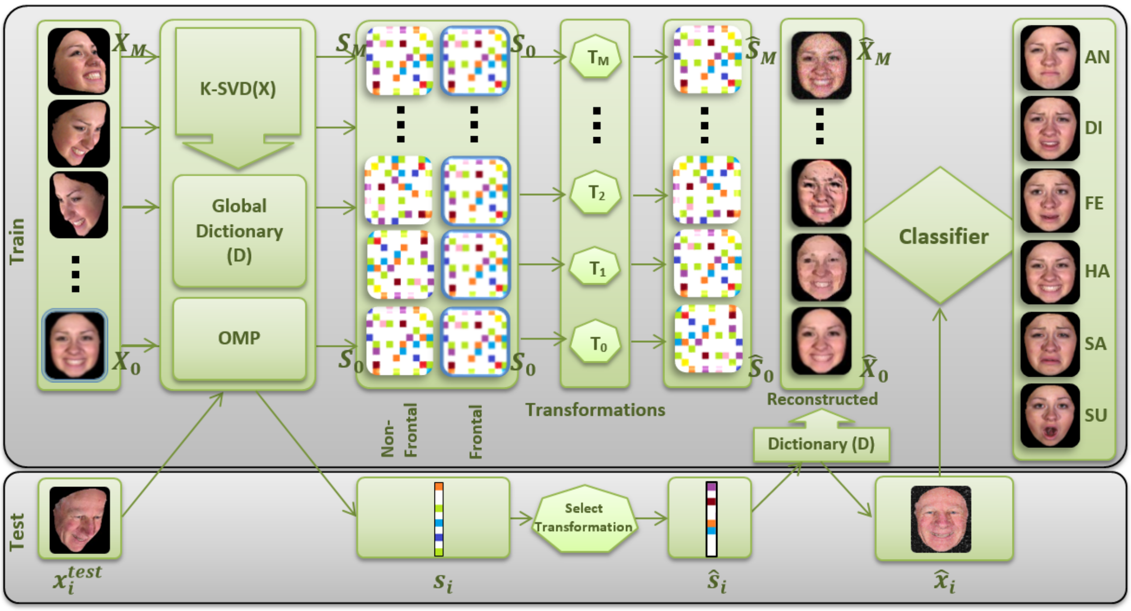


Figure 4.3: The overall structure of pose specific regression mapping of sparse features. Train: A global dictionary is trained with K-SVD and facial features are encoded. Transformations are estimated using local collections of frontal and non-frontal sparse features, and finally, features are reconstructed using the global dictionary. Test: Input sample is encoded via OMP and encoded vector mapped to frontal using appropriate transformation. Mapped vector is then reconstructed using dictionary and finally it is classified for final expression recognition.

regression for each part of data can be more effective than a general sparse coding. In the following section we introduce pose specific sparse coding for transforming non-frontal facial features to the frontal based on the viewpoints.

4.2.1 Mapping by means of Pose Specific Sparse Features

Similar to extending global linear transformation for pose specific mapping, we can extend general sparse based transformation into the partial (pose specific) mappings. Thus let S_0 be a set of sparse features of frontal facial features and $S_{\theta_1}, S_{\theta_2}, \ldots, S_{\theta_M}$ are M sets of sparse features of non-frontal facial features where all sets have the same number of samples. Equation 4.5 can be rewritten for sparse features as:

$$T^S_{\theta_i} = S_{0^c}(S^T_{\theta_i} S_{\theta_i})^{-1} S^T_{\theta_i} \quad i = 1, 2, \ldots, M \tag{4.10}$$

$$\widehat{S}_{\theta_i} = T^S_{\theta_i} S_{\theta_i} \tag{4.11}$$

Again $T^S_{\theta_i}$ is i^{th} transformation which has been estimated using correspondent sparse features. $\hat{S}_{\theta_i}$ defines the transformed sparse codes, and the approximated features of mapped frontal view can be reconstructed using the global dictionary D by means of:

$$\widehat{X}_{\theta_i} = D\widehat{S}_{\theta_i} \tag{4.12}$$

The overall structure of our regression mapping using pose specific of sparse features is illustrated in Fig. 4.3. We have evaluated the performance of pose specific mapping of sparse features in comparison with other regression-based approaches in chapter 5. As we explained in 3.2.4, we employed OMP which is much efficient than MP for representing each input sample. Nevertheless, OMP which is an iterative algorithm is still expensive in terms of time complexity, therefore, an idea to enhance our proposed sparse-based mapping is improving OMP algorithm or replacing it by another idea that select the best codebooks to represent input sample as sparse features. To this end, we propose an alternative idea by ignoring OMP and replace it via ridge regression in the following section.

4.2.2 Mapping by means of Fast Pose Specific Sparse Coding

As mentioned in previous section, OMP is used to find the best encoding of a new test sample regarding to the dictionary D. However, OMP is an extension of Matching Pursuit (MP) with better results than standard MP but it needs more computation. It has been shown by previous research [132] that the sparsity constrain must not be needed during the reconstruction. Therefore, given a sparse codebook create by equation 4.8, we can reformulate the solution of finding best encoding S by replacing $l0 - norm$ for the coefficients, which could be called Ridge Regression [119] as:

$$\|X_{\theta_i} - DS\|_2^2 + \lambda \|S\|_2 \tag{4.13}$$

Where D is global dictionary and X_{θ_i} is a set of input data regarding to the i^{th} pose. However, it eliminates the rules leading to sparsity but we are using $l2 - norm$ because of two clear reasons: first, to avoid over fitting during the regression; second, to stabilize transformations specially when we know there are collinearity between the frontal and correspondence non-frontal features. Moreover, as mentioned before, the sparsity constrain must not be needed during the reconstruction due to using regression in our work. The parameter λ also allows us to detract the singularity problem. equation 4.13 is a ridge regression model and the solution is given by least square solution as:

$$S_{\theta_i} = D(X_{\theta_i}^T X_{\theta_i})^{-1} X_{\theta_i}^T \quad i = 1, 2, \ldots, M \tag{4.14}$$

$$\widehat{y}_{\theta_i} = S_{\theta_i} X_{\theta_i} \tag{4.15}$$

Where $\widehat{y}$ is a representation of input data onto the dictionary proposed instead of using OMP. Although, the dictionary can be computed offline but it is clear that input test samples should be transformed to sparse representation during the processing (online) therefore proposed algebraic computation in equation 4.14 is much faster than OMP which is an iterative algorithm and updates coefficients after every steps. Investigations of pose specific sparse coding and fast pose specific sparse coding are described in chapter 5.

4.3 Non-linear Pose Specific Mapping

A linear mapping, even a pose-specific one as described in the previous sections, can capture variations from the non-frontal view to the frontal only in a very limited way. Whereas, the problem of multiview facial expression recognition and its mappings from non-frontal to frontal is a nonlinear problem due to the numerous variations among of the data (e.g. gender, expression, skin tone, ethnicity, personality, etc.) that they have originally nonlinear behavior. We therefore introduce a more complex mapping model, based on polynomial kernels. Let us first introduce h_n, a function that applies to a feature vector such that:

$$h_n(x) = \begin{bmatrix} x^{\cdot 0} \\ x^{\cdot 1} \\ \vdots \\ x^{\cdot n} \end{bmatrix} \quad , \tag{4.16}$$

Where $x^{\cdot i}$ is the element-wise exponent i applied to each element of x. Let $h_n(X_\theta)$ be the matrix whose columns are the results of $h_n(.)$ applied to the feature vectors for the training images the head is seen under pose θ.

We can now compute a new mapping T^h_θ that applied to polynomial kernels of the feature vectors:

$$T^h_\theta = X_0(h_n(X_\theta)^T h_n(X_\theta) + \lambda I)^{-1} h_n(X_\theta)^T \quad . \tag{4.17}$$

The term λI is required for regularization, otherwise the system would be under-constrained. In practice we use $n = 10$, which we empirically found to provide a good trade-off between mapping accuracy and non-overfitting.

Given a feature vector x_θ computed for an image of a face seen under pose θ, we can predict the corresponding feature vector $\tilde{x}^0_\theta$ as if the face was under a frontal view in the image by computing:

$$\tilde{x}_{\theta\to 0} = T^h_\theta h_n(x_\theta) \quad .$$

We can then similarly extend pose-specific based mappings and sparse coding based mappings with polynomial kernel based idea as introduced in equation 4.17, therefore, we have nonlinear kernel-based pose-specific mapping as:

$$T^h_{\theta_i} = X_0(h_n(X_{\theta_i})^T h_n(X_{\theta_i}) + \lambda I)^{-1} h_n(X_{\theta_i})^T \quad . \tag{4.18}$$

Consequently, nonlinear kernel-based pose-specific mapping for sparse features in each head pose can be rewritten as:

$$T_{\theta_i}^{h_s} = S_0(h_n(S_{\theta_i})^T h_n(S_{\theta_i}) + \lambda I)^{-1} h_n(S_{\theta_i})^T \quad . \tag{4.19}$$

Our evaluations on regression-based approaches including linear mapping, pose specific and sparse representation show that the current idea (nonlinear pose specific mapping) outperforms not only the state-of-the-art but also the other proposed methods in terms of accuracy. More details are discussed on experimental results in chapter 5.

4.4 Mapping Forests (MF)

At the beginning of the current chapter we described that the regression based mapping approaches are appropriate methods that can provide solutions for multiview facial expression recognition. Pose specific regression mapping outperformed the linear regression mapping and non-linear mapping performs transformations which are more accurate than pose specific regression. We expected such results due to the non-linearity of the MFER problem; therefore, we can estimate more accurate transformations if we provide still precise nonlinear mappings that can learn our MFER behavior very well. Obviously, better transformations lead to better expression recognition.

As reviewed in section 2.6, decision forests is an efficient structure for such mappings due to the possibility of learning the mappings very precise. Nevertheless, we need a continues structure (e.g. regression forests) because of our continues problem. Therefore, we benefit from the idea of regression forests. In addition, an important attribute of the regression forests is the stability of the forests on outliers due to the combination of lots of trees. Such robustness is valuable in our approach since the features are basically nonlinear and natural. In other side, an exquisite intrinsic attribute of the (random or regression) forests is the information gain evaluation at each node of the trees during the training; such a way that confirm selecting the best splitting feature for the splitting function. It can guarantee the maximum discrimination for both subbranches. This is similar to determine much accurate mappings in our mapping problem.

Therefore, mapping forests which is inspired from regression forests consist of randomize trees which are adapted for our specific problem. It is for predicting the continues value of the response variable **y** from predictor variable **x** where realizing the relationship between **x** and **y**.

We split data into the smaller subsets in each node of the trees and perform pairwise ridge regression for each subset of data at leaf nodes to learn the best optimized mapping solution. The optimization problem is such that the objective function is computed by minimizing mapping error between pairs of data. Therefore, we learn a mapping function $f(.)$ which is depended on input data X_{NF}. On the leaves, we have multiple non-linear mapping functions that are specified for the data trough the depth of the trees. Assuming

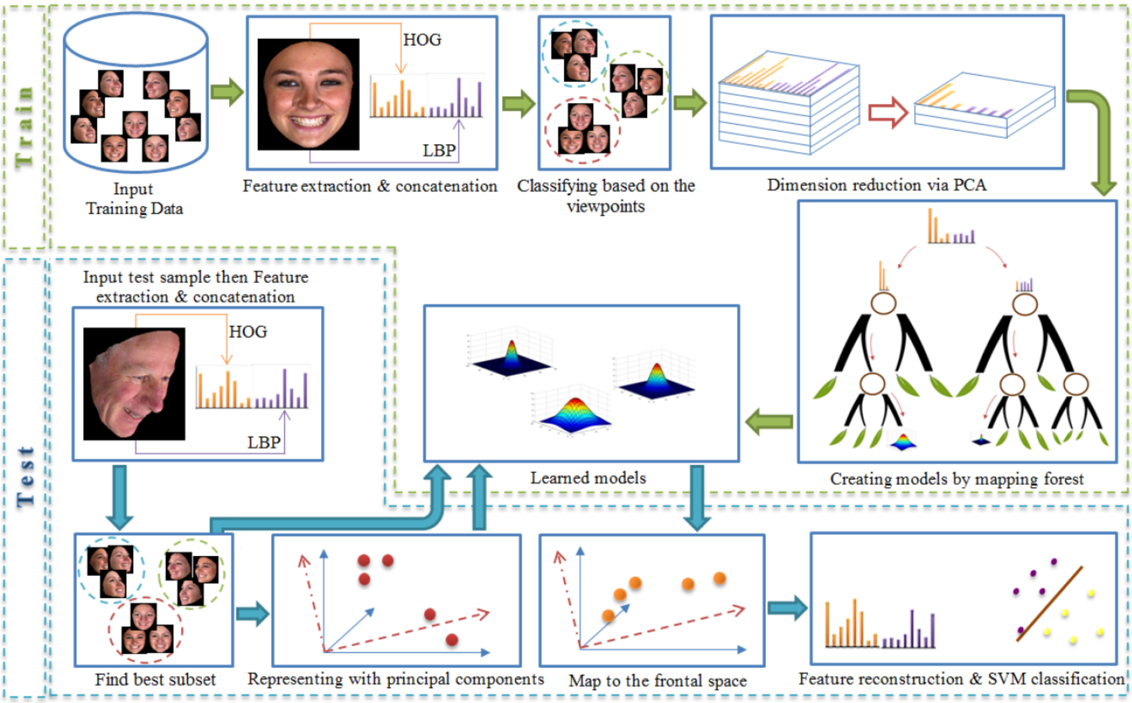

Train
Input
Training Data
HOG
LBP
Feature extraction & concatenation
Classifying based on the viewpoints
Dimension reduction via PCA
Creating models by mapping forest
Learned models
Test
Input test sample then Feature extraction & concatenation
HOG
LBP
Find best subset
Representing with principal components
Map to the frontal space
Feature reconstruction & SVM classification

Figure 4.4: The overall structure of Mapping Forests. In the training step, basic features are extracted and concatenated into the feature vectors for each sample. Then they are classified based on the viewpoints also dimension is reduced by means of PCA. Mapping forests estimate the transformation models from pairs of features (non-frontal, frontal) and provide them for test data. In the test step, the features are extracted from a new unseen sample and concatenated in the same as in the training step. The best subset is then selected based on the viewpoint also features are represented in the new feature space provided by principal components (provided in the training). We transform the new features to the frontal feature space using the detected learned models. Finally, we reconstruct and classify the transformed features in terms of expression recognition.

the loss of mapping by function L:

$$m = \underset{f(X_{NF})}{\arg\min} \sum_{n=1}^{N} L(X_{Fr}, f(X_{NF})X_{NF}) \tag{4.20}$$

Mapping m is provided by replacing loss with squared value:

$$m = \underset{f(X_{NF})}{\arg\min} \sum_{n=1}^{N} \|(X_{Fr} - f(X_{NF})X_{NF})\|^2 \tag{4.21}$$

Therefore, we learn a mapping model m from training data X_{Fr} which efficiently matched with the input non-frontal data X_{NF}. To this end, we solve the regularized least square problem with well-known closed form solution:

$$\hat{X}_{Fr} = (X_{NF}^T X_{NF} + \lambda I)^{-1} X_{NF}^T X_{Fr} \tag{4.22}$$

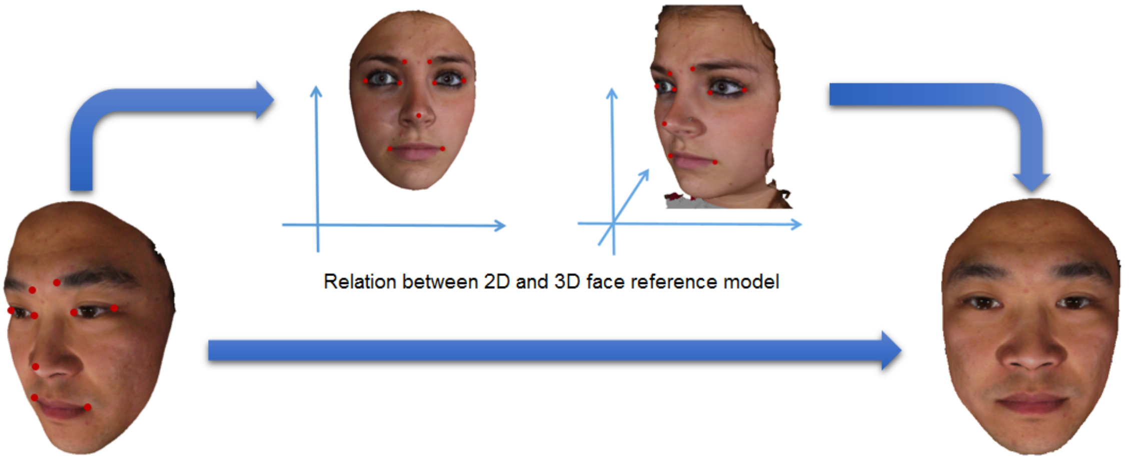


Figure 4.5: 3D frontalization: The overall structure of frontalization using 3D model and 2D reference face image.

In equation 4.22, X can also be replaced by non-linear kernel $\Phi(X)$ where the polynomial and Gaussian are two popular kernel functions. We propose to use MF for minimizing loss function in a way that selects a branch of each node in the tree with maximum matches with input vector. Therefore, we provide ridge regression similar to the equation 4.22 as objective function such that decreases the mapping error in each node. The average of the mapping functions at the leaves from all trees will be considered to present our non-linear transformation. Note that the splitting functions are based on the extracted features as discussed in equation 2.11. The response function is $r_\theta(\mathbf{x}) = \mathbf{x}[\theta_1] - \theta_{th}$ over each feature vector $\mathbf{x}$. Also, the other parameters (e.g. max depth, number of trees, etc.) are empirically balanced which are described in details by the following chapter. The overall structure of proposed mapping forests is illustrated in Figure 4.4.

4.5 3D Face Mapping and Inpainting Unavailable Parts

To provide a multiview facial expression recognition system, we proposed several approaches that work on the raw or feature space, where they mapped extracted non-frontal to the approximated frontal. In this section, we propose our approach on mapping model by means of 3D facial information.

The terminology of the 'frontalization' refers to the process of constructing a frontal face from arbitrary non-frontal views that is first introduced by Hassner et al. [37]. Frontalization can build the frontal faces either by means of approximating the 3D facial shapes [143] or by synthesizing the frontal facial views [37]. Both approaches have several advantages and a few side effects, to this end, we have proposed a complementary approach that improves the quality of the results in both techniques.

In fact, both approaches suffer from compensating unavailable parts, although both used symmetric information to fill unavailable regions but first, the results can be still improved and second, there is no solution for asymmetric regions (e.g. nose, whole of the mouth, etc.).

Our complementary approach relies on exploiting learning method to find a matched face texture and transfer unavailable parts from such face. We named the exploited face as guidance face (GF) which can be extremely match with the face in the query by providing some constraints. The constraints are based on our prior knowledge such as gender, skin tone and ethnicity. Of course, increasing the constraints leads to discover an admirable GF while our model can easily upgrades or reduces the constraints. Although, we choice the best match patch using an optimization algorithm, nevertheless, we found that these three constraints are smartly desirable for recognition purpose. Moreover, transferring information from GF to the query face image is efficiently done by means of belief propagation (BP) which is an eligible optimization algorithm.

Estimating the missing parts of the face using prior knowledge is an idea to improve the state-of-the-art. In this part, our major novelty is exploiting learning based approach to discover a guidance face sample from training data that can satisfy our defined constraints, our contribution is also including that such guidance face transfers the missed knowledge by optimizing the cost functions of constraints using belief propagation algorithm. Moreover, intrinsic image components (albedo and shading) are used due to the RGB instabilities while the intrinsic components are more reliable with image editing. Where, the variations on scene lighting can be produced by adjusting the shading component and reflectance component (albedo) can modify the face colors and textures.

4.6 Compensating Unavailable Face Parts

As introduced, the 3D mapping approach that focuses on the frontalizing face image into the frontal has some unavailable or occluded parts. These occlusions affects dramatically on the MFER precision. We reviewed two approaches (frontalization and 3DMM) for frontalizing in chapter 2. Both of theses methods need a post-processing step for compensating unavailable facial parts.

The popular concept for compensating unavailable parts is employing symmetric information. Although, symmetric information are useful and have advantages but it is not enough due to the complicated challenges on frontalized faces (e.g. for such a part of a face which is not symmetric).

The goal is to inpaint the occluded/unavailable regions of the face to generate a complete face. Our approach first performs facial analysis on the input face to infer the high-level attributes such gender, ethnicity, skin tone, etc. It then uses these metrics to retrieve a guidance face from the face dataset. In addition, the input face image will be decomposed into intrinsic shading and reflectance layers. Our approach then performs patch-based inpainting separately on the shading and reflectance images, using the guidance face, symmetry measures and local cues from the known regions. The inpainted shading and reflectance images are then combined to produce the final, complete face.

4.6.1 Problem Formulation

We formulate our patch-based inpainting as an MRF cost minimization problem. Let I be the input face image and I_{g} be the guidance face image selected by our approach. Let $\hat{I}$ and $\hat{I}_{\text{g}}$ be the horizontally-flipped images of I and I_{g}. Denote the set of hole patches as $\mathbb{H} = \{\mathbf{H}_l\}$ and the set of known patches as $\mathbb{K} = \{\mathbf{K}_m\}$, which contains patches extracted from I, I_{g}, $\hat{I}$ and $\hat{I}_{\text{g}}$. $\mathbf{H}_l$ is itself a set containing all pixels that lie within the patch, with each pixel indexed by local patch coordinate $\mathbf{p}$. Likewise for $\mathbf{K}_m$. The optimizer aims to select some of the known patches for repairing the hole patches.

For notational convenience, we define $H_l(\mathbf{p})$ as the pixel location in image coordinates, such that $I(H_l(\mathbf{p}))$ is the pixel value on image I. Likewise for $K_m(\mathbf{p})$, we define:

$$T(K_m(\mathbf{p})) = \begin{cases} I(K_m(\mathbf{p})) & K_m \text{ is from } I \\ \hat{I}(K_m(\mathbf{p})) & K_m \text{ is from } \hat{I} \\ I_{\text{g}}(K_m(\mathbf{p})) & K_m \text{ is from } I_{\text{g}} \\ \hat{I}_{\text{g}}(K_m(\mathbf{p})) & K_m \text{ is from } \hat{I}_{\text{g}} \end{cases}. \tag{4.23}$$

Also, H_l^{-1} returns the corresponding known patch's index, such that $\mathbf{K}_{H_l^{-1}}$ is the patch that repairs $\mathbf{H}_l$. The MRF total cost function is defined as:

$$\begin{aligned} \mathcal{C}(\mathbb{H}) = \quad & w_{\text{data}}\mathcal{C}_{\text{data}}(\mathbb{H}) + w_{\text{pair}}\mathcal{C}_{\text{pair}}(\mathbb{H}) + \\ & w_{\text{sym}}\mathcal{C}_{\text{sym}}(\mathbb{H}) + w_{\text{g}}\mathcal{C}_{\text{g}}(\mathbb{H}), \end{aligned} \tag{4.24}$$

where $\mathcal{C}_{\text{data}}(\mathbb{H})$, $\mathcal{C}_{\text{pair}}(\mathbb{H})$, $\mathcal{C}_{\text{sym}}(\mathbb{H})$ and $\mathcal{C}_{\text{g}}(\mathbb{H})$ are respectively the data, pairwise, symmetry prior, and guidance face prior cost terms. We set $w_{\text{data}} = 1.0$, $w_{\text{pair}} = 0.5$, $w_{\text{sym}} = 0.5$ and $w_{\text{g}} = 0.1$ in our experiments unless otherwise specified. Each energy term is formulated as follows.

4.6.2 Cost Terms

Data Term $\mathcal{C}_{\text{data}}(\mathbb{H})$. We define a data cost to encourage the known patch used for repairing a hole patch to match well with the intersection between the hole patch and the known region (refer to Figure 4.6(a)). Let $\mathbf{H}_l' \in \mathbf{H}_l$ be the intersection (blue in Figure 4.6) of the hole patch $\mathbf{H}_l$ with the known region. The data cost is defined so that the selected known patch matches well with the pixels of the known region at the intersection:

$$\mathcal{C}_{\text{data}}(\mathbb{H}) = \frac{1}{Z_{\text{data}}} \sum_l \sum_{\mathbf{p} \in \mathbf{H}_l'} \|I(H_l(\mathbf{p})) - T(K_{H_l^{-1}}(\mathbf{p}))\|^2, \tag{4.25}$$

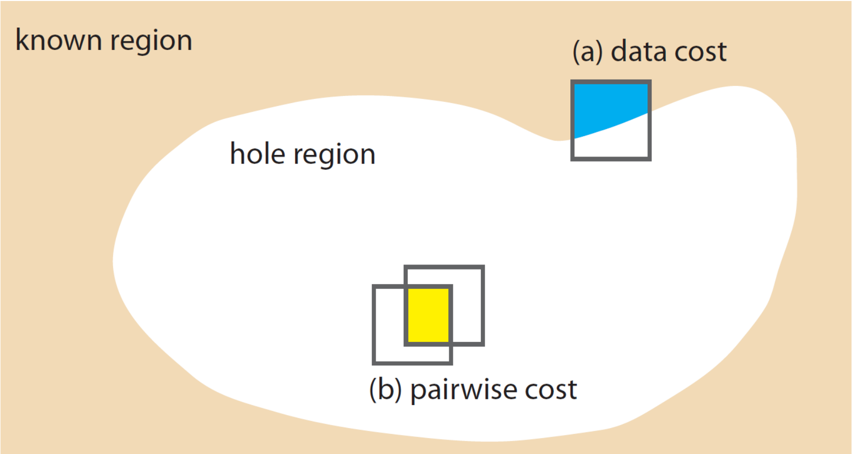


Figure 4.6: An illustration of data and pairwise costs. (a) The data cost encourages the known patch used for covering a hole patch to match well with the intersection (blue) between the hole patch and the known region. (b) The pairwise cost encourages two known patches used for covering two hole patches to match well in their overlapping region (yellow).

where Z_{data} as the normalization factor is the total number of pixels in the known region that are covered by a hole patch.

Pairwise Term $\mathcal{C}_{\text{pair}}(\mathbb{H})$. We define a pairwise cost to encourage two known patches used for covering two hole patches to match well in their overlapping region (refer to Figure 4.6(b)). Suppose $\{\mathbf{H}_{l1}, \mathbf{H}_{l2}\}$ is a pair of overlapping hole patches, and $\mathbf{K}_{H_{l1}^{-1}}$ and $\mathbf{K}_{H_{l2}^{-1}}$ respectively denote their repairing known patches. Suppose also that pixel $\mathbf{p}_a$ of $\mathbf{K}_{H_{l1}^{-1}}$ coincides with pixel $\mathbf{p}_b$ of $\mathbf{K}_{H_{l2}^{-1}}$, when $\mathbf{K}_{H_{l1}^{-1}}$ and $\mathbf{K}_{H_{l2}^{-1}}$ are pasted onto $\mathbf{H}_{l1}$ and $\mathbf{H}_{l2}$. We penalize solutions where the overlapping pixel values are inconsistent:

$$\mathcal{C}_{\text{pair}}(\mathbb{H}) \quad = \quad \frac{1}{Z_{\text{pair}}} \sum_{\{\mathbf{H}_{l1},\mathbf{H}_{l2}\}} \sum_{\{\mathbf{p}_a,\mathbf{p}_b\}} \|T(K_{H_{l1}^{-1}}(\mathbf{p}_a)) \quad - \quad T(K_{H_{l2}^{-1}}(\mathbf{p}_b))\|^2, \tag{4.26}$$

where Z_{pair} as the normalization factor is the total number of pixels in the overlapping regions of hole patches.

Symmetry Prior $\mathcal{C}_{\text{sym}}(\mathbb{H})$. We define a symmetry prior cost to evaluate the likelihood of using a known patch extracted from a certain location for repairing a hole patch at another location according to symmetry.

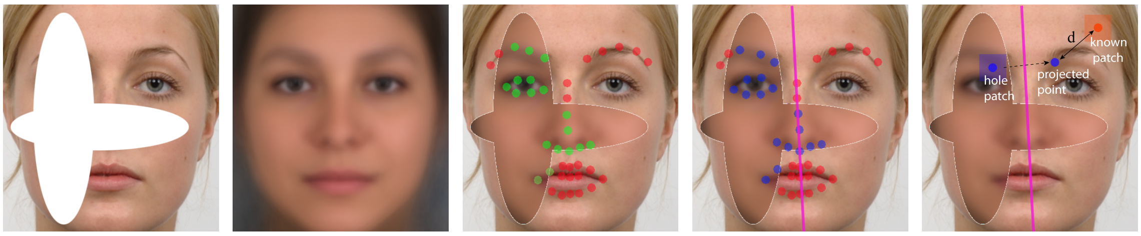
$$\mathcal{C}_{\text{sym}}(\mathbb{H}) = \frac{1}{|\mathbb{H}|} \sum_{l} \gamma(H_l, K_{H_l^{-1}}), \tag{4.27}$$

where $\gamma(H_l, K_{H_l^{-1}}) \in [0, 1]$ is a function measuring symmetry compatibility between a hole patch and a known patch. We describe how to define this function in 4.6.3.

Guidance Face Prior $\mathcal{C}_{\mathrm{g}}(\mathbb{H})$. We define a guidance face cost to penalize the use of known patches extracted from the guidance face over known patches extracted from the input face.

$$\mathcal{C}_{\mathrm{g}}(\mathbb{H}) = \frac{1}{|\mathbb{H}|}\sum_{l} \delta(K_{H_l^{-1}}), \tag{4.28}$$

where $\delta(K_{H_l^{-1}}) = 1$ if the known patch $K_{H_l^{-1}}$ is extracted from the guidance face; otherwise $\delta(K_{H_l^{-1}}) = 0$. Users can use the weight w_{g} to control how much repairing via the input face is favored over repairing via the guidance face. We set a small weight ($w_{\mathrm{g}} = 0.1$) to slightly favor the former.



(a) Input face (b) Mean face (c) Landmarks (d) Refined landmarks & estimated symmetry line (e) Symmetric measurement

Figure 4.7: Symmetry measurement. (a) Input face. (b) Mean face; (c) Landmarks detected on the input face that has been temporarily completed using the mean face. Landmarks on the mean face are shown in green, and those on the input face are shown in red; (d) Refined landmarks (blue) and estimated symmetry line (purple); (e) Symmetric measurement of a known patch (orange). The hole patch to be repaired (blue) first has its center point projected about the estimated symmetry line. Then the distance between the projected point and the center point of the known patch (orange) is measured. The longer the distance, the higher the symmetry cost.

4.6.3 Symmetry Prior

We describe the function $\gamma(H_l, K_{H_l^{-1}})$ for measuring the symmetry prior cost of using known patch $K_{H_l^{-1}}$ to repair hole patch H_l. The general idea is that the repairing should preserve symmetry.

Figure 4.7 illustrates our approach of measuring symmetry. First we locate landmarks on our input face to estimate a symmetry line. Since our input face may contain a large occlusion region, as a first step we use the mean face to temporarily complete the face. The mean face is selected by averaging the faces in the cluster that the input face belongs to (we will provide details about finding the cluster in 4.6.4). We then run a landmark estimator [11] to estimate the landmark locations on the temporarily completed face. The above process just gives us a rough estimation of the landmark locations, because the mean face may not resemble the true face well. We further refine the landmark locations by using the knowledge of relative positions of corresponding landmarks learned in the face dataset as follows.

We estimate the landmark locations for each face image in our dataset. Each landmark $\mathbf{u}$ stores a 2D position. As each landmark $\mathbf{u}$ should have a corresponding landmark $\mathbf{v}$ on the other side of the face according to symmetry[1], we can learn the projection matrix R from $\mathbf{u}$ to $\mathbf{v}$ by:

$$\arg\min_{R} \|V - R\,U\| \,, \tag{4.29}$$

where U is a matrix with each column storing the landmark $\mathbf{u}$ of each face image, and V is defined similarly. R can be estimated by ridge regression.

On the temporarily completed face, for each landmark $\mathbf{v}$ in the occlusion region, we find its corresponding landmark $\mathbf{u}$ on the other side of the face and compute the projected landmark $\hat{\mathbf{v}}$ by $\hat{\mathbf{v}} = R\,\mathbf{u}$. We refine the location of $\mathbf{v}$ as the average of $\mathbf{v}$ and $\hat{\mathbf{v}}$ (blue points in Figure 4.7(d)). Using all pairs of corresponding landmarks, we estimate a symmetry line for the face (purple line in Figure 4.7(d)). The symmetry line is computed such that if the landmarks on the left are projected about the line to the right, the overlap between the projected landmarks and the landmarks on the right is maximized.

The estimated symmetry line helps us to define our symmetry prior cost. Figure 4.7(e) shows an illustration. Given a hole patch H_l (the blue patch in Figure 4.7(e)) in the occlusion region, we want to evaluate whether a known patch $K_{H_l^{-1}}$ (the orange patch in Figure 4.7(e)) is good for repairing according to symmetry. To do this, we first project the center point of the hole patch about the symmetry line to the right-hand side. We then measure the distance d between the projected point and the center point of the known patch. We define our symmetry prior cost to increase with d:

$$\gamma(H_l, K_{H_l^{-1}}) = 1 - e^{\frac{d}{D}}, \tag{4.30}$$

where D as the normalization factor is the length of the diagonal of the image. In case the known patch is obtained from the guidance face instead of the input face, we compute the vector from the projected point to the nearest landmark point on the input face. We also compute the vector from the center point of the known patch to the same landmark point on the guidance face. Distance d is computed as the length of the difference of the two vectors.

4.6.4 Guidance Face Selection

We describe our approach for selecting a guidance face to replenish known patches for repairing. The guidance face patches are particularly important in case the occlusion

[1] For example, the landmark at the left mouth corner should correspond to the landmark at the right mouth corner. We manually define all the correspondences.

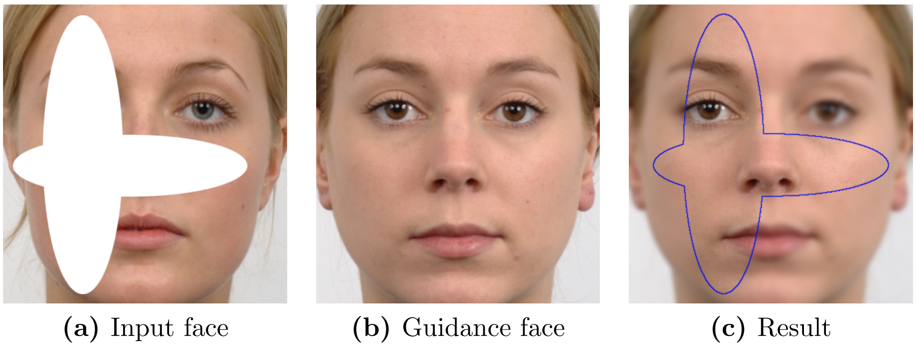

(a) Input face **(b)** Guidance face **(c)** Result

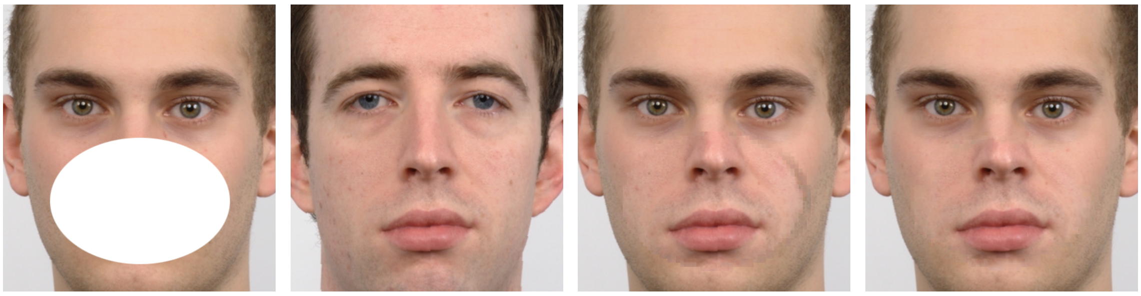
Figure 4.8: Contribution of the guidance face. (a) Input face; (b) Selected guidance face based on skin tone, gender, ethnicity and expression; (c) Candidate region from guidance face for contribution in occluded face.

mask covers a large region of the face (*e.g.*, the entire nose is occluded, or both eyes are occluded) such that no appropriate known patches can be extracted from the input face for repairing. Figure 4.8 shows an illustration.

The face images in our face dataset carry ethnicity, expression and gender labels. The skin tone of a face image is computed by running mean shift clustering on the RGB values of the pixels in the forehead and cheek regions, which discards outliers caused by sharp illumination. The value of the cluster's center is taken as the skin tone. We cluster the face images according to their skin tones using k-means clustering. For each cluster, we train separate linear SVM classifiers [15] for classifying ethnicity, expression and gender using the face images in the cluster. We use histograms of oriented gradients (HOG) as our features with a cell size of 20×20 pixels. The accuracies of our SVM classifiers for gender, ethnicity and expression are 91.67%, 89.5% and 81.33% respectively, as we cross-validate in training.

Given an input face image for inpainting, we first find the closest cluster it belongs to according to its skin tone. We then apply the trained SVM classifiers of that cluster to classify its ethnicity, expression and gender. Note that before we do the classification, we transfer pixels from the mean face of that cluster to cover up the occluded region of the input face. We then get the prediction of ethnicity, expression and gender of the input face image from the SVM classifiers. Finally, we choose the guidance face image as the face image with the best matching score combining the ethnicity, expression and gender matching scores, among all the face images in that cluster.

Depending on the size and location of the occlusion mask, the prediction accuracy of our SVM classifiers may drop. We experience accuracies of about 75% for the examples we show in our works. We also experimented with other classifiers (*e.g.*, random forest classifiers, and non-linear SVM with different kernels) and found that they gave similar or weaker performance. We decided to use linear SVM classifiers for simplicity and better accuracy.



(a) Input face (b) Guidance face (c) RGB (d) Intrinsic

Figure 4.9: Improvement brought by running our optimization on intrinsic image components. (a) Input face; (b) Guidance face; (c) Result obtained by running our optimization on the RGB image. Artifacts appear near the boundary of the occlusion region, and the skin tone of the inpainted region is slightly different from the known region; (d) Improved result obtained by running our approach on intrinsic image components.

4.6.5 Intrinsic Image Decomposition (IID)

Since the advantages of IID in image editing (e.g. inpainting) is desirable in our approach we propose to use decomposed components instead of *rgb* space. Slight inconsistency in skin tone, face geometry or illumination between the input face and the guidance face may result in significant artifacts in the inpainting result. Figure 4.9 shows a failure case of applying our patch-based inpainting directly in the *rgb* image domain. Artifacts appear between the hole and known regions owing to the inconsistent skin tones of the input and guidance face images.

To circumvent this problem, we aim to isolate the effects of variation in skin tone and shading distribution in our optimization. We apply the recent intrinsic image decomposition technique by Zhao et al. [136].

To decompose both the input image I and guidance face image I_{g} into their intrinsic components, namely, reflectance (albedo) and shading images as:

$$\begin{aligned} I &= R_I \cdot S_I, && (4.31) \\ I_{\mathrm{g}} &= R_{I_{\mathrm{g}}} \cdot S_{I_{\mathrm{g}}}, && (4.32) \end{aligned}$$

where R_I, S_I, $R_{I_{\mathrm{g}}}$, $S_{I_{\mathrm{g}}}$ are the reflectance and shading images of the input image I and guidance face image I_{g}, respectively. Based on the decomposition results, we match the skin tone and shading level between the input image and guidance face image by:

$$R'_{I_g}(c) = \frac{M(R_I)}{M(R_{I_g})} R_{I_g},\ c \in \{r, g, b\} \tag{4.33}$$

$$S'_{I_g} = \frac{M(S_I)}{M(S_{I_g})} S_{I_g}, \tag{4.34}$$

where $M(\cdot)$ is the median operator.

We then run our patch-based inpainting approach to repair reflectance image R_I using R'_{I_g} as guidance, and shading image S_I using S'_{I_g} as guidance. We combine the repaired intrinsic images together to produce the inpainting result. Figure 4.10 shows an illustrative example. Essentially, our optimizer minimizes:

$$\mathcal{C}'(\mathbb{H}_r, \mathbb{H}_s) = \mathcal{C}(\mathbb{H}_r) + \mathcal{C}(\mathbb{H}_s), \tag{4.35}$$

where $\mathbb{H}_r$ and $\mathbb{H}_s$ represent different selected patches for repairing the reflectance image R_I and the shading image S_I.

The advantage of applying our approach on intrinsic images separately is that the variation caused by shading will not mingle with the variation caused by reflectance, which is an important consideration since different face images are generally captured under different light conditions and hence exhibit different shading patterns. Our optimizer will be able to find different compatible patches for repairing the reflectance image and the shading image separately. Figure 4.10 shows the improvement for different examples.

4.6.6 Optimization

We solve our MRF optimization problem by Belief propagation (BP). BP is an iterative global optimization algorithm introduced by Pearl [91]. It has become popular in image processing and computer vision, specially on image inpainting due to the attention and estimating the marginal calculations. Underlying our patch-based formulation is an MRF graph, where each patch is represented by a node, and a pair of overlapping patches is connected by an edge. This structure is shown in Figure 4.11a.

At each iteration of the optimization, each node sends a message to its neighboring nodes about the beliefs of the labels (*i.e.*, patches) that the neighboring nodes possess. We specifically apply loopy belief propagation [87, 114] to solve our optimization problem because our graph generally contains loops.

In an MRF graph which is a graphical model of a joint probability distribution and consists of an undirected graph, each node sends message to all neighbor nodes. The message $M_{ij}^{x_k}$ from node i to node j indicates how likely node i believes that node j has corresponding label x_k where label x_k represents k^{th} patch of the face image. After multiple iterations, such conversations likely converge to a consensus that determines the marginal probabilities of all variables where this estimated marginal probability is called belief [97]. The messages are updated locally which means that for an example node, the outgoing

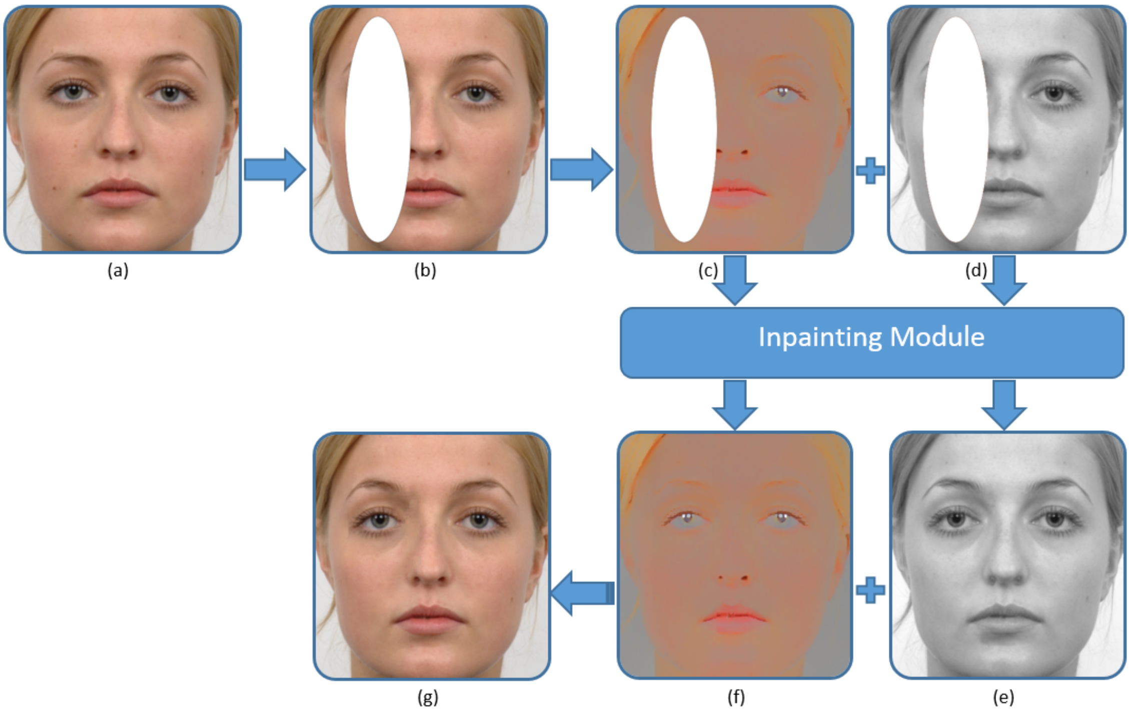


Figure 4.10: Intrinsic Image Decomposition (IID). (a) Ground truth, (b) Occluded sample, (c), (d) Image components where the former is albedo and the second is shading, (e), (f) are inpainted results of both components and finally, (g) is the reconstructed of inpainted albedo and shading layers.

messages is updated based on the incoming messages from previous iterations. This is the main idea behind BP algorithm and messages passing. Figure 4.11b shows a simple MRF for 4 nodes which is as:

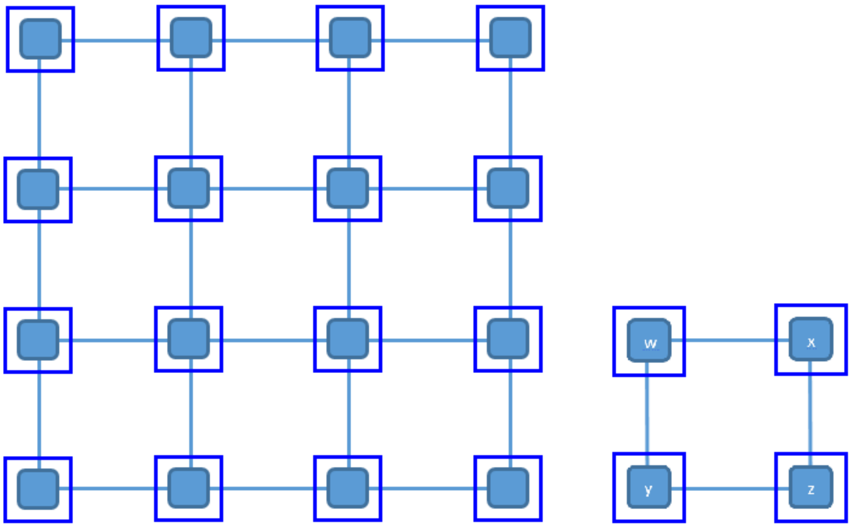
$$P(X) = \frac{1}{Z} \prod_{f_j} f_j(x_j) \tag{4.36}$$

Or particularly:

$$P(w, x, y, z) = \frac{1}{Z} f_{wx}(w, x) f_{xz}(x, z) f_{yz}(y, z) f_{wy}(w, y) \tag{4.37}$$

Where it is a message passing procedure to the node z and estimates the marginal distribution $p(z)$.

BP provides an exact solution if there is no loop among of such graph, however, the result of BP is an approximation of the result if there is at least a loop in the graph. In other words, different status exist for updating the messages while the graphical model is tree, which means there is no loop, an optimal status is by obtaining convergence after computing each message. But for a graph-based graphical status that can be contains loops, such optimal status does not exist and a general solution is updating all messages



(a) Lattice of nodes in a graph for BP algorithm.

(b) A simple MRF with four nodes.

Figure 4.11: Schematic of Lattice of nodes (a) and MRF for image representation with undirected graph.

simultaneously at each iteration [129].

The task of BP in our work is selecting the best patch (node) based on the similarity (marginal distribution) of the neighbors. Particularly, BP computes passing influence of neighbor patches for each specific patch that means each patch receive the idea from neighbors about itself. This structure is very appropriate for inpainting images where we can receive the guessing information of neighbors for each specific patch or pixels. Nevertheless, this information from neighbors can be a direct value or a combination of several mixed knowledge.

4.7 Discussion

We proposed several mapping approaches to perform frontal faces from non-frontal. Linear mapping by general transformation, pose specific transformation and pose specific sparse coding transformation were approaches that we proposed on the beginning of this chapter. We proposed also an extended version of linear mapping as non-linear kernel-based mapping approach that is capable to perform more appropriate transformations. The nature of the transformations are non-linear due to the head pose, face structure, variations, etc. We therefore proposed also an efficient approach based on forests namely mapping forests that provides more accurate non-linear transformations automatically. The last proposed idea is exploiting 3D face model that performs frontalized faces from non-frontal. During the frontalization we may need to inpaint unavailable parts of the faces that they are due to the head pose. We proposed a novel inpainting approach that uses high-level facial attributes. It performs a guidance face that can be used for inpainting unavailable parts. Our evaluations on the performance of the proposed approaches and comparison with the state-of-the-art is provided in the following chapter.

5 Experimental Results

Contents

In this chapter we provide an extensive empirical results on proposed approaches in chapter 4. As proposed in previous chapter we used both raw data and features during our analysis. We show that our idea to map data from non-frontal to frontal can improve the problem of facial expression recognition in arbitrary views and probably for a general class of the problems on non-frontal analysis due to the simplifying the original problem. We organized our experimental results similar to the approaches that we introduced in chapter 4. We first introduce several facial expression datasets that we used during our research from frontal facial expression recognition to multiview recognition. We present then the results of our basic frontal expression recognition. Later, our results from different methods towards multiview facial expression recognition are presented, evaluated and discussed.

5.1 Datasets

A few years ago the lack of suitable facial expressions dataset was a serious gap within facial expression recognition (FER) systems. Two early attempts for collecting such dataset lead to generate Japanese Female Facial Expression (JAFFE) and Cohn-Kanade (CK) datasets. Later, several datasets introduced that support not only the basic problem of

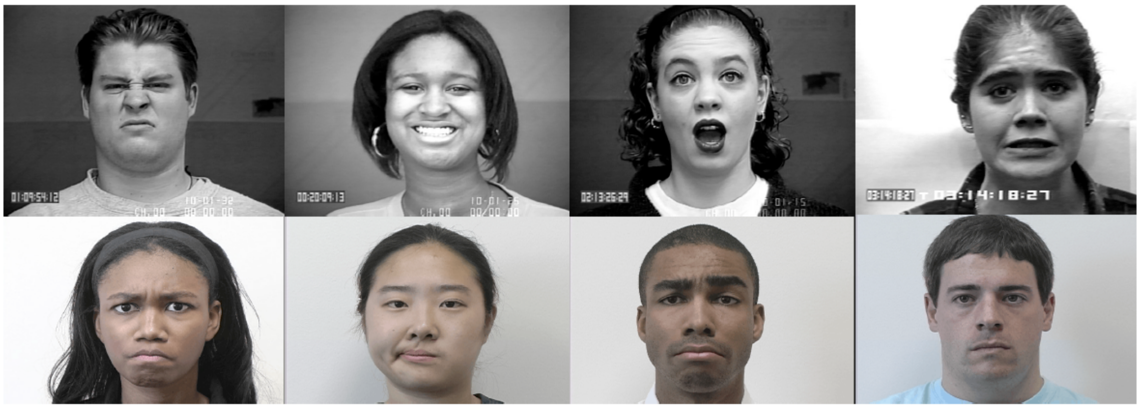


Figure 5.1: Several examples from CK and CK+ datasets. The samples in first row are from CK and the second row is examples related to the CK+

facial expression recognition but also other challenges like influence of illumination, pose, occlusion, continues processing, etc. Some of them are: CMU Multi-PIE, Binghamton University 3D Facial Expression (BU3DFE), Radboud Faces Database (RaFD), Bosphorus, Oulu, AR and MMI. We used some of these datasets in our approaches, therefore, a brief overview and their properties is presented in the following.

5.1.1 Cohn-Kanade (CK) and CK+ Dataset

An early attempts and the most famous facial expression recognition (FER) dataset is CK (CK+ is extended version of CK) that was proposed by Jeffry Cohn and Takeo Kanade in 2000 [57]. Without any doubt they are pioneer on the problem of facial expression recognition and their joint works with Paul Ekman leads to prepare such dataset. CK contains 182 subjects and 2105 sequence from adults in variant ethnicity, age, gender and skin tone. CK dataset extended later in 2010 as CK+ where the number of subjects increased by 27% and the number of sequences increased 22%. The third version of CK is also preparing. CK+ is useful also for the purpose of facial expression tracking or early expression recognition.

The CK+ facial expression database [79] consists of 210 adults with ages from 18 to 50 years; 69% female; 81% Euro-American, 13% Afro-American and 6% other groups. There are seven classes of facial expressions including: Angry (45 subjects), disgust (59 subjects), fear (25 subjects), happy (69 subjects), sadness (28 subjects), surprise (83 subjects) and contempt (18 subjects). All images are frontal with 8 bit color or grayscle and 640x480 or 640x490 pixels resolution. Figure 5.1 shows several example of CK and CK+ faces.

5.1.2 Multi-PIE Dataset

CMU Multi-PIE is a popular multi-purpose dataset in facial analysis. It is supporting variations on pose, illumination, facial expression, etc. containing 337 subjects taken across

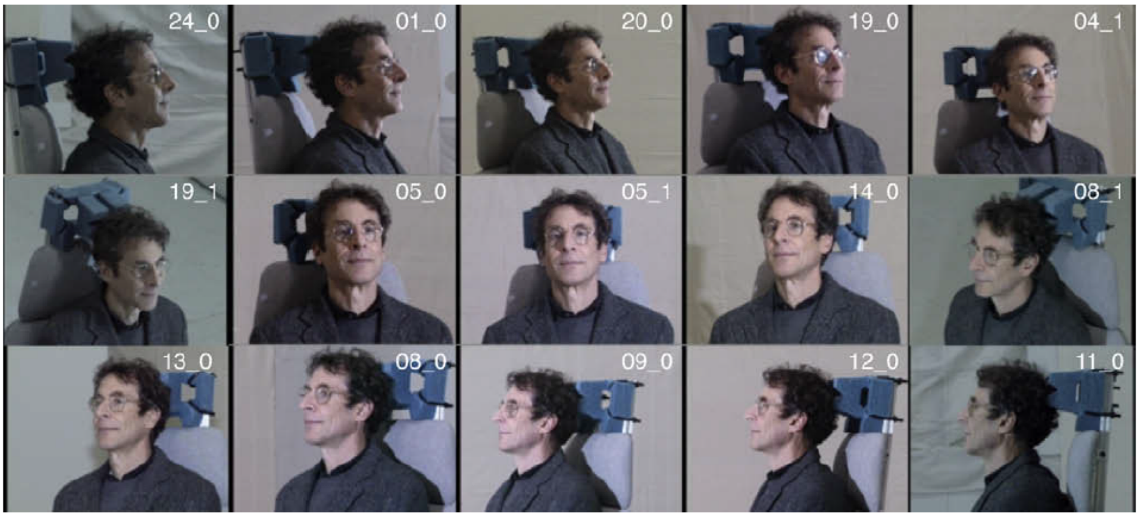


Figure 5.2: Multi-PIE dataset examples in different viewpoints.

15 different viewpoints [35]. Pose variations are between -90° to 90° with an interval of 15° which means there are 13 different viewpoints for subjects and two other cameras are used to simulate a typical surveillance camera view. Multi-PIE images are under 19 illumination conditions which are taken in four recording sessions. Moreover, it contains five facial expressions: disgust (DI), scream (SC), smile (SM), squint (SQ) and surprise (SU) plus neutral (NE). In order to evaluate our approaches on facial expression recognition, we first select all subjects where all of their expressions are available; therefore 145 subjects were selected. Then, we cropped facial regions using a semi-automatic algorithm into the size of 175 × 200 pixels. An example of Multi-PIE faces in different viewpoints illustrated in Figure 5.2 where an example of cropped faces is shown by Figure 5.3.

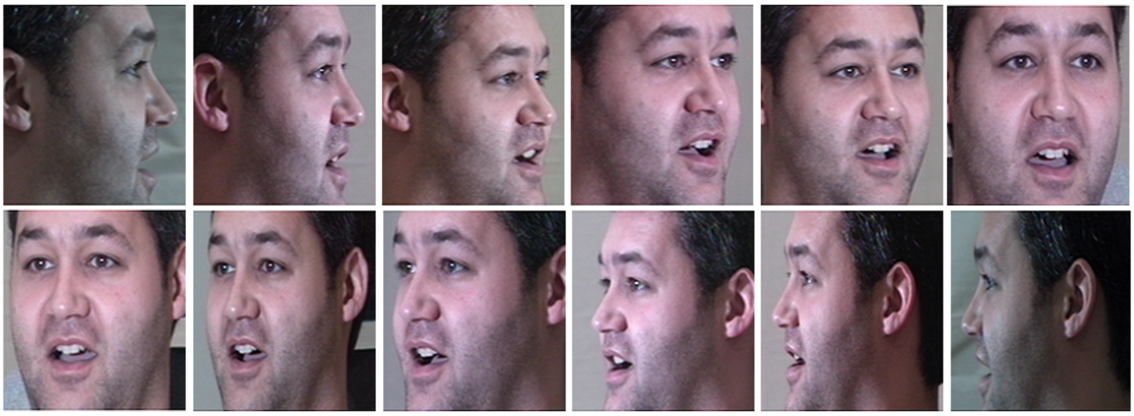
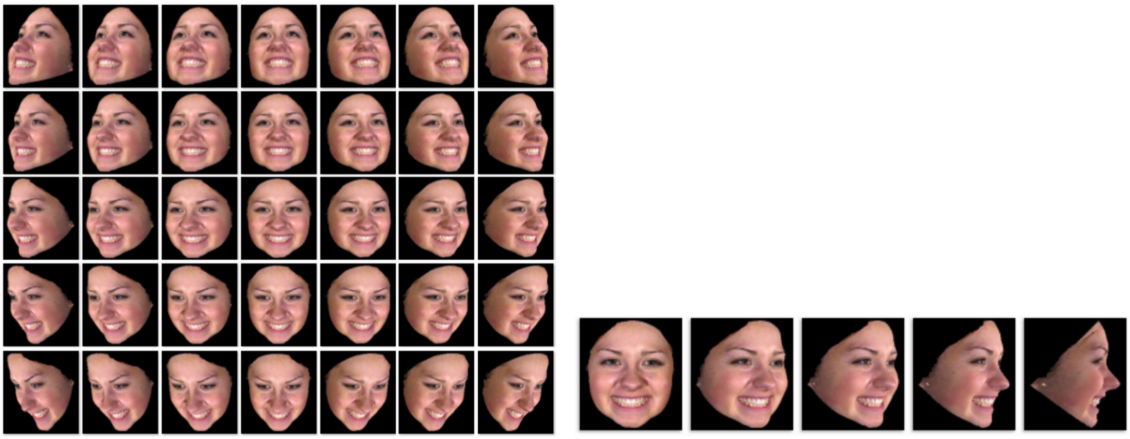


Figure 5.3: Cropped facial examples of a subject in Multi-PIE dataset.



(a) (b)

Figure 5.4: BU3DFE. (a) First protocol, contains 35 viewpoints over pan and tilt, (b) Second protocol, five head poses from frontal to side view.

5.1.3 BU3DFE Dataset

BU3DFE is another popular publicly available dataset containing 3D scanned faces of 100 subjects with six basic expressions, namely anger (AN), disgust (DI), fear (FE), happiness (HA), sadness (SA) and surprise (SU) in 4 levels of expression intensities plus one neutral (NE) which means, there are 25 samples per person and 2500 frontal samples totally [130]. As usually done, we rendered multiple views of the 3D faces from seven pan angles (0°, ±15°, ±30°, ±45°) and five tilt angles (0°, ±15°, ±30°), which gives 35 poses we used to compare our evaluations with the methods that have similar protocol likes [115–118, 142] Figure 5.4a, illustrates rendered faces for a subject sample regarding to the 35 head poses.

In addition we rendered images from five angles: 0°, 30°, 45°, 60° and 90° as a second protocol of BU3DFE to compare our model with studies that applied this protocol such as: [42, 45]. A specific subject in this protocol with variant head pose is demonstrated in Figure 5.4b. In general, as there are 6 expressions for 100 subjects over the highest level of expression intensity in 35 viewpoints, we have 21000 samples for the first protocol. Similarly, we have 3000 samples for the second protocol which is from 6 expressions for 100 subjects over the highest level of expression intensity in five viewpoints.

5.1.4 Radboud Faces Dataset

The Radboud Faces Database (RaFD) is a recently published facial expression dataset proposed by Langner et al. [68] RaFD contains high quality images from 67 subjects including Caucasian male (20 adults), female (19 adults) and children (4 boys and 6 girls) plus 18 Moroccan male adults. They are in eight different expression groups including: anger, disgust, fear, happiness, sadness, surprise, contempt and neutral. Each emotion is shown with three different gaze directions (left, frontal and right) and all pictures were

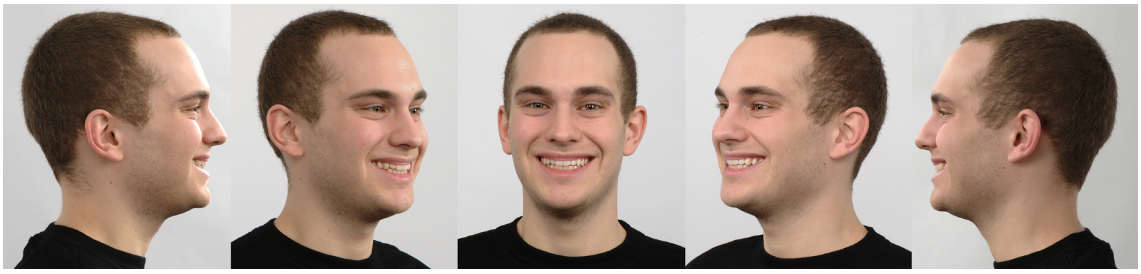


Figure 5.5: An example of RaFD dataset in different viewpoints.

taken from five camera angles simultaneously. Where the viewpoints are (0°, ±45°, ±90°). Therefore, 8040 samples are in this dataset. Figure 5.5 shows a subject example of RaFD dataset in different viewpoints.

5.2 Frontal Facial Expression Recognition

Our multiview facial expression recognition approaches rely on mapping faces (facial features) to frontal. Therefore, an efficient framework for frontal facial expression recognition is desirable. A literature review on the frontal facial expression recognition introduced in 2.4 shows that there are numerous efficient approaches to address this problem. Nevertheless, our contribution on employing local facial components and proposed hierarchical classification 3.2.7.1 improved the FER accuracy. Developing computer vision techniques make this possibility that we can detect and extract facial components. This inspired originally from object detection which is specified on the facial components by learning thousands facial components samples (like the method of face detection [124]). We benefit from such localization due to the influence of facial components on expression recognition. Therefore, we detect the facial components and extract their features for evaluation.

Facial Expressions	Salient facial region(s)
Happiness	Mouth, Eyebrows, Eyes
Surprise	Eyes, Mouth, Eyebrows
Sadness	Eyebrows, Mouth
Disgust	Mouth, Nose, Eyebrows, Eyes
Fear	Eyebrows, Mouth, Eyes
Anger	Eyebrows, Eyes, Mouth

Table 5.1: Face regions affected by facial expressions based on our study of the CK+ and RaFD databases ordered by effectiveness.

Our localization focused on mouth region individually and eyes plus eyebrows with together which are significant for facial expressions. Such detected components is demonstrated in Figure 5.6 while we found by analyzing the FER examples that which com-

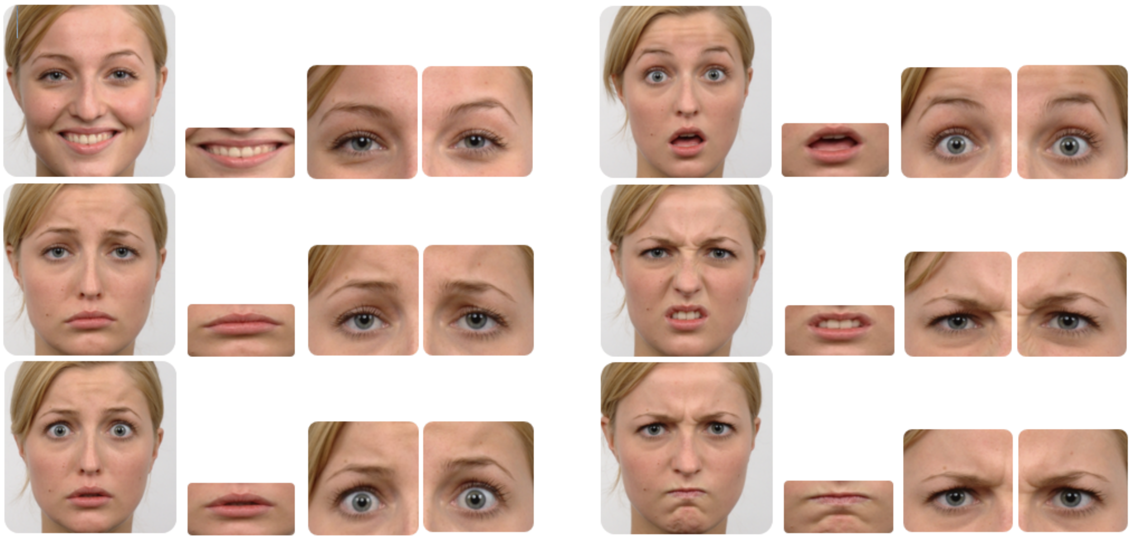


Figure 5.6: Localization by detecting and employing facial components instead of whole of the face.

ponents are related to what expressions and their importance. The result based on the importance is shown in Table 5.1, as can be seen, the difficulty is that these components effectiveness overlapped on the different expressions.

However [62] confirms that nose is beneficial for recognizing 'disgust' but we found in our evaluations that it is a component which increase the ambiguity of recognizing other expressions. Figure 5.7 shows this component in all variant expressions, although, it is distinguishable for disgust and happy but it is almost same for other expressions that can increase the difficulties for discrimination.

5.2.1 Active Regions and Feature Extraction

Active regions are parts of a face that are affected by facial expressions including the facial components. To identify and automatically crop active regions, we employed the Viola & Jones algorithm [124] while Histogram of Oriented Gradient (HOG) features and Local Directional Pattern (LDP) are used to describe the detected active regions.

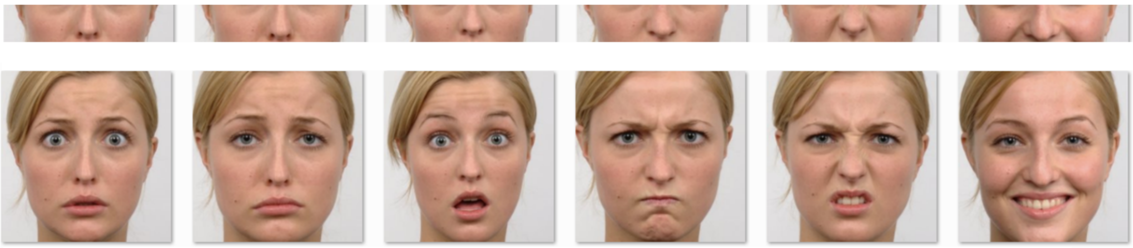


Figure 5.7: Nose, and its sameness on different expressions.

The HOG uses edge information and introduced originally for person detection. HOG descriptors have already been shown to be successful when applied to facial expression recognition and face analysis [18]. Histogram of gradient directions or edge orientations of each part of an image is considered by the HOG and this is the main intention that we used HOG in our model to benefit from gradient information and its advantages. For example, different humans may have different eye color or different skin color but their contours are more stable than color. In other words, contours are more powerful than color for discriminating emotions, especially with eye color or even skin variations.

On the other hand, Local Directional Pattern (LDP) is an appearance-based feature descriptor proposed by Jabid [50]. An LDP feature is obtained by computing the edge response values of an especial filter in 8 directions at each pixel and encoding them into an 8-bit binary number using the relative strength of these edge responses. We considered the concatenation of the HOG and LDP features extracted from whole active regions for discrimination.

5.2.2 Hierarchical Classification for Frontal Faces

Support Vector Machine (SVM) which is well known and popular statistical supervised learning classifier is used for classification. Kernel-based SVM is also popular where the linear, polynomial and radial basis function (RBF) kernels are most commonly used as non-linear SVM classification.

SVM was originally designed for binary classification, but multiclass classification can also be achieved by one-against-all or one-against-one techniques. We used one-against-one linear multiclass SVM. We compute first the multiclass classification rate for all six classes as an initial point then select two highest rates for level one, select then next two highest rates classes for level two and finally other two classes in level three to prepare the hierarchical structure.

The idea behind this hierarchical strategy is classifying simple classes in the first levels to simplify the discrimination of other classes in the next levels by decreasing the data classes.

Figure 5.8 illustrates our classification overview where two highest rate class 1 (SU) and class 2 (HA) are classified in the first level and two next highest rate class 3 (FE) and class 4 (DI) are classified in the second level and the rest two class 5 (AN) and class 6 (SA) are finally discriminated in the last level.

5.2.3 Experimental Results and Comparison

By exploiting active regions and concatenation of HOG and LDP with our hierarchical classification model, we achieved 95.33% and 96.52% recognition rate on CK+ and RaFD databases respectively where we used Leave-One-Subject-Out (LOSO) cross validation. Moreover, our experience with kernel based classification shows no improvement using

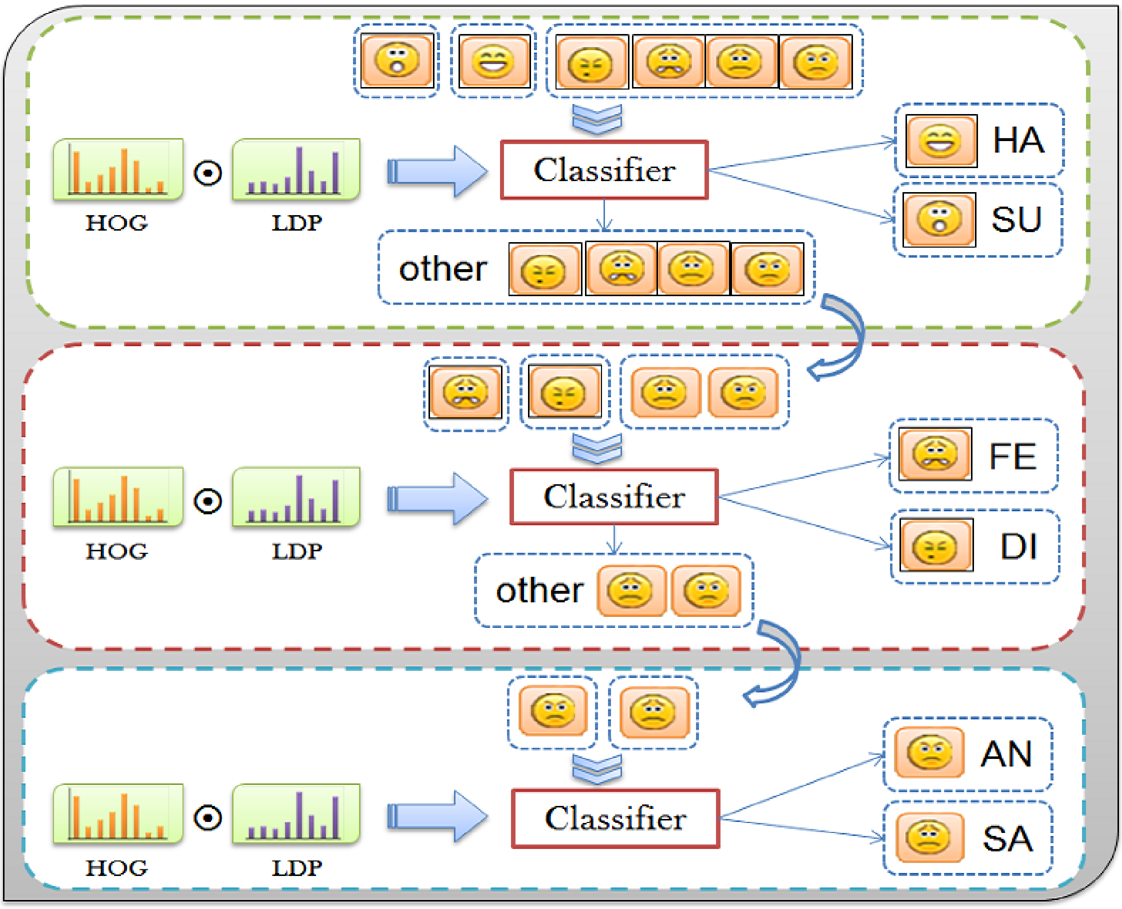
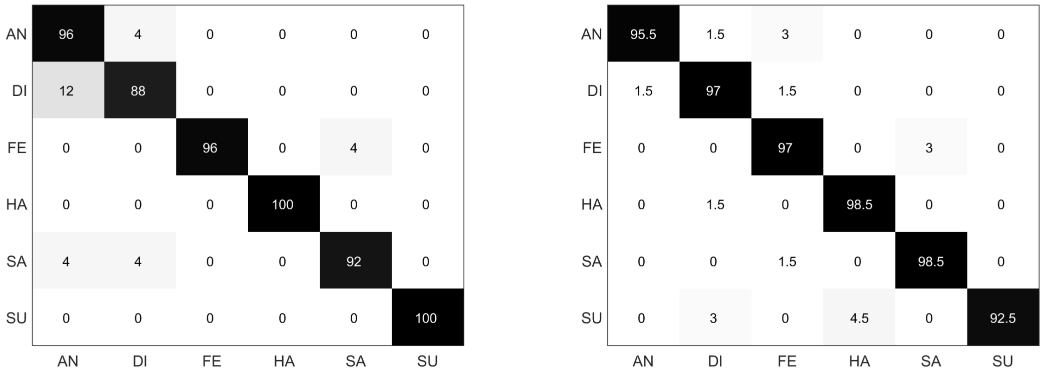


Figure 5.8: Proposed hierarchical classification idea that simplify discrimination by ignoring classified data in the first levels.

polynomial (84.0%) or RBF (84.88%) kernels. Both HOG and LDP are originally gradient-based feature descriptors but the locality of LDP is stronger than HOG and we obtained about 10% improvement when using HOG and LDP together.

In addition, we evaluated other descriptors to investigate the effectiveness of the features. We obtained 90.33% with HOG (alone), 54.66% with SURF, 71% with LBP and 82% with LDP. Moreover, the impact of all face parts is apparent where we achieved the recognition rate 81.33% with only mouth part, 63.55% with eyes and eyebrows, 73.33% with whole face but 95.33% with all of active regions. That obviously confirm the proposed idea of localization by means of facial active regions.

Figure 5.9a and Figure 5.9b show the confusion matrices of our frontal model on CK+ and RaFD databases. The most confusion occurs between expressions anger (AN) and disgust (DI) in CK+ database, due to the expressions having overlaps. On the other hand in both CK+ and RaFD databases, happiness has most recognition rate because all active regions have clear variations.



	AN	DI	FE	HA	SA	SU
AN	96	4	0	0	0	0
DI	12	88	0	0	0	0
FE	0	0	96	0	4	0
HA	0	0	0	100	0	0
SA	4	4	0	0	92	0
SU	0	0	0	0	0	100

(a)

	AN	DI	FE	HA	SA	SU
AN	95.5	1.5	3	0	0	0
DI	1.5	97	1.5	0	0	0
FE	0	0	97	0	3	0
HA	0	1.5	0	98.5	0	0
SA	0	0	1.5	0	98.5	0
SU	0	3	0	4.5	0	92.5

(b)

Figure 5.9: The confusion matrices of proposed hierarchical approach on (a) CK+ and (b) RaFD datasets. Our evaluations are with six expressions similar to the most related works to be comparable with the state-of-the-art.

Moreover, Table 5.2 shows the comparison between proposed approach and the state-of-the-art. All reported approaches use SVM classification and performed their results with Leave-One-Subject-Out (LOSO) cross-validation on CK+. However, other works such as [50, 61, 63] exploited 10-fold cross-validation and different feature descriptors like PLBP, LDP and PHOG that obtained 93.5%, 94.9% and 95.3% recognition accuracy respectively; Also [74] achieved 87.43% recognition rate by means of action units (AUs) with same LOSO cross-validation but Adaboost classifier.

Another work [48] obtained 93.96% accuracy by means of fuzzy inference system (FIS) with assistant of edge of active regions on RaFD database. On the same database, [76] used 33 geometric facial points and employed 10-fold cross-validation with SVM that achieved 84.46% ± 1.07 average recognition rate whereas our accuracy on the same dataset but LOSO cross-validation is significantly gained up to 96.5%. The result of our classification on frontal FER is important because our MFER system relies on mapping non-frontal to frontal and decide then based on the expression recognition on frontalized facial image.

Method	**Features**	**Accuracy**
[54]	Action Units + Texture	86.82%
[34]	Type-II DCT	88.90% ± 2.70
[75]	Facial Feature + Optical Flow	92.50%
[80]	Action Units	92.82%
[78]	Landmark Points	90.40%
Our Hierarchical	HOG + LDP	95.33%

Table 5.2: Comparison with the state-of-the-art appearance-based facial expression recognition on the CK+ dataset using SVM classifier and LOSO cross-validation.

5.3 MFER by means of Linear Regression

By introducing our baseline model on facial expression recognition (FER) in 2.4 we therefore proposed mappings in chapter 4 to transform non-frontal faces to the frontal and use frontalized faces for facial expression recognition in the same manner. The first idea for such mapping is linear regression which introduced in section 4.1. The equation 4.4 provides a general transformation that maps all facial features into the frontal. This idea is included by an approximation between all vectorized facial features in non-frontal views to the correspondence frontal. The approximation can be used then for a new unseen sample in an arbitrary viewpoint to generate an estimation of target features. However, exploring such transformations which can exactly map arbitrary views into the frontal is impossible due to the real world non-linear data and its complexity but our general estimation still performs result better than the baseline in [40, 41].

General transformation encouraged us to perform more accurate mappings. We decreased inherent variations of the features to provide more accurate transformations. Therefore, categorized facial features into the smaller groups by means of the head pose knowledge. We investigated viewpoint-classified features as Pose Specific Classification (PSC) that is an average of all head pose expression recognition. We learned also our model to categorizes first the features in terms of viewpoints then compute the approximation between each cluster and correspondence frontal. Hence, we compute numerous (equal to the head pose classes) local transformations instead of one general transformation. The details of Pose Specific Linear Mapping (PSLM) is proposed in section 4.1.

Given an unseen (non-frontal) input face, we first categorize it into the same cluster in terms of head pose by means of a supervised method as described in 3.2.6.2 then uses the correspondence transformation to map input sample into the frontal. Similar to the general mapping, PSLM is efficient and also outperforms the state-of-the-art on MFER. Figure 5.10 illustrates schematic of both PSC and PSLM approaches where the features are classified in terms of expression in PSC after categorizing into the smaller groups. Instead, PSLM estimates and uses the transformations to approximate a new frontalized face from input test sample, It pass then the frontalized sample for expression recognition based on the frontal analyzable approaches (like our method that analyzed in 5.2).

5.3.1 Sparse-based Linear Mappings

An extension of PSLM can be represented via sparse representation which is introduced in section 4.2.1. Linear Mapping of Sparse Features (LMSF) is by creating a dictionary given the training data such a way that minimizes the reconstruction error as introduced in equation 4.8. Sparse representation can be either general or local where all training data is used for both creating the dictionary and spares feature representation. The sparse features can be used then for general mapping. In contrast, by the local model each cluster is represented into the sparse coding individually. This model uses similar categorizing

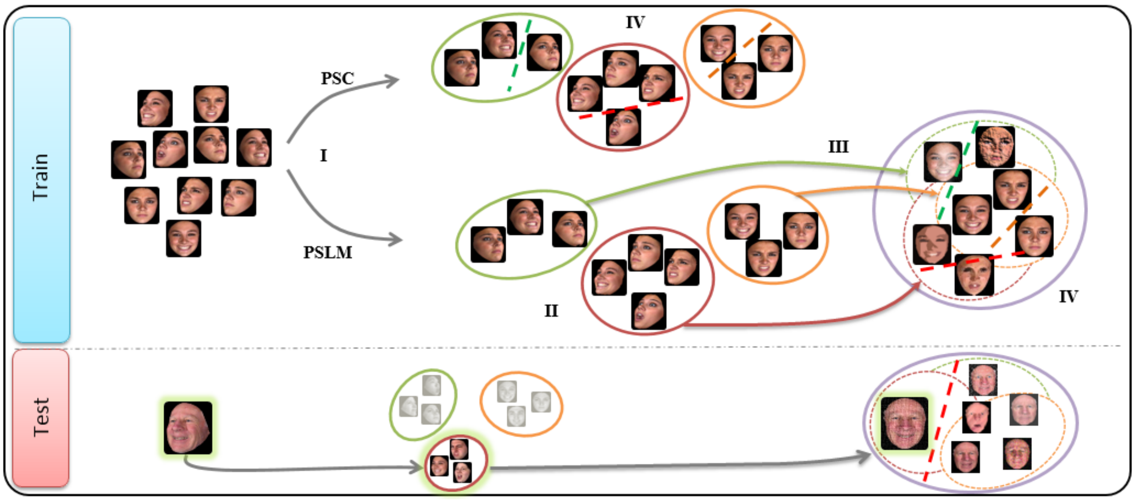


Figure 5.10: PSC and PSLM schematic. PSC categorizes data into the several categories based on the viewpoints. PSLM has the same categorizing process plus estimating transformations that approximate the frontal view of a test sample using learned transformations.

technique to the PSLM for splitting all data into the smaller groups. Therefore, there are numerous clusters that we use for learning our approximations for each specific cluster. We then map a new test sample to the frontal space by means of correspondence approximation while the test sample has before been represented in sparse feature. We show that sparse coding is almost in par with the state-of-the-art but it is very efficient in terms of memory usage. Nevertheless, we proposed an efficient extension of feature representation that improve the basic sparse representation in terms of time complexity in section 4.2.2. Generating the dictionary can be done with K-SVD [1]. It is very expensive in terms of time complexity but fortunately it is an offline task. On the other hand, a new input test sample must be first represented in sparse coding that is an online process. A most popular used approach is OMP [120] that represent an input test sample in sparse feature by means of dictionary which is created before. OMP is an iterative process and computationally expensive. Our alternative for OMP namely Fast Linear Mapping of Sparse Features (FLMSF) which is proposed in 4.2.2, it is very efficient method that not only preserve the OMP accuracy but also improve significantly the running time of the process. We perform our evaluations of all linear mappings in the following.

5.3.2 Experimental Results of Linear Mappings

Our proposed linear mapping approaches for MFER can be separated into the three modules of a pipeline procedure that we follow on the standard evaluation scheme as:

a) Feature extraction: We apply a concatenation of HOG [19] and LBP [89] features. HOG is a gradient-based descriptor and it is stable on illumination variation. Moreover, it is a fast descriptor in comparison to the SIFT and LDP (Local Directional Pattern) due to

Dataset	BU3DFE-P1		BU3DFE-P2		Multi-PIE	
Methods	Accuracy	Time (ms)	Accuracy	Time (ms)	Accuracy	Time (ms)
PSC	77.66%	145	76.63%	48	80.94%	154
PSLM	78.04%	164	77.87%	73	81.96%	169
LMSF	76.04%	360	75.16%	67	74.61%	55
FLMSF	77.61%	71	75.63%	21	75.20%	32

Table 5.3: The accuracy of the MFER approaches on BU3DFE protocol 1 (P1) and protocol 2 (P2) and Multi-PIE protocol 1. The time complexity is based on millisecond (ms) on the same machine.

the simple computation. On the other hand, LBP is a common texture-based descriptor which is used widely in face analysis. It has been shown that a concatenation of HOG and LBP can improve human detection performance by [127]. In our experiments, the extracted features are considered as feature vectors for all facial images.

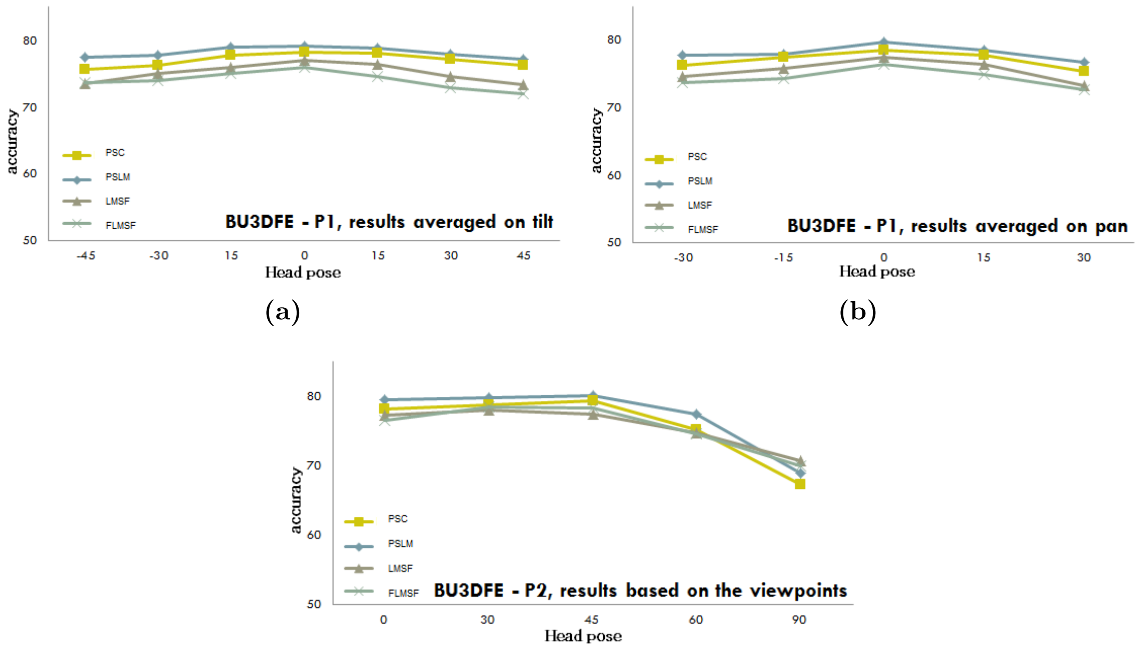
b) Mappings: Linear mappings including PSLM, LMSF and FLMSF estimates transformations from non-frontal to frontal view. It gives us the ability of approximating unavailable features from related available features. Therefore, we employ a pairwise sets of basic features in training data to make mappings for PSLM and a pairwise sets of sparse features to make transformations for LMSF and FLMSF.

c) Classification: All samples are mapped to the frontal, and linear SVM [15] is applied for classifications during all experiments. We consider 5-fold cross validation for both BU3DFE and Multi-PIE datasets where the highest level of expression intensity on BU3DFE is employed in our experiments.

Parameters like dictionary size and sparsity in K-SVD are evaluated, where the best result is achieved by a dictionary size of 200 with sparsity 50 (75% of dictionary elements are used for encoding). The performance of LMSF and FLMSF is very close to each other, although, FLMSF is slightly better than LMSF in accuracy but significantly better in running time. FLMSF is the best method for multiview facial expression recognition concerning time complexity, due to the tremendous reduction of feature dimensionality and the fast ridge regression step, while having better result than state-of-the-art on BU3DFE. Moreover, PSLM results on both datasets show that is the best method concerning the accuracy. A detailed comparison between proposed methods in both accuracy and time complexity is shown in Table 5.3. As can be seen, PSLM has highest accuracy but it is not as fast as FLMSF whereas FLMSF has the best running time and outperforms the state-of-the-art.

For the first protocol, which is BU3DFE-P1 (35 viewpoints), our overall accuracy rate for LMSF, FLMSF and PSLM 76.04%, 77.61% and 78.04% where PSLM and FLMSF are the best on accuracy and running time respectively. FLMSF is about 5 times faster than LMSF and 2 times faster than LMSF in this protocol.

Performing comparison of our approaches over the variations in pan and tilt, illustrated in Figure 5.11a and Figure 5.11b, note that the results (a) are averaged across



(a) (b)

(c)

Figure 5.11: Proposed methods (PSC, PSLM, LMSF and FLMSF) performance on the first protocol of BU3DFE: (a) averaged on pan (b) averaged on tilt and (c) performance on the second protocol of BU3DFE.

corresponding tilt angles and (b) are averaged across corresponding pan angles.

Some related works evaluated their results on the second protocol of BU3DFE. The overall performance of LMSF, FLMSF and PSLM on this protocol are 75.16%, 75.63% and 77.87% respectively. Moreover, the comparison between proposed mapping approaches on this protocol (BU3DFE-P2 including 5 viewpoints) is illustrated in Figure 5.11c (c).

We also evaluated our mapping approaches on Multi-PIE dataset. The overall performances and time complexities of the proposed methods are provided in Table 5.3 where LMSF achieved 74.61% accuracy rate in 55 milliseconds, FLMSF obtained 75.20% accuracy in 32 milliseconds and PSLM in 169 milliseconds with 81.96% accuracy outperforms not only the other proposed approaches but also the state-of-the-art about 5%. Again PSLM and FLMSF are clearly efficient approaches in terms of accuracy and time complexity for multiview facial expression recognition; Figure 5.12 shows the performance of proposed approaches across 13 viewpoints of Multi-PIE dataset.

The above comparisons shows that linear mapping is an efficient and successful approach for MFER where we compare both PSLM and FLMSF approaches with the state-of-the-art in Table 5.4. Moreover, the main important points of PSLM are its applicability, simplicity and high accuracy which are desirable for real applications.

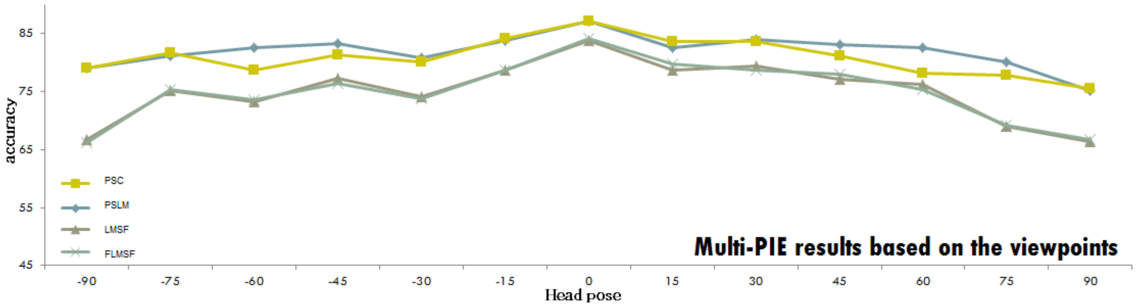


Figure 5.12: Proposed methods (PSC, PSLM, LMSF and FLMSF) performance on the first protocol of Multi-PIE (13 viewpoints).

Approach	Evaluated Dataset	Accuracy
GSCF by Tariq et al. [116]	BU3DFE-P1	76.10%
SSVQ by Tariq et al. [117]	BU3DFE-P1	76.34%
SSE by Tariq et al. [118]	BU3DFE-P1	76.60%
BDA/GMM by Zheng et al. [142]	BU3DFE-P1	68.20%
Our FLMSF	BU3DFE-P1	77.61%
Our PSLM	BU3DFE-P1	78.04%
LBPms by Moore and Bowden [86]	BU3DFE-P2	72.43%
DNPE by Huang et al. [45]	BU3DFE-P2	72.47%
LPP by Hu et al. [40]	BU3DFE-P2	74.46%
LGBP by Moore and Bowden [86]	BU3DFE-P2	77.67%
Our FLMSF	BU3DFE-P2	75.63%
Our PSLM	BU3DFE-P2	77.87%
DNPE by Huang et al. [45]	Multi-PIE-P1	76.83%
Our FLMSF	Multi-PIE-P1	75.20%
Our PSLM	Multi-PIE-P1	81.96%

Table 5.4: Comparison of proposed linear mappings (PSLM and FLMSF) with the state-of-the-art.

5.3.2.1 Linear Mapping Invariance Analysis and Robustness

We investigate our proposed linear mapping approaches with respect to noisy data where three kinds of evaluations are performed: 1) Influence of occlusion, 2) Reducing training data and 3) Head pose analysis. All of these three evaluations are important challenges for which we provide the details and results in the following:

5.3.2.2 Evaluation of Occlusion Presence

In this experiment we have randomly included a white square block in different sizes of 40 × 40, 50 × 50 and 60 × 60 pixel where the original face image is in size of 200 × 220. As our PSLM model is based on the regression transformation and it can virtually perform unavailable/invisible features, the performance is not influenced much; however,

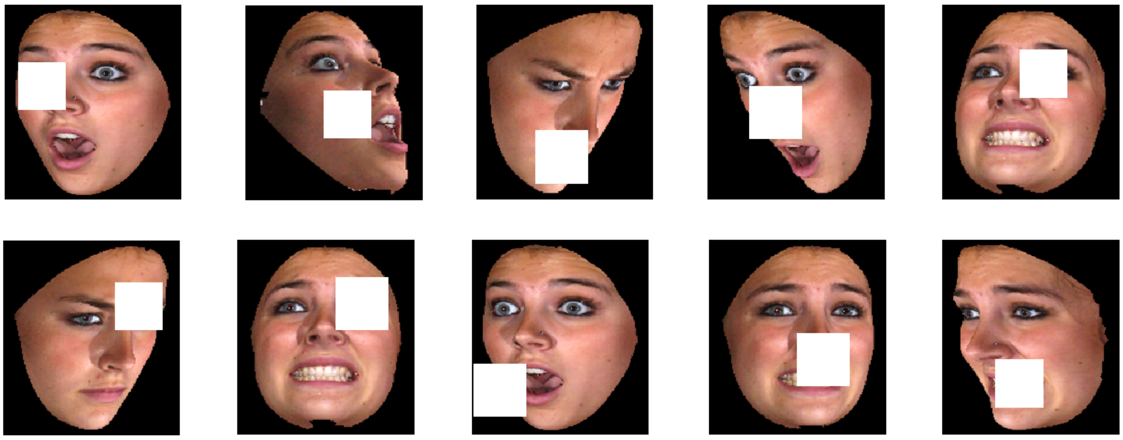


Figure 5.13: Randomly occluded face samples in different viewpoints.

Block size Method	Without occlusion	40 × 40	50 × 50	60 × 60
PSC	77.66%	69.15%	65.30%	61.50%
PSLM	78.04%	71.10%	67.44%	63.32%
LMSF	76.04%	61.65%	56.64%	50.66%
FLMSF	77.61%	61.83%	56.19%	50.46%

Table 5.5: Occlusion evaluation of proposed approaches on BU3DFE-P1, three different block size 40 × 40, 50 × 50 and 60 × 60 is applied for robustness evaluation.

sparse coding based methods cannot handle this amount of noise. Figure 5.13 shows some samples of occluded faces in different size of white block and its random position. Table 5.5 summarizes the results of the proposed approaches on first protocol of BU3DFE (35 viewpoints). All methods decrease slightly, while interestingly the sparse representations are more influenced by occlusion, but PSLM again performs the best.

5.3.2.3 Evaluation on Reducing Training Data

Without any doubt, a large number of training data can increases the complexity and classification time, therefore, similar accuracy using tiny training data is desirable. In this experiment we evaluate our approaches by deleting some viewpoints within training data in the first protocol of BU3DFE, while testing on all 35 viewpoints. This reduces the number of transformations and classifiers available for PSLM and FLMSF, and proves the robustness and generalization capabilities of our proposed ideas. For this purpose we have ignored (a) two columns, (b) two rows and (c) two columns plus two rows of viewpoints in this protocol which means we have ignored 10 viewpoints (i.e. 4800 samples) in task (a), 14 viewpoints (i.e. 6720 samples) in task (b) and finally 20 viewpoints or 57.1% of training data in task (c). Table 5.6 shows the results of our approaches with reducing training data.

Block size Method	Without ignoring	28.5% (10 vp)	40% (14 vp)	57.14% (20 vp)
PSC	77.66%	74.88%	74.68%	72.81%
PSLM	78.04%	76.76%	76.50%	75.10%
LMSF	76.04%	70.33%	70.05%	68.22%
FLMSF	77.61%	73.07%	72.20%	70.85%

Table 5.6: Influence of reducing training data on proposed linear mapping approaches evaluated on BU3DFE-P1.

Method	Ground truth	Correct pose	First neighbor	Second neighbor
PSC	79.52%	77.66%	66.33%	50.48%
PSLM	80.04%	78.04%	77.10%	74.18%
LMSF	78.33%	76.04%	72.94%	72.66%
FLMSF	79.86%	77.61%	74.20%	73.03%

Table 5.7: MFER evaluations with artificially head poses errors on BU3DFE-P1.

As can be seen, in all tasks PSLM is more stable than other methods and if we reduce 40% of training data the expression recognition is 76.50%, which is still better than the state-of-the-art. The missing training data is better compensated by our projections to a common frontal representation than with PSC.

5.3.2.4 Evaluation of Head Poses Estimation Error

As we process features based on the viewpoints, robustness to erroneous viewpoint estimations is critical for robust results. The experimental results in Table 5.3 are generated by an automatic viewpoint classification and therefore already included a small amount of head pose errors. Nevertheless, in this experiment, we artificially create two levels of pose estimation noise, which means during testing we randomly replace each viewpoint estimation by one of its neighboring ones, 15 or 30 degrees farther (see Figure 5.4a), therefore taking wrong classifiers in PSC, and wrong projections and classifiers in PSLM, LMSF and FLMSF. Table 5.7 shows averaged results over 8 runs of selecting wrong neighboring poses. It can be seen that all our regression-based approaches are almost stable with respect to pose errors as expected due to the regression to a common frontal view. The PSC approach decreases as it is trained purely on view specific data.

5.3.3 Conclusion on Linear Mappings

In this section, we introduced three linear regression based mappings: Pose Specific Linear Mapping (PSLM), Linear Mapping of Sparse Features (LMSF) and Fast Linear Mapping of Sparse Features (FLMSF) for multiview facial expression recognition. Our approaches are capable to estimate unavailable/invisible information by using transformations which are learned with regression-based models. We have shown that the proposed PSLM and FLMSF models outperform not only LMSF but also the state-of-the-art approaches. Moreover, FLMSF time complexity is significantly better than other methods and it can be applied in real-time applications due to the running time efficiency. We have also shown that our linear mapping approach (PSLM) is almost stable with small occlusions; reduce training data or severe head poses error. Investigation of non-linear mappings for transforming non-frontal features into the frontal views is an impressive direction that is discussed in 4.3 and its results are proposed by the following section.

5.4 MFER using Non-linear Regression

Exploring pose specific linear mappings and it successful results persuaded us to advert our attention on kernel-based non-linear mappings. It is proposed in section 4.3 where solving the equations 4.18 and 4.19 can perform a kernel-based non-linear mapping on feature space and sparse coding respectively. We proposed to use polynomial kernel as explained in equation 4.17.

Our non-linear mapping approaches including feature space and sparse representation can be also tuned by enhancing our learning approach. Working with smaller subsets as we do has the advantage of dealing with smaller ranges of variation. However, it also reduces the amount of training data. We therefore propose to exploit training data from neighborhoods. Therefore our training data is augmented by means of the samples from similar poses. We consider four neighbors "lower", "upper", "right", and "left" poses. To obtain the neighborhoods, after detecting head pose, we select the neighbors based on our prior knowledge about the head poses and neighborhoods. For instance, we know from BU3DFE dataset that four neighbors of 'Frontal' view are the viewpoints at left with head pose (-15°, 0), right with head pose (15°, 0), above with head pose (0, -15°) and at the bottom with head pose (0, 15°). Therefore, with the similar knowledge we determine the neighborhoods for each detected head pose. Moreover, we do not consider related neighbor if there is no neighborhood. Our experiments show that this strategy slightly improves the overall accuracy.

5.4.1 Experimental Results on Non-linear Mapping

We consider our setup setting as possible as similar to the linear mappings, the cross-validation that is 5-fold on BU3DFE (protocol 1 and 2) and Multi-PIE (protocol 1) plus protocol 2 which is contains seven viewpoints to compare with related works.

We evaluate our proposed non-linear mapping approach in compare with basic Pose Specific Classification (PSC) 4.1 which is without any transformation as baseline; Pose Specific Linear Mapping (PSLM) which is explained in the same section 4.1 that maps faces from each viewpoint to frontal using linear mapping and evaluates them for facial expression; A version of PSLM with Sparse Features is presented as LMSF in section 4.2.1 and the improved version of LMSF in terms of time complexity proposed as Fast LMSF or FLMSF in section 4.2.2; finally the kernel-based non-linear version of PSLM described in section 4.3 as Non-linear Pose Specific Mapping (NPSM).

The performances of LMSF and FLMSF are very close to each other, but FLMSF is faster at run-time: As shown in Table 5.8, FLMSF is much faster than all the other approaches on all protocols for both BU3DFE and Multi-PIE datasets because it does not use OMP, decreases the feature dimensionality from basic features (with dimension equal to 4580) to sparse representation in 200 dimensions, and relies on the fast ridge regression step, while having better results than several related works. Moreover, PSLM

Dataset & Protocols	BU3DFE		Multi-PIE	
Number of viewpoints	P1(35 vp)	P2(5 vp)	P1(13 vp)	P2(7 vp)
Methods				
PSC	50	48	56	53
PSLM	77	71	79	79
LMSF	96	93	90	88
FLMSF	35	38	50	39
NPSM	92	88	82	81

Table 5.8: Mapping approaches time complexity for transforming non-frontal facial faces to the frontal by exploiting linear and non-linear mappings evaluated on the BU3DFE and Multi-PIE datasets. Note that the running time is analyzed on the same machine based on the millisecond (ms) per sample.

Dataset & Protocols	BU3DFE		Multi-PIE	
Number of viewpoints	P1(35 vp)	P2(5 vp)	P1(13 vp)	P2(7 vp)
Methods				
PSC	77.66%	76.36%	80.94%	82.07%
PSLM	78.04%	77.87%	81.96%	82.55%
LMSF	76.04%	75.16%	74.61%	77.04%
FLMSF	77.61%	75.63%	75.20%	76.89%
NPSM	**79.26%**	**78.79%**	**82.43%**	**83.09%**

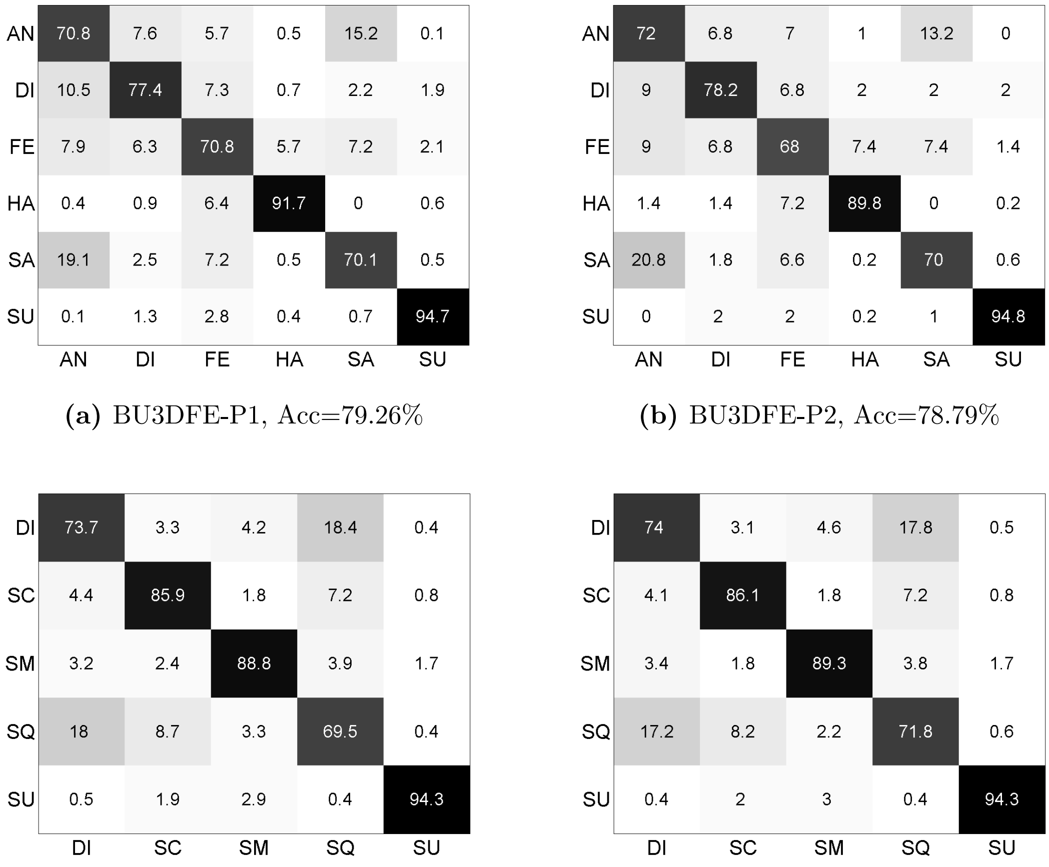
Table 5.9: Mapping approaches accuracy for transforming non-frontal facial faces to the frontal by exploiting linear and non-linear mappings evaluated on the BU3DFE and Multi-PIE datasets.

results on both datasets show that is better than sparse-based methods concerning the accuracy while NPSM which is a kernel-based non-linear version of PSLM outperforms all methods with 79.26% on BU3DFE-P1, 78.79% on BU3DFE-P2, 82.43% on Multi-PIE-P1 and 83.09% on Multi-PIE-P2. A detailed comparison between proposed methods is shown in Table 5.9.

Moreover, Figure 5.14 demonstrates four confusion matrices for the NPSM method. It can be seen that most of the confusion is between sadness and anger on both protocols of BU3DFE and similarly the most confusion on Multi-PIE protocols is between disgust and squint. The best recognized expressions are surprise and then happiness due to the clear variations.

5.4.2 Comparison with the state-of-the-art

A comparison between our approaches with the state-of-the-art on both protocols of BU3DFE and Multi-PIE is evaluated. Table 5.10 shows that NPSM outperforms the state-of-the-art for all protocols of BU3DFE and Multi-PIE. [85] proposed an approach similar to our PSC method introduced in 4.1 but based on a new descriptor (LGBP). They reported 80.17% accuracy on Multi-PIE dataset with seven viewpoints similar to



	AN	DI	FE	HA	SA	SU
AN	70.8	7.6	5.7	0.5	15.2	0.1
DI	10.5	77.4	7.3	0.7	2.2	1.9
FE	7.9	6.3	70.8	5.7	7.2	2.1
HA	0.4	0.9	6.4	91.7	0	0.6
SA	19.1	2.5	7.2	0.5	70.1	0.5
SU	0.1	1.3	2.8	0.4	0.7	94.7

(a) BU3DFE-P1, Acc=79.26%

	AN	DI	FE	HA	SA	SU
AN	72	6.8	7	1	13.2	0
DI	9	78.2	6.8	2	2	2
FE	9	6.8	68	7.4	7.4	1.4
HA	1.4	1.4	7.2	89.8	0	0.2
SA	20.8	1.8	6.6	0.2	70	0.6
SU	0	2	2	0.2	1	94.8

(b) BU3DFE-P2, Acc=78.79%

	DI	SC	SM	SQ	SU
DI	73.7	3.3	4.2	18.4	0.4
SC	4.4	85.9	1.8	7.2	0.8
SM	3.2	2.4	88.8	3.9	1.7
SQ	18	8.7	3.3	69.5	0.4
SU	0.5	1.9	2.9	0.4	94.3

(c) MultiPIE-P1, Acc=82.43%

	DI	SC	SM	SQ	SU
DI	74	3.1	4.6	17.8	0.5
SC	4.1	86.1	1.8	7.2	0.8
SM	3.4	1.8	89.3	3.8	1.7
SQ	17.2	8.2	2.2	71.8	0.6
SU	0.4	2	3	0.4	94.3

(d) MultiPIE-P2, Acc=83.09%

Figure 5.14: Confusion matrices for the non-linear pose specific mapping. (a),(b) the confusions for two protocols of BU3DFE and (c),(d) two protocols of the Multi-PIE dataset.

Multi-PIE-P2 but they used six expressions from 100 subjects. [139] reported 81.7% on the same dataset with his GSRRR method whereas our NPSM reaches 83.09% for the same seven viewpoints. Table 5.10 shows that our non-linear mapping approach is the best technique for MFER.

5.4.3 Analysis of Robustness

We consider three important challenges for practical multiview facial expression recognition similar to the evaluations on linear mappings in section 5.3.2.1: Presence of occlusion, amount of training data, and head pose estimation errors. There is no standard protocols, and we first detail the choices we made before comparing with previous methods.

Methods	Dataset/Protocol	Accuracy
BDA/GMM by [142]	BU3DFE-P1	68.20%
EHMM by [115]	BU3DFE-P1	75.30%
GSCF by [116]	BU3DFE-P1	76.10%
SSVQ by [117]	BU3DFE-P1	76.34%
SSE by [118]	BU3DFE-P1	76.60%
PSLM by [53]	BU3DFE-P1	78.04%
NPSM [ours]	BU3DFE-P1	**79.26%**
LBP^{ms} by [86]	BU3DFE-P2	72.43%
DNPE by [45]	BU3DFE-P2	72.47%
LPP by [42]	BU3DFE-P2 [a]	73.06%
LGBP by [86]	BU3DFE-P2	77.67%
PSLM by [53]	BU3DFE-P2	77.87%
NPSM [ours]	BU3DFE-P2	**78.79%**
DNPE by [45]	Multi-PIE-P1 [b]	76.83%
PSLM by [53]	Multi-PIE-P1	81.96%
NPSM [ours]	Multi-PIE-P1	**82.43%**
PSLM [ours]	Multi-PIE-P2	82.55%
NPSM [ours]	Multi-PIE-P2	**83.09%**

[a] with 4 level of intensities

[b] 100 subjects instead of our protocol with 145 subjects

Table 5.10: Comparing our PSLM and NPSM methods with the state-of-the-art.

5.4.3.1 Evaluation of Occlusion Presence

For this experiment we inserted white square blocks of various sizes (40 × 40, 50 × 50 and 60 × 60, in 200 × 220 face images in BU3DFE-P1) at random places. Figure 5.13 shows examples of occluded faces. Table 5.11 summarizes the results of the proposed approaches with the first protocol of BU3DFE (35 viewpoints) on these perturbed images. All methods decrease slightly, those using sparse representations decrease more, but NPSM still performs the best. Our PSLM and NPSM methods are not much influenced by these artificial occlusions. By contrast, sparse coding-based methods cannot handle this amount of noise.

5.4.3.2 Evaluation of Reducing Training Data

Acquiring labeled data is always cumbersome, and similar accuracy using less training data is desirable. In this experiment we evaluate our approaches by removing some viewpoints from the training data for the first protocol of BU3DFE, while testing on all 35 viewpoints.

For this purpose we have ignored (a) two columns, (b) two rows and (c) two columns plus two rows of viewpoints in this protocol which means we have ignored 10 viewpoints

Occlusion size	Ground Truth	40 × 40	50 × 50	60 × 60
Methods				
PSC	77.66%	69.15%	65.30%	61.50%
PSLM	78.04%	71.10%	67.44%	63.32%
LMSF	76.04%	61.65%	56.64%	50.66%
FLMSF	77.61%	61.83%	56.19%	50.46%
NPSM	**79.26%**	**72.08%**	**67.89%**	**63.76%**

Table 5.11: Evaluation of the influence of occlusions on the accuracy of the proposed approaches with BU3DFE-P1. We considered three different block sizes 40 × 40, 50 × 50 and 60 × 60.

Reduced training	Ground Truth	28.50%	40%	57.14%
Methods				
PSC	77.66%	74.88%	74.68%	72.81%
PSLM	78.04%	76.76%	76.50%	75.10%
LMSF	76.04%	70.33%	70.05%	68.22%
FLMSF	77.61%	73.07%	72.20%	70.85%
NPSM	**79.26%**	**77.10%**	**76.80%**	**75.40%**

Table 5.12: Evaluation of the influence of reducing the amount of training data on the proposed approaches accuracy.

(i.e. 4800 samples) in task (a), 14 viewpoints (i.e. 6720 samples) in task (b) and finally 20 viewpoints or 57.1% of training data in task (c), as shown in Figure 5.15.

The results are illustrated in Table 5.12, which shows that our approaches' robustness on reducing training data. NPSM and PSLM are more stable than the other methods while we reduce 28.5% of training data the expression recognition of NPSM and PSLM are 77.10% and 76.76% respectively, which are still better than the state-of-the-art.

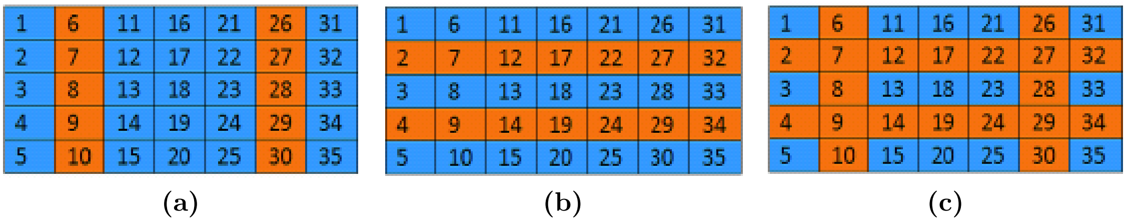


Figure 5.15: Ignoring some training data to test the need for large training sets. (a) Ignoring two columns of training data: viewpoints 6-10 and 26-30); (b) ignoring two rows of 2 and 4 from training data; (c) ignoring two rows and two columns of training data.

5.4.3.3 Evaluation of Head Poses Estimation Error

As we process features based on the viewpoints, robustness to erroneous viewpoint estimations is critical for robust results. The experimental results in Table 5.9 were obtained

Used viewpoint Methods	Ground Truth	First level of neighborhood	Second level of neighborhood
PSC	77.66%	66.33%	50.48%
PSLM	78.04%	77.10%	74.18%
LMSF	76.04%	72.94%	72.66%
FLMSF	77.61%	74.20%	73.03%
NPSM	**79.26%**	**78.19%**	**76.64%**

Table 5.13: Influence of the error when estimating the head pose on the recognition of the facial expressions for the methods we propose.

by an automatic viewpoint classification and therefore already included a small amount of head pose errors, 10.84% for BU3DFE-P1, 2.43% for BU3DFE-P2 and less than 1% for both protocols of the Multi-PIE dataset.

In this experiment, we artificially add two levels of pose estimation noise: During testing we randomly replace each viewpoint estimation by one of its neighboring ones, 15 or 30 degrees farther, therefore taking wrong classifiers in PSC, wrong transformations and classifiers in PSLM, LMSF, FLMSF and NPSM. Table 5.13 shows averaged results over 8 runs of selecting wrong neighboring poses. It illustrates that all our mapping-based approaches are almost perfectly stable with respect to pose estimation errors. The PSC approach decreased as it is trained purely on view specific data.

5.4.4 Conclusion on Non-linear Mappings

In this section, we proposed non-linear mapping for the problem of MFER. It is more appropriate for approximating transformations between source (non-frontal) and target (frontal) data. Our experimental results show that proposed non-linear pose specific mapping (NPSM) approach not only achieved the best result in comparison with linear mapping but also outperforms the state-of-the-art. Moreover, NPSM shows that is almost stable under head pose estimation errors, partial occlusion and small training datasets.

5.5 Mapping Forests (MF) for MFER

By the previous experimental results we show that the idea of mapping non-frontal facial features to the frontal is not only efficient but also outperformed the state-of-the-art. The intuition of employing non-linear mappings emphasizes that we can propose as accurate as possible transformations including: linear, non-linear, pose specific, etc. Although we showed in section 4.3 and 5.4 that non-linear mappings improve our mappings and the state-of-the-art but exploring such kernel and determining the kernel parameters is itself another problem. Therefore, an idea to perform appropriate mappings is exploiting regression forests. As introduced in section 2.6, decision forests are very efficient data structure for such problems where we can perform desirable mapping functions trough the forests. We split data into the smaller subsets in each node of the trees then perform pairwise ridge regression for each subset of data at leaf nodes to learn the best optimized mapping solution.

There are advantages to exploit forests for our mappings: 1) It is very fast due to the structure of decision trees. 2) Forests can perform very accurate transformations. Therefore, as explained in section 4.4 instead of using non-linear mappings which is limited to the popular kernels we ask from forests to perform appropriate mappings which could be either linear or non-linear.

5.5.1 Training the Trees

Similar to the [24, 29] we learn all trees in mapping forests (MF) independently and define $S_j = \{X_{Fr}, X_{NF}\} \in \mathcal{X} \times \mathcal{Y}$ as a subset of training data for a given node j. The goal is to find splitting function $\psi(X_{NF}, \theta_j)$ (e.g. equation 2.11) at S_j such a way that splits data into two branches that maximizes the information gain. The splitting starts usually from the root through a branch of the tree to the leaves. We define $\theta_j = X_{NF}\{j-k, \dots, j+k\}$ where $k = \sqrt{len(X_{NF}}/2$ which means $2 \times k$ members of feature vectors X_NF decide for splitting.

5.5.2 Features Type

In this part, we employed the basic features for the input face images $\mathcal{I}$ in all three color channels. The features are includes 1) HOG [23] and 2) LBP [89] descriptions with cell size 25 pixels that they are concatenated as a feature vector for each train and test sample. The main motivation to employ these two descriptors is that HOG provides gradient information of the images whereas LBP describes the intensity. Therefore, we are certain to extract the maximum possible information from the faces. The concatenating of these two feature descriptors makes large feature vectors and very large datasets, therefore, it will be very expensive in terms of time complexity for evaluation. To this end, we employed well-known dimension reduction technique PCA which is reviewed in section 3.2.3 for reducing the dimensionality of data. The different datasets that we used, have

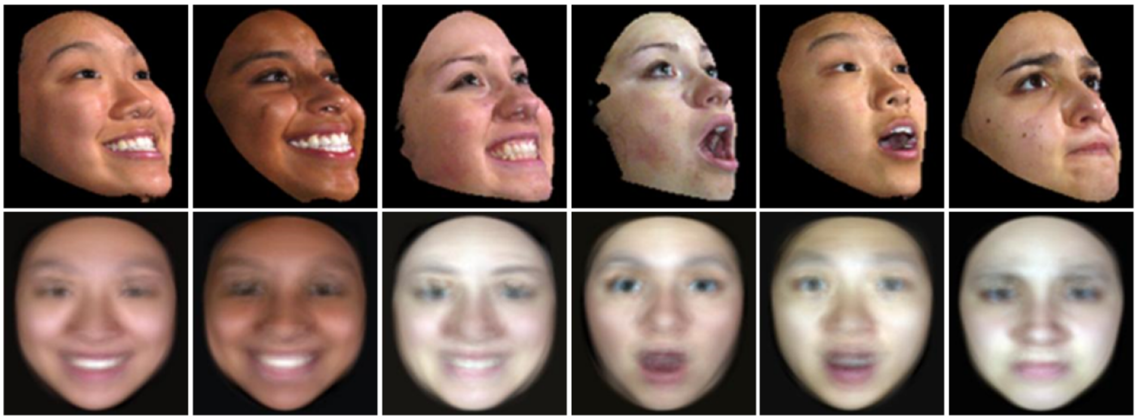


Figure 5.16: Visualization of reconstructed face parts by means of mapping forests. Samples in raw features are transformed to the frontal in presence of different variations.

feature vectors with dimension greater than 14000 (BU3DFE images are in size of 200 × 225 pixels whereas Multi-PIE images size is 175 × 200 pixels). Therefore, we select d=500 during the PCA to reduce the dimensionality of data (both train and test) to a constant number. This means, first 500 principal components are selected for this purpose that makes the similarity of equal or greater than 98% to the original data. For a test sample, we extract and concatenate features as similar as training paradigm; then we represent the extracted features by means of correspondence training data parameters such as the parameters provided by the principal components.

5.5.3 Experimental Results and Setup System

We perform a quantitative and qualitative experiments on both BU3DFE and Multi-PIE. We compare then our results with several approaches and the state-of-the-art. Both datasets are used with two different protocols as introduced in section 5.1 where we used 5-fold cross-validation.

In the train step, we first split all data into the several smaller groups using supervised technique based on the viewpoints like PSC in section 4.1, we then learn mapping models for each group using proposed MF approach.

In the test step, we first approximate the class of the test sample based on the head pose then use the correspondence model provided by MF. The advantage of this pipeline is that each test sample with various head pose which is still not available in our training data will be adapted with the closest subset and therefore, the variation of the head pose will be ignored using MF. Consequently, it can performs the frontal faces which are more simplify for facial analysis also it is almost stable with head pose errors. Figure 5.16, shows reconstructed test samples from non-frontal to frontal provided by proposed MF approach.

Dataset	BU3DFE		Multi-PIE	
Protocol	P1 (35vp)	P2 (5vp)	P1 (13vp)	P2 (7vp)
Random Forests	59.93%	58.10%	68.95%	68.49%
SVM	77.66%	76.36%	80.94%	82.07%

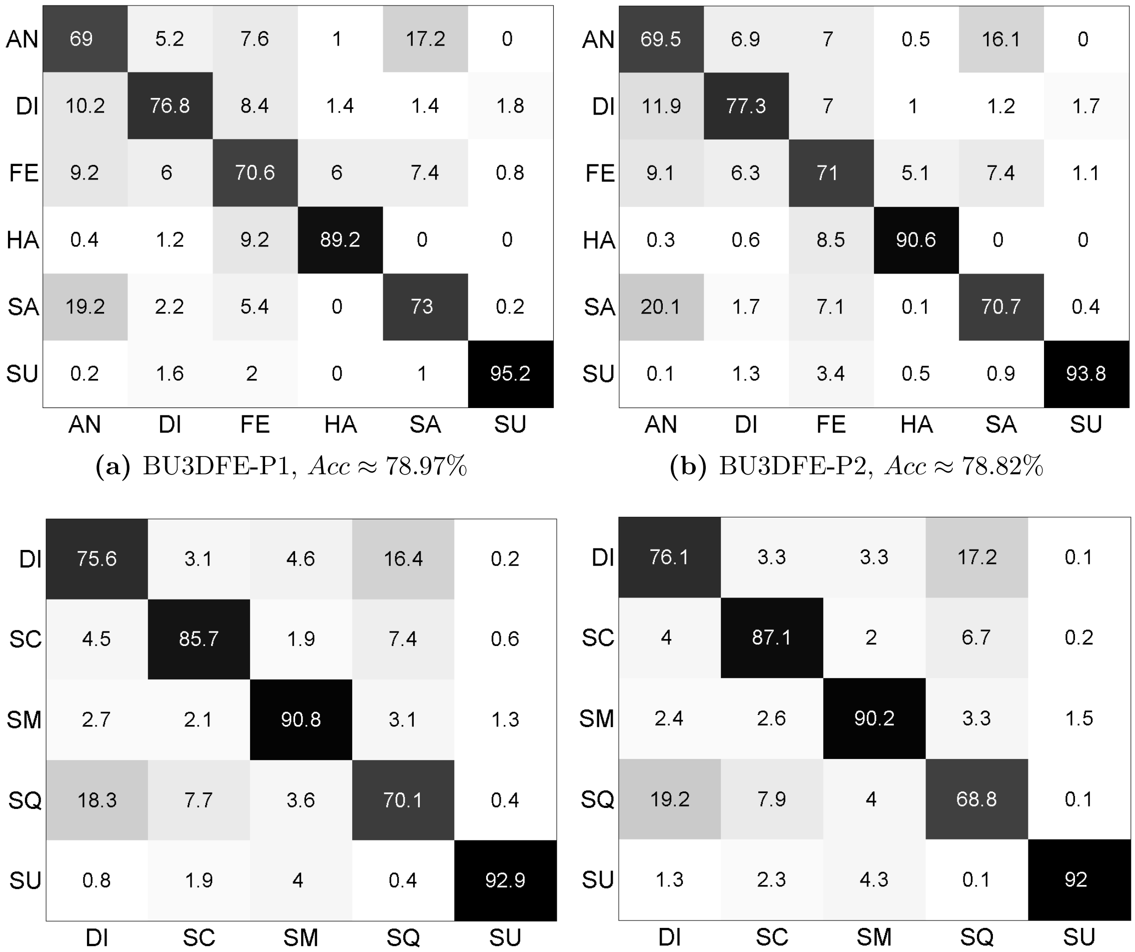
Table 5.14: Baseline pose specific MFER by means of Random Forests and SVM classifiers

5.5.4 Mapping Forests Experimental Results

The results in Table 5.14 is the view-based expression recognition that is evaluated on each specific viewpoint and then averaged through all of them. Our proposed mapping forests approach for the problem of MFER is also evaluated with the following settings: Parameters like number of trees, maximum depth, target features, minimum number of data points for splitting, etc. evaluated and the best results are reported. We have found that 50 is a convenient number of trees and more than 50 just increase the time complexity but there is no significant changes on accuracy. With increasing the depth, the running time will be highly increase and no big improve when we use the depth greater than 4. However, the number of features on splitting each node is very fluctuating and affects on the results. Table 5.15, illustrates MFER results using proposed MF model through the four protocols. Moreover, Figure 5.17 also illustrates their confusion matrices that can be seen the most confusion is between sadness and anger on both protocols of BU3DFE and similarly, between disgust and squint in both protocols of Multi-PIE whereas the best discrimination in both datasets is on surprise and smile due to the clear variations.

5.5.5 Comparison of Mapping Forests and state-of-the-art

In this section, we compare our mapping forests approach with the state-of-the-art on both protocols of BU3DFE and Multi-PIE. Table 5.16 illustrates that MF outperforms the related works in all protocols of BU3DFE and Multi-PIE. However, there are different setup system on facial expression datasets, nevertheless this is clearest comparison with related works and state-of-the-art. In addition, [85] proposed an approach similar to PSC in [53] but that is based on a new descriptor(LGBP), they have reported 80.17% accuracy on Multi-PIE dataset with 7 viewpoints similar to Multi-PIE-P2 but six expressions from 100 subjects, [140] with the same dataset reported 81.7% for his GSRRR method whereas our MF performs 82.84% for 7 same viewpoints on 5 expressions but 145 subjects. Different methods in table 3 show that there are variant approaches to address the problem of MFER, although, our mapping forests approach is one of the best technique for MFER where we can see that our MF approach is straightforward solution with the best results. Moreover, the main important points of proposed idea are its applicability, speed and high accuracy which are desirable for real applications.



	AN	DI	FE	HA	SA	SU
AN	69	5.2	7.6	1	17.2	0
DI	10.2	76.8	8.4	1.4	1.4	1.8
FE	9.2	6	70.6	6	7.4	0.8
HA	0.4	1.2	9.2	89.2	0	0
SA	19.2	2.2	5.4	0	73	0.2
SU	0.2	1.6	2	0	1	95.2

(a) BU3DFE-P1, $Acc \approx 78.97\%$

	AN	DI	FE	HA	SA	SU
AN	69.5	6.9	7	0.5	16.1	0
DI	11.9	77.3	7	1	1.2	1.7
FE	9.1	6.3	71	5.1	7.4	1.1
HA	0.3	0.6	8.5	90.6	0	0
SA	20.1	1.7	7.1	0.1	70.7	0.4
SU	0.1	1.3	3.4	0.5	0.9	93.8

(b) BU3DFE-P2, $Acc \approx 78.82\%$

	DI	SC	SM	SQ	SU
DI	75.6	3.1	4.6	16.4	0.2
SC	4.5	85.7	1.9	7.4	0.6
SM	2.7	2.1	90.8	3.1	1.3
SQ	18.3	7.7	3.6	70.1	0.4
SU	0.8	1.9	4	0.4	92.9

(c) MultiPIE-P1, $Acc \approx 83.02\%$

	DI	SC	SM	SQ	SU
DI	76.1	3.3	3.3	17.2	0.1
SC	4	87.1	2	6.7	0.2
SM	2.4	2.6	90.2	3.3	1.5
SQ	19.2	7.9	4	68.8	0.1
SU	1.3	2.3	4.3	0.1	92

(d) MultiPIE-P2, $Acc \approx 82.84\%$

Figure 5.17: The confusion matrices of mapping forests on two protocols of BU3DFE (a),(b) and (c),(d) confusion matrices for two protocols of Multi-PIE dataset.

Dataset	BU3DFE		Multi-PIE	
Protocol	P1 (35vp)	P2 (5vp)	P1 (13vp)	P2 (7vp)
Random Forests	78.97%	78.82%	83.02%	82.84%

Table 5.15: MFER using proposed mapping forests evaluated on both BU3DFE and Multi-PIE datasets

5.5.6 Discussion on Mapping Forests

There are different solutions to address the problem of multiview facial expression recognition. One of the successful idea to perform such transformation is exploiting regression transformation where it has been used by several works with different improvements. In

Methods	Dataset/Protocol	Accuracy
BDA/GMM [142]	BU3DFE-Prot.1	68.20
EHMM [115]	BU3DFE-Prot.1	75.30
GSCF [116]	BU3DFE-Prot.1	76.10
SSVQ [117]	BU3DFE-Prot.1	76.34
SSE [118]	BU3DFE-Prot.1	76.60
PSR [53]	BU3DFE-Prot.1	78.04
Mapping Forest (MF)	BU3DFE-Prot.1	**78.97**
LBP^{ms} [86]	BU3DFE-Prot.2	72.43
DNPE [45]	BU3DFE-Prot.2	72.47
LPP [40]	BU3DFE-Prot.2 [a]	73.06
LGBP [86]	BU3DFE-Prot.2	77.67
PSR [53]	BU3DFE-Prot.2	77.87
Mapping Forest (MF)	BU3DFE-Prot.2	**78.82**
DNPE [45]	Multi-PIE-Prot.1 [b]	76.83
PSR [53]	Multi-PIE-Prot.1	81.96
Mapping Forest (MF)	Multi-PIE-Prot.1	**83.02**
PSR [53]	Multi-PIE-Prot.2	82.55
Mapping Forest (MF)	Multi-PIE-Prot.2	**82.84**

[a] With 4 level of intensities.

[b] 100 subjects instead of our protocol which uses 145 subjects.

Table 5.16: Multi-view facial expression recognition comparison between state-of-the-art and proposed mapping forests

this section, we mentioned that each part of a non-frontal face can be transform to frontal with an appropriate mapping function. The mapping functions are adapted by mapping forests which make this ability that perform the best continues mapping functions among of thousands transformations. The advantage of such local mappings is while the problem of MFER is a non-linear problem due to the several variations; a simple global mapping cannot provide desirable transformation.

Therefore, a convenient idea is that using appropriate mapping functions for different facial parts. For instance, the variation and effectiveness of mouth is clearly stronger than nose on different expressions. Consequently, proposing specific transformations for each part of the face improve our overall results. On the other hand, finding a desirable transformation between many transformations for each specific part of a face is another difficulty where we solved both problem by proposing mapping forests. MF is able to perform specific transformations for each part of the face and it provides the best mapping functions based on the learning data.

The ability of classifying facial expressions in different head poses is an important property for the facial expression systems. Our MF approach relies on the random regression forests and exploits continuous mapping functions for non-frontal facial features. The exploited mapping s which are learned during the training forests give us this ability to estimate virtually unavailable or partial occluded facial parts. We compared our MF model with the state-of-the-art and showed that MF outperforms the state-of-the-art. Comparison with related works and state-of-the-art in both BU3DFE and Multi-PIE datasets specifically in Table 5.16 demonstrates that our MF model using non-linear mapping can deal with latent variations on the expressions in different head poses and achieves the best accuracy.

5.6 3D based Mapping Models

Investigation of the results on proposed idea of approximating frontal faces shows that is a promising idea and frontalized faces (features) are more useful in terms of facial expression recognition. This is correct for exploiting both raw data and basic features. For instance, in the previous section (5.5) we showed that mapping forests can truly estimate frontalized faces (features) which are useful and efficient for recognizing the expressions. However, the frontalized results are slightly smooth due to the employed PCA. We are therefore motivated to preserve the facial information details to avoid such smoothness. To this end, we use again the idea of mapping non-frontal faces to the frontal like previous approaches, instead, we aim to use 3D information. The idea is inspired from that a non-frontal face is captured from a 3D face model in a specific viewpoint onto the 2D space (weak projection).

The solution named frontalization which is reviewed in section 2.7.2, that employ a 3D face model and map then the input 2D face using a 3D model sample into the frontal. The problem is providing a 3D transformation from 2D image which is ill-posed problem [8], nevertheless, we can approximately discover a transformation between 2D and 3D landmarks (Posit). We can approximate the frontalized faces by exploiting such transformation and prior knowledge about frontal face and its landmarks .

On the other hand, we can also estimate a frontal face by exploiting 3D morphable model and adjust the model on the 2D face image then map our estimated model into the frontal as introduced in section 2.7.1.

We provided the frontalized faces like the samples that illustrated in Figure 5.18 but frontalized samples could be including the regions that they are unavailable due to the head pose. These regions have been compensated with proposed inpainting approach in section 4.6. Figure 5.19 illustrates a face sample with unavailable mouth which is inpainted awkwardly. It is our motivation to exploit high-level facial attributes.

5.6.1 Inpainting Unavailable Face Parts

An overview on our inpainting approach is demonstrated in Figure 5.20. The input is a face image with an occlusion mask, which can be a photo being edited in software or an intermediate result of face frontalization. The goal is to inpaint the occluded region of the face to generate a complete face. Our approach first performs facial analysis on the input face to infer the gender, ethnicity, skin tone and expression. It then uses these metrics to retrieve a guidance face from the face dataset. In addition, the input face image will be decomposed into intrinsic shading and reflectance layers. Our approach then performs patch-based inpainting separately on the shading and reflectance images, using the guidance face, symmetry measures and local cues from the known regions. The inpainted shading and reflectance images are then combined to produce the final, complete face.

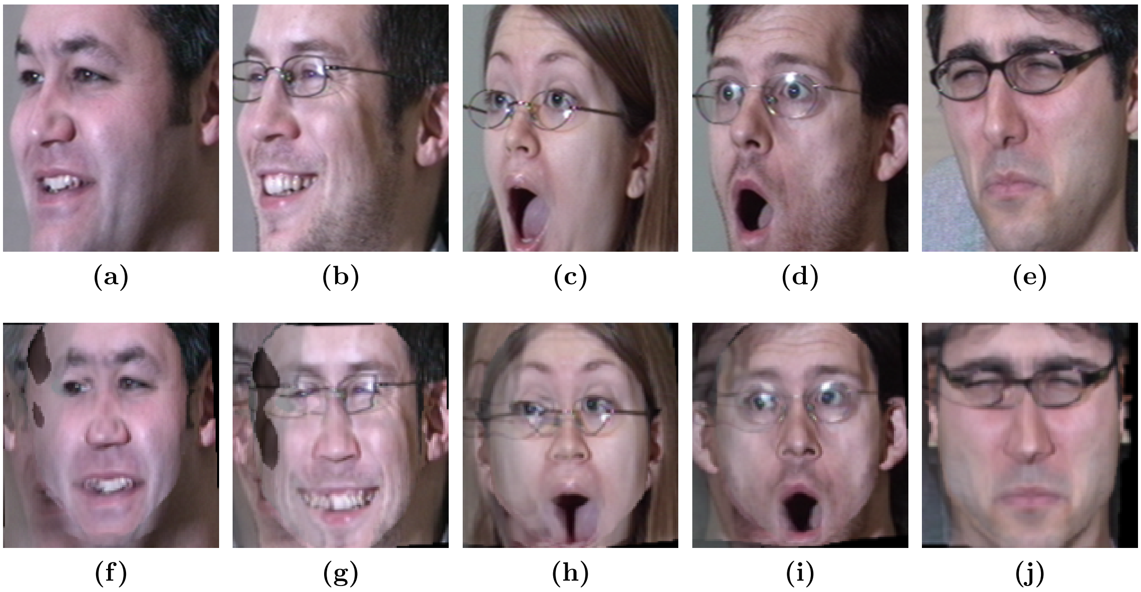


Figure 5.18: Frontalized samples in variant expressions and viewpoints.

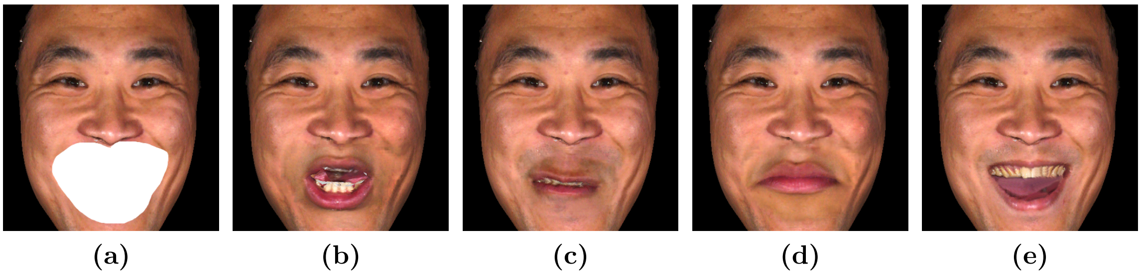


Figure 5.19: Awkwardly inpainting. (a) is an occluded face sample, (b)-(d) are its awkwardly inpainted result and (e) is the ground truth.

5.6.2 Guidance Face Selection

We describe our approach for selecting a guidance face to replenish known patches for repairing. The guidance face patches are particularly important in case the occlusion mask covers a large region of the face (e.g., the entire nose is occluded, or both eyes are occluded) such that no appropriate known patches can be extracted from the input face for repairing. Figure 5.21 shows an illustration of guidance face contribution.

The face images in our face dataset carry ethnicity, expression and gender labels. The skin tone of a face image is computed by running mean shift clustering on the RGB values of the pixels in the forehead and cheek regions, which discards outliers caused by sharp illumination. The value of the cluster's center is taken as the skin tone. We cluster the face images according to their skin tones using k-means clustering. For each cluster, we

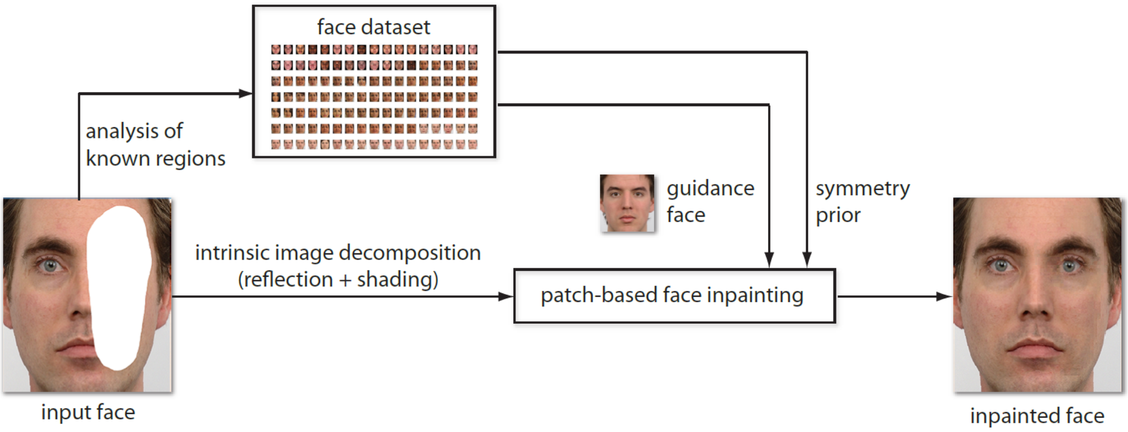

face dataset
analysis of known regions
guidance face
symmetry prior
intrinsic image decomposition (reflection + shading)
patch-based face inpainting
input face
inpainted face

Figure 5.20: Overview of our inpainting approach.

train separate linear SVM classifiers [15] for classifying ethnicity, expression and gender using the face images in the cluster. We use histograms of oriented gradients (HOG) as our features with a cell size of 20×20 pixels. The accuracies of our SVM classifiers for gender, ethnicity and expression are 91.67%, 89.5% and 81.33% respectively, as we cross-validate in training.

Given an input face image for inpainting, we first find the closest cluster it belongs to according to its skin tone. We then apply the trained SVM classifiers of that cluster to classify its ethnicity, expression and gender. Note that before we do the classification, we transfer pixels from the mean face of that cluster to cover up the occluded region of the input face. Where, mean face is an average of 100 aligned faces to represent general structure of the faces. We then get the prediction of ethnicity, expression and gender of the input face image from the SVM classifiers. Finally, we choose the guidance face image as the face image with the best matching score combining the ethnicity, expression and gender matching scores, among all the face images in that cluster.

5.6.3 Optimization

We solve our MRF optimization problem by belief propagation [91]. Underlying our patch-based formulation is an MRF graph, where each patch is represented by a node, and a pair of overlapping patches is connected by an edge. At each iteration of the optimization, each node sends a message to its neighboring nodes about the beliefs of the labels (*i.e.*, patches) that the neighboring nodes possess. We specifically apply loopy belief propagation [87, 114] to solve our optimization problem because our graph generally contains loops.

5.6.4 Experiments on Facial Inpainting

We use the intrinsic image decomposition implementation from Zhao et al. [136] as introduced in section 2.8. For the non-frontal face images used in our examples, we use

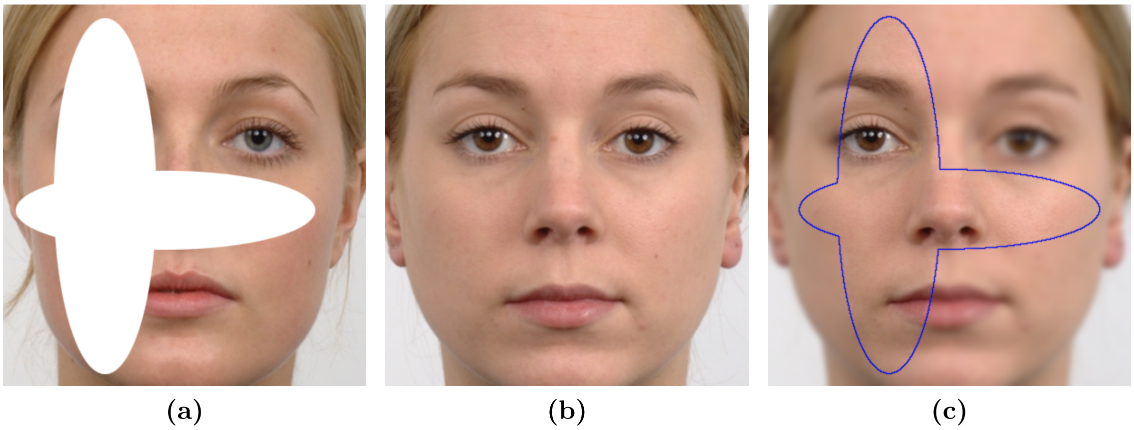


Figure 5.21: Contribution of the guidance face. (a) Input face; (b) Selected guidance face based on skin tone, gender, ethnicity and expression; (c) Candidate region from guidance face for contribution in occluded face.

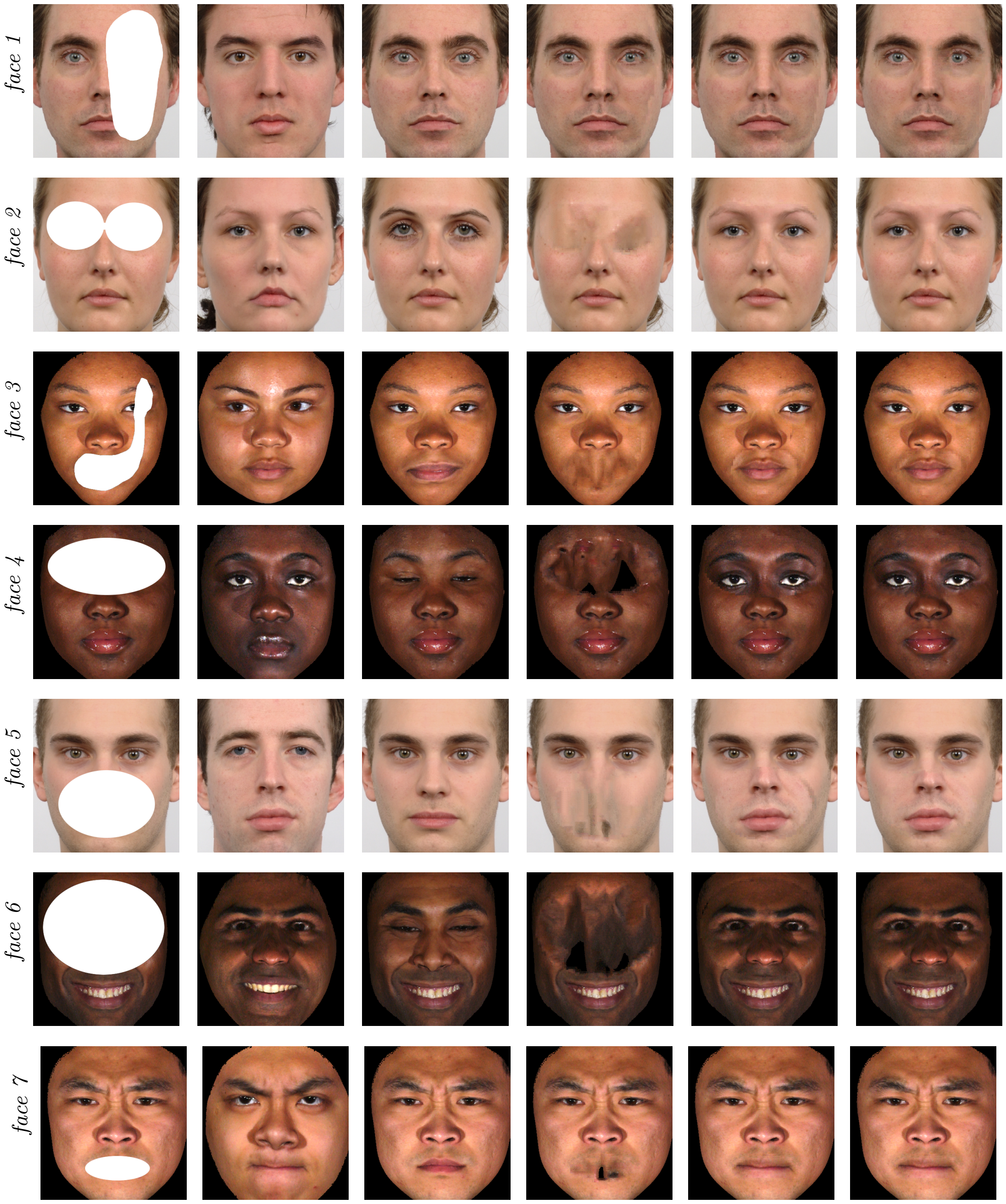
the frontalization implementation from Hassner et al. [37] to frontalize the faces before running our approach. For our SVM classifiers, we use the implementation of LibSVM [15] and the HOG features as proposed by Dalal et al. [19].

5.6.4.1 Concatenation of Data

We use face images obtained from several public datasets: BU3DFE [130], RaFD [68], and MultiPIE [35], which cover both genders and a wide variety of skin tones, ethnicities and expressions. As the original datasets do not carry ethnicity labels, we manually assign an ethnicity label (African, Asian, European or Middle-Eastern) for each face image. There are six categories of expressions: anger, disgust, fear, happy, sadness and surprise. All face images are down-sampled to 420 × 400 pixels, and are aligned based on the positions of eyes and noses. Unless otherwise specified, in all of our experiments we use a patch size of 12 × 12 pixels, which results in about 5800 overlapping patches in a face image. Our face dataset contains 1481 images in total. We use all images for the training components of our approach, except for the images that we show as examples.

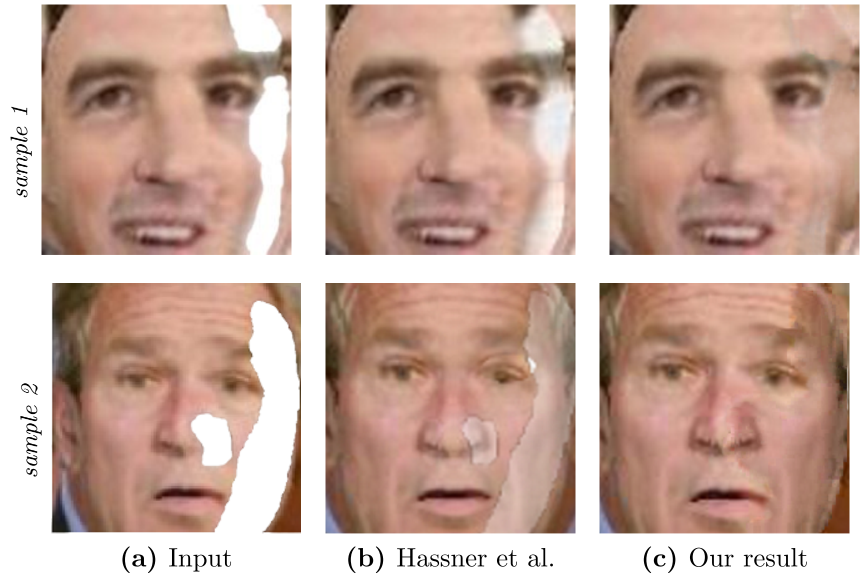
5.6.4.2 Results and Discussion on Facial Inpainting

Different Faces. Figure 5.22 shows the results of running our approach on face images showing different genders and a variety of expressions, ethnicities and skin tones. The occlusion masks we use are considerably larger than those used in previous works [37, 82, 147], with some of them covering about half of the face (e.g., *face 1* & *6*). Our approach is able to inpaint the missing regions of the faces reasonably well. We also show the advantages brought by the use of the guidance face, which enables a reasonable inpainting even if an entire facial component is missing (e.g., both eyes in *face 2*, *4* &



(a) Input face **(b)** Guidance face **(c)** Ground truth **(d)** Result (no GF) **(e)** Result (RGB) **(f)** Result (Intrinsic)

Figure 5.22: Inpainting results. (a) Input face; (b) Guidance face; (c) Ground truth; (d) Result obtained without using the guidance face; (e) Result obtained by running our approach directly on the input image; (f) Result obtained by running our approach on the intrinsic image components followed by combining the outputs to form an RGB image.



(a) Input (b) Hassner et al. (c) Our result

Figure 5.23: Inpainting of frontalized faces.

6; entire mouth in *face 3*, *5* & *7*). Note that just applying general image inpainting techniques [65, 114] with no face prior knowledge would result in awkwardly inpainted faces, such as a face with the entire mouth (e.g., *face 3*, *5* & *7*) or both eyes (e.g., *face 2*, *4* & *6*) missing. We also compare the results obtained by running our approach on the input RGB images directly with those obtained by running our approach on intrinsic images. The latter shows obvious improvements, with the inpainted regions blending more naturally with the known regions.

Another advantage brought by our formulation is that our optimizer can adaptively select a known patch from another region of the input image, or a known patch from the guidance face for repairing a hole patch. For example, in *face 3*, the missing region of the eye on the right is repaired with the eye on the left, while the mouth is repaired with the mouth extracted from the guidance face. By reasoning about high-level attributes such as expression, gender and ethnicity of the input face for selecting a guidance face, our approach can inpaint a face more naturally with high-level consistency. For example, in *face 6*, the inpainted face still shows a smiling face of an African male; in *face 7*, the inpainted face still shows an angry face of an Asian male.

Qualitative Comparison. We also run our approach on frontalized face images which exhibit missing regions that need to be inpainted. Figure 5.23 shows the results. We compare our inpainting results with those of Hassner et al. obtained from running their publicly available code [37]. Our results are comparable to theirs while exhibiting more facial details. Note that our approach can be used to inpaint a more severely-occluded face image (e.g.the frontalized face of a very oblique face) even if an entire facial component is

	Natural	Fake	Indistinguishable
Inpainted Images	47.8%	47.8%	4.4%
Original Images	82.4%	13.7%	3.9%

Table 5.17: Preliminary Perceptual Study Results.

missing, thanks to the use of a guidance face. We believe our face inpainting approach is complementary to their frontalization approach as a useful postprocessing step.

Preliminary Perceptual Study. We also conducted a preliminary perceptual study to evaluate the quality of our results using 20 natural face images from our dataset. The images show faces of different genders, ethnicities and expressions. Ten of the images were occluded and repaired with our approach. We recruited 17 computer science students to participate in our study. The images are shown to each student in random order. Students are asked to label each image as either "natural", "fake" or "indistinguishable".

Table 5.17 shows the results of our study. Among the inpainted images, 47.8% are labeled as "natural", 47.8% as "fake" and 4.4% as "indistinguishable". Among the original images, 82.4% are labeled as "natural", 13.7% as "fake" and 3.9% as "indistinguishable". We believe our approach does a satisfactory job in inpainting, considering that humans are good at detecting even the slightest unnatural appearance in faces [90].

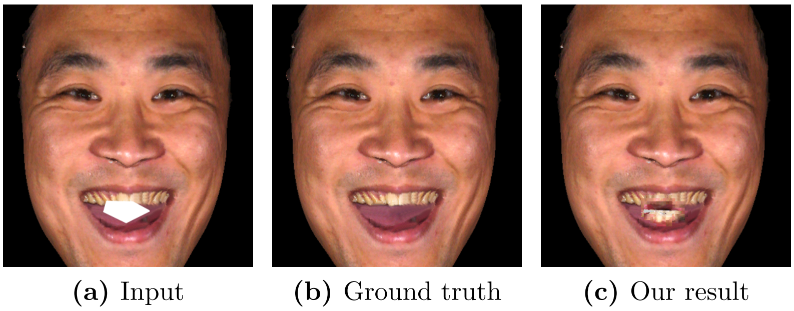

(a) Input (b) Ground truth (c) Our result

Figure 5.24: A failure case. Our approach cannot inpaint the inner region of an open mouth satisfactorily.

Failure Case. Figure 5.24 shows a failure case of our approach. Our approach cannot inpaint the inner region of an open mouth well due to the complexities of the teeth and tongue geometry; neither the input face nor the guidance face provides good patches for repairing. Specific prior knowledge about the inner region of a mouth could be used to improve the inpainting. Our approach also does not handle partially occluded tattoos or hair covering part of a face.

Dataset & Protocols	BU3DFE		Multi-PIE	
Number of viewpoints	P1(35 vp)	P2(5 vp)	P1(13 vp)	P2(7 vp)
Methods				
3D Mapping	82.50%	82.33%	85.67%	85.28%

Table 5.18: Mapping approaches accuracy using frontalized faces evaluated on the BU3DFE and Multi-PIE datasets.

Dataset & Protocols	BU3DFE		Multi-PIE	
Number of viewpoints	P1(35 vp)	P2(5 vp)	P1(13 vp)	P2(7 vp)
Methods				
PSC	77.66%	76.36%	80.94%	82.07%
PSLM	78.04%	77.87%	81.96%	82.55%
LMSF	76.04%	75.16%	74.61%	77.04%
FLMSF	77.61%	75.63%	75.20%	76.89%
NPSM	79.26%	78.79%	82.43%	83.09%
MF	78.97%	78.82%	83.02%	82.84%
3D Mapping	82.50%	82.33%	85.67%	85.28%

Table 5.19: The results and comparison between all proposed mapping approaches

5.6.5 MFER with 3D Mapping Idea

We proposed inpainted frontalized faces in the previous section. Therefore, our MFER problem changed to the frontal facial expression recognition which is simpler than basic MFER. We use frontalized faces for expression recognition as introduced in section 5.2. Table 5.18 is the results of multiview facial expression recognition using 3D mapping idea on the BU3DFE and Multi-PIE datasets. Moreover, we have provided a comparison on all proposed approaches trough this research thesis in Table 5.19 and finally a comparison between our best mapping approaches and the state-of-the-art is illustrated in Table 5.20.

5.7 Discussion

In this chapter we provided our evaluations of proposed mapping approaches. Our proposed approaches are included 1) linear mapping using basic features and sparse features, 2) Non-linear mapping which is an enhanced model of linear mapping using kernel-based transformations, 3) Mapping forests, that is originally a very efficient non-linear mapping model where the mapping functions are selected automatically trough the forests, and finally, 4) Mapping non-frontal faces into the frontal by means of 3D face model and inpainting unavailable/occluded regions.

We have shown that our proposed idea of using mapped faces (features) are successful and can perform better performance than non-frontal faces in terms of recognizing facial expressions. However, we can find in the early study [41] that recognizing expressions

Methods	Dataset/Protocol	Accuracy
BDA/GMM [142]	BU3DFE-Protocol1	68.20%
EHMM [115]	BU3DFE-Protocol1	75.30%
GSCF [116]	BU3DFE-Protocol1	76.10%
SSVQ [117]	BU3DFE-Protocol1	76.34%
SSE [118]	BU3DFE-Protocol1	76.60%
NPSM (ours)	BU3DFE-Protocol1	79.26%
MF (ours)	BU3DFE-Protocol1	78.97%
3D Mapping (ours)	BU3DFE-Protocol1	82.50%
LBP^{ms} [86]	BU3DFE-Protocol2	72.43%
DNPE [45]	BU3DFE-Protocol2	72.47%
LPP [40]	BU3DFE-Protocol2 [a]	73.06%
LGBP [86]	BU3DFE-Protocol2	77.67%
NPSM (ours)	BU3DFE-Protocol2	78.79%
MF (ours)	BU3DFE-Protocol2	78.82%
3D Mapping (ours)	BU3DFE-Protocol2	82.33%
DNPE [45]	Multi-PIE-Protocol1 [b]	76.83%
NPSM (ours)	Multi-PIE-Protocol1	82.43%
MF (ours)	Multi-PIE-Protocol1	83.02%
3D Mapping (ours)	Multi-PIE-Protocol1	85.67%
NPSM (ours)	Multi-PIE-Protocol2	83.09%
MF (ours)	Multi-PIE-Protocol2	82.84%
3D Mapping (ours)	Multi-PIE-Protocol2	85.28%

[a] With 4 level of intensities.

[b] 100 subjects instead of our protocol which uses 145 subjects.

Table 5.20: Multi-view facial expression recognition comparison between our best mapping approaches and state-of-the-art

in non-frontal faces has still better accuracy than the frontal but today there are several approaches that can outperform the baseline performance.

The advantages of proposed approaches in not limited only to the accuracy but also the applicability, simplicity and time complexity are other advantages that made our approaches desirable for real world applications.

We also showed that our mapping approaches robustness on several variations. For instance, reducing the training data; presence of occlusions and head poses error are issues that proposed for evaluating our approaches robustness.

All of our approaches are nearly stable with occlusion. Nevertheless, our last idea that uses inpainting on frontalized faces is capable to compensate unavailable/occluded parts efficiently.

Our evaluations on the extensive results with several datasets and comparison with the state-of-the-art show that our mapping approaches are successful in terms of accuracy, time complexity and robustness.

6 Summary and Conclusion

6.1 Summary and Conclusion

In this thesis we investigated the problem of multiview facial expression recognition. We address two main problems including: First, can we exploit from frontalized faces (features) instead of non-frontal to simplify and boost the expression recognition? Second, what is an ideal transformation to perform frontalized faces (features) for expression recognition purpose? We focused on mapping approaches and performed several mapping models that can estimate frontalized faces (features) which are simplified for investigating on expression recognition.

Firstly, we showed that a linear regression-based mapping can estimate frontalized faces or features from correspondence non-frontal. A successful outcome of this work is that pose specific transformation outperforms always generic mapping. Moreover, it is simple and fast which is desirable for real world applications. Experimental results on this mapping approach showed that it is not only better than the baseline but also outperformed the state-of-the-art. Proposed mapping investigation demonstrated also that it can gain the robustness of the recognition on decreasing training data, mistake head pose estimation and occlusion presence.

Secondly, we proposed a non-linear regression-based mapping approach namely NPSM that can perform more accurate mappings. As naturally the problem of estimating frontal faces or features from non-frontal is non-linear due to the gender, ethnicity, age, expressions, etc. we proposed non-linear mappings instead of linear. Evaluations on the experimental results proved our claim and showed improvements where the frontalized features are more similar to the frontal features. Again the same robustness investigation preserved previous findings and showed that non-linear mapping is nearly stable on presence of occlusion, reducing training data and erroneous head pose estimation too.

Thirdly, we found that non-linear mapping can successfully improve our method but exploiting an appropriate kernel is as difficult as finding the mapping model. Therefore, we proposed to use mapping forests which is an improved idea inspired from the random forests. MF is capable to determine automatically non-linear mappings based on the training data trough the forests. It is very fast and accurate to determine non-linear transformations due to the employed structure.

Lastly, a 3D-based mapping approach is introduced that estimated frontalized faces by exploiting the correspondence mapping function. To meet such mapping function, a 3D reference face model and correspondence frontal image is used as a model. The mapping function is achieved by approximating the detected landmarks trough both of 3D reference model and correspondence frontal image. The frontalized face has some unavailable regions due to the head pose. We proposed inpainting approach using high-level facial attributes to compensate that regions. Both qualitative and quantitative evaluations showed that the frontalized faces seems successfully as natural images and recognizing the expressions has significant improvement than the previous approaches and the state-of-the-art on two most popular BU3DFE and Multi-PIE datasets.

By investigating proposed mapping approaches, we confirm that frontalized faces are more useful than non-frontal faces in terms of facial expression recognition. Therefore, for providing an efficient multiview facial expression recognition system we suggested to map non-frontal data by means of one of the proposed approaches and recognize then the expressions.

Although, the proposed approaches are better than the state-of-the-art but their results are still far to say that the problem of MFER is solved. Therefore, MFER is still an open problem where increasing the discriminants between the expressions or reducing the confusions could be a direction for upcoming works.

B List of Publications

My work at the Institute for Computer Graphics and Vision led to the following peer-reviewed publications. For the sake of completeness of this Thesis, they are listed in chronological order along with the respective abstracts.

B.1 2015

Multi-view Facial Expressions Recognition using Local Linear Regression of Sparse Codes

Mahdi Jampour, Thomas Mauthner and Horst Bischof

In: *Proceedings of Computer Vision Winter Workshop (CVWW)*

February 2015, Schloss Seggau, Styria, Austria
(Accepted for oral presentation)

Abstract: We introduce a linear regression-based projection for multi-view facial expressions recognition (MFER) based on sparse features. While facial expression recognition (FER) approaches have become popular in frontal or near to frontal views, few papers demonstrate their results on arbitrary views of facial expressions. Our model relies on a new method for multi-view facial expression recognition, where we encode appearance-based facial features using sparse codes and learn projections from non-frontal to frontal views using linear regression projection. We then reconstruct facial features from the projected sparse codes using a common global dictionary. Finally, the reconstructed features are used for facial expression recognition. Our regression of sparse codes approach outperforms the state-of-the-art results on both protocols of BU3DFE dataset.

B.2 2015

Pairwise Linear Regression: An Efficient and Fast Multi-view Facial Expression Recognition

Mahdi Jampour, Thomas Mauthner and Horst Bischof

In: *Proceedings of International Conference on Automatic Face and Gesture Recognition (FG)*

May 2015, Ljubljana, Slovenia
(Accepted for oral presentation)

Abstract: Multi-view facial expression recognition (MFER) is an active research topic in facial analysis. In fact, not only the accuracy but also time complexity is desirable for real applications. In this paper, we introduce a new fast and robust approach for recognizing facial expressions in arbitrary views. Our approach relies on learning linear regressions between pairs of non-frontal and frontal sets to virtually compensate occluded facial parts. First, we learn linear regression for projecting from non-frontal to frontal views. Such approximated frontal training features are applied for training view specific facial expression classifiers. We propose a number of different variants of our approach, including sparse encoding and ridge-regression for feature representation. While classical pose specific methods strongly depend on the quality of the pose estimation step, our approaches maintain their superior behavior even under severe pose noise. We evaluate on both BU3DFE and Multi-PIE datasets and outperform the state-of-the-art in classification accuracy, even with a simple pose specific baseline method, while being extremely robust to feature noise and erroneous viewpoint estimation with our pairwise regression approaches.

B.3 2015

Mapping Forests: A Mapping Model for Multiview Facial Expression Recognition

Mahdi Jampour, Thomas Mauthner, Samuel Schulter and Horst Bischof

Submitted for: *journal of Pattern Recognition Letters*

July 2015
(under review)

Abstract: We introduce a new robust approach for head pose invariant facial expressions recognition which is based on Mapping Forests (MF). MF relies on learning non-linear mappings deduced from pairs of source (non-frontal) and target (frontal) training data. It improves the performance of mappings due to providing non-linear transformations by means of random regression forests. In contrast to the related approaches, we employ regression forests to learn mapping models which are naturally non-linear. Therefore, MF can provide more accurate nonlinear mappings to compensate non-available or partial occluded facial features generously. In the experiments, we demonstrate that proposed MF approach is not only in par or better than linear mapping approaches and the state-of-the-art but also very time efficient which is appropriate for real-time applications. We investigate the efficiency and performance of our MF approach on two BU3DFE and Multi-PIE datasets.

B.4 2015

Pose-Specific Non-Linear Mappings in Feature Space towards Multiview Facial Expression Recognition

Mahdi Jampour, Vincent Lepetit, Thomas Mauthner and Horst Bischof

Invited paper for: *journal of Image and Vision Computing*

September 2015
(under review)

Abstract: We introduce a novel approach to recognizing facial expressions over a large range of head poses. Like previous approaches, we map the features extracted from the input image to the corresponding features of the face with the same facial expression but seen in a frontal view. This allows us to collect all training data into a common referential and therefore benefit from more data to learn to recognize the expressions. However, by contrast with such previous work, our mapping depends on the pose of the input image: We first estimate the pose of the head in the input image, and then apply the mapping specifically learned for this pose. The features after mapping are therefore much more reliable for recognition purposes. In addition, we introduce a non-linear form for the mapping of the features, and we show it is robust to occasional mistakes made by the pose estimation stage. We evaluate our approach with extensive experiments on two protocols of the BU3DFE and Multi-PIE datasets, and show that it outperforms the state-of-the-art on both datasets.

B.5 2016

Face Inpainting based on High-Level Facial Attributes

Mahdi Jampour, Chen Li, Kun Zhou, Stephen Lin, Lap-Fai Yu and Horst Bischof

Submitted in: *IEEE Conference on Computer Vision and Pattern Recognition*

Jun 2016, Las Vegas, USA
(under review)

Abstract: We introduce a novel data-driven approach for face inpainting, which makes use of the observable region of an occluded face as well as its inferred high-level facial attributes, namely gender, ethnicity, and expression. Based on the idea that the realism of a face inpainting result depends significantly on its overall consistency with respect to these high-level attributes, our method selects a guidance face that matches the targeted attributes and utilizes it together with the observable input face regions to inpaint the missing areas. These two sources of information are balanced using an adaptive optimization, and the inpainting is performed on the intrinsic image layers of the face to enhance the resulting visual quality. Our experiments demonstrate this approach to be effective even for inpainting entire facial components such as the mouth. By accounting for high-level facial attributes, our method generates more natural facial appearances as determined in a perceptual study.

Bibliography

[1] Aharon, M., Elad, M., and Bruckstein, A. (2006). K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transaction on Signal Processing*, 54:4311–4322. (page 41, 51, 79)

[2] Ahmed, F. (2012). Gradient directional pattern: A robust feature descriptor for facial expression recognition. *Electronics Letters*, 48(19):1203–1204. (page 3, 12)

[3] Almaev, T. and Valstar, M. (2013). Local gabor binary patterns from three orthogonal planes for automatic facial expression recognition. In *Affective Computing and Intelligent Interaction (ACII), 2013 Humaine Association Conference on*, pages 356–361. (page 12)

[4] Azazi, A., Lutfi, S., and Venkat, I. (2014). Analysis and evaluation of surf descriptors for automatic 3d facial expression recognition using different classifiers. In *Information and Communication Technologies (WICT), 2014 Fourth World Congress on*, pages 23–28. (page 12)

[5] Barron, J. and Malik, J. (2013). Shape, illumination, and reflectance from shading. Technical Report UCB/EECS-2013-117, EECS Department, University of California, Berkeley. (page 32)

[6] Bartlett, M., Littlewort, G., Frank, M., Lainscsek, C., Fasel, I., and Movellan, J. (2005). Recognizing facial expression: machine learning and application to spontaneous behavior. In *CVPR*, pages 568–573. (page 12)

[7] Bengio, Y., Courville, A., and Vincent, P. (2013). Representation learning: A review and new perspectives. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(8):1798–1828. (page 40)

[8] Bertero, M., Poggio, T., and Torre, V. (1988). Ill-posed problems in early vision. *Proceedings of the IEEE*, 76(8):869–889. (page 98)

[9] Blanz, V. and Vetter, T. (1999). A morphable model for the synthesis of 3d faces. In *SIGGRAPH*, pages 187–194. (page 28)

[10] Blanz, V. and Vetter, T. (2003). Face recognition based on fitting a 3d morphable model. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(9):1063–1074. (page 13)

[11] Burgos-Artizzu, X. P., Perona, P., and P., D. (2013). Robust face landmark estimation under occlusion. In *ICCV*, pages 1–8. (page 61)

[12] Chakrabarti, D. and Dutta, D. (2014). Facial expression recognition using pca and various distance classifiers. In Sengupta, S., Das, K., and Khan, G., editors, *Emerging Trends in Computing and Communication*, volume 298, pages 79–85. (page 13)

[13] Chen, J., Chen, D., Gong, Y., Yu, M., Zhang, K., and Wang, L. (2012). Facial expression recognition using geometric and appearance features. In *Proceedings of the 4th International Conference on Internet Multimedia Computing and Service*, pages 29–33. (page 15)

[14] Chen, Q. and Koltun, V. (2013). A simple model for intrinsic image decomposition with depth cues. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 241–248. (page 32)

[15] Chih-Chung, C. and Chih-Jen, L. (2011). Libsvm: A library for support vector machines. *ACM T. Intell. Syst. Technol.*, 2. (page 44, 45, 63, 80, 100, 101)

[16] Cohn, J., Ambadar, Z., and Ekman, P. (2007). *Observer-based measurement of facial expression with the Facial Action Coding System*. Oxford University Press. (page 9)

[17] Criminisi, A., Shotton, J., and Konukoglu, E. (2012). Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning. *Foundations and Trends in Computer Graphics and Vision*, pages 81–227. (page 24, 26)

[18] Dahmane, M. and Meunier, J. (2011). Emotion recognition using dynamic grid-based HoG features. In *Automatic Face & Gesture Recognition*. (page 75)

[19] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR*. (page 5, 38, 79, 101)

[20] Danelakis, A., Theoharis, T., and Pratikakis, I. (2014). A survey on facial expression recognition in 3d video sequences. *Multimed. Tools Appl.* (page 13)

[21] De la Torre, F., Chu, W.-S., Xiong, X., Vicente, F., Ding, X., and Cohn, J. (2015). Intraface. In *FG*, pages 1–8. (page 11)

[22] Ding, L., Ding, X., and Fang, C. (2014). 3d face sparse reconstruction based on local linear fitting. *The Visual Computer*, 30(2):189–200. (page 21)

[23] Dollar, P., Tu, Z., Perona, P., and Belongie, S. (2009). Integral channel features. In *BMVC*, pages 1–11. (page 92)

[24] Dollar, P. and Zitnick, C. (2013). Structured forests for fast edge detection. In *ICCV*, pages 1841–1848. (page 23, 92)

[25] Duan, F., Huang, D., Tian, Y., Lu, K., Wu, Z., and Zhou, M. (2015). 3d face reconstruction from skull by regression modeling in shape parameter spaces. *Neurocomputing*, 151, Part 2:674 – 682. (page 21)

[26] Ekman, P. and Friesen, W. (1978). *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press. (page 9)

[27] Fan, X. and Tjahjadi, T. (2015). A spatial-temporal framework based on histogram of gradients and optical flow for facial expression recognition in video sequences. *Pattern Recognition*, 48(11):3407 – 3416. (page 14)

[28] Fanelli, G., Yao, A., Noel, P., Gall, J., and Gool, L. (2010). Hough forest-based facial expression recognition from video sequences. In *ECCV Workshops*, pages 195–206. (page 3)

[29] Fanello, S., Keskin, C., Kohli, P., Izadi, S., Shotton, J., Criminisi, A., Pattacini, U., and Paek, T. (2014). Filter forests for learning data-dependent convolutional kernels. In *CVPR*, pages 1709–1716. (page 92)

[30] Fang, Y. and Chang, L. (2015). Multi-instance feature learning based on sparse representation for facial expression recognition. In He, X., Luo, S., Tao, D., Xu, C., Yang, J., and Hasan, M., editors, *MultiMedia Modeling*, volume 8935, pages 224–233. (page 13)

[31] Felzenszwalb, P., Girshick, R., McAllester, D., and Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645. (page 48)

[32] Friesen, W. and Ekman, P. (1983). *EMFACS-7: Emotional Facial Action Coding System*. University of California Press. (page 10)

[33] Gang, L., Yong, Z., Yan-Lei, L., and Jing, D. (2011). Three dimensional canonical correlation analysis and its application to facial expression recognition. In *Intelligent Computing and Information Science*, pages 56–61. (page 13)

[34] Gehrig, T. and Ekenel, H. (2011). Facial action unit detection using kernel partial least squares. In *Workshop on ICCV*. (page 77)

[35] Gross, R., Matthews, I., Cohn, J., Kanade, T., and Baker, S. (2010). Multi-pie. *Image and Vision Computing*, 28:807–813. (page 71, 101)

[36] Hassner, T. (2013). Viewing real-world faces in 3d. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 3607–3614. (page 30)

[37] Hassner, T., Harel, S., Paz, E., and Enbar, R. (2015). Effective face frontalization in unconstrained images. In *CVPR*, pages 4295–4304. (page 30, 32, 57, 101, 103)

[38] Hesse, N., Gehrig, T., Hua, G., and Ekenel, H. (2012). Multi-view facial expression recognition using local appearance features. In *ICPR*, pages 3533–3536. (page 12, 19)

[39] Honeine, P. (2012). Online kernel principal component analysis: A reduced-order model. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(9):1814–1826. (page 40)

[40] Hu, Y., Zeng, Z., Yin, L., Wei, X., Tu, J., and Huang, T. (2008a). Multi-view facial expression recognition. In *FG*, pages 1–6. (page 12, 17, 18, 78, 82, 96, 106)

[41] Hu, Y., Zeng, Z., Yin, L., Wei, X., Tu, J., and Huang, T. (2008b). A study of non-frontal-view facial expressions recognition. In *ICPR*, pages 1–4. (page 11, 17, 18, 78, 105)

[42] Hu, Y., Zeng, Z., Yin, L., Wei, X., Zhou, X., and Huang, T. (2008c). Multi-view facial expression recognition. In *FG*, pages 1–6. (page 72, 89)

[43] Huang, G. B., Jain, V., and Learned-Miller, E. (2007a). Unsupervised joint alignment of complex images. In *International Conference on Computer Vision (ICCV)*, page 1. (page 30)

[44] Huang, G. B., Ramesh, M., Berg, T., and Learned-Miller, E. (2007b). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst. (page 30)

[45] Huang, X., Zhao, G., and Pietikainen, M. (2013). Emotion recognition from facial images with arbitrary views. In *BMVC*, pages 76.1–76.11. (page 20, 72, 82, 89, 96, 106)

[46] Huang, X., Zhao, G., Pietikainen, M., and Zheng, W. (2014). Robust facial expression recognition using revised canonical correlation. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 1734–1739. (page 13)

[47] Huang, Y., Zhang, X., Fan, Y., Yin, L., Seversky, L., Lei, T., and Dong, W. (2011). Reshaping 3d facial scans for facial appearance modeling and 3d facial expression analysis. In *Automatic Face Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 422–429. (page 12, 13, 14)

[48] Ilbeygi, M. and Shah-Hosseini, H. (2012). A novel fuzzy facial expression recognition system based on facial feature extraction from color face images. *Engineering Applications of Artificial Intelligence*, 25(1). (page 3, 77)

[49] Ishraque, S., Banna, A., and Chae, O. (2012). Local gabor directional pattern for facial expression recognition. In *Computer and Information Technology (ICCIT), 2012 15th International Conference on*, pages 164–167. (page 12)

[50] Jabid, T., Kabir, M., and Chae, O. (2010). Facial expression recognition using local directional pattern (ldp). In *ICIP*, pages 1605–1608. (page 12, 75, 77)

[51] Jampour, M., Lepetit, V., Mauthner, T., and Bischof, H. (2015a). Pose-specific non-linear mappings in feature space towards multiview facial expression recognition. *Image and Vision Computing*, ??:?? (page 23)

[52] Jampour, M., Mauthner, T., and Bischof, H. (2015b). Multi-view facial expressions recognition using local linear regression of sparse codes. In *Proc. Computer Vision Winter Workshop (CVWW)*. (page 23)

[53] Jampour, M., Mauthner, T., and Bischof, H. (2015c). Pairwise linear regression: An efficient and fast multi-view facial expression recognition. In *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, pages 1–8. (page 13, 23, 89, 94, 96)

[54] Jeni, L. A., Takacs, D., and Lorincz, A. (2011). High quality facial expression recognition in video streams using shape related information only. In *ICCV*. (page 77)

[55] Jolliffe, I. (2002). *Principal Component Analysis*. Springer-Verlag New York. (page 40)

[56] Kabir, M., Jabid, T., and Chae, O. (2010). A local directional pattern variance (ldpv) based face descriptor for human facial expression recognition. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, pages 526–532. (page 12)

[57] Kanade, T., Cohn, J., and Tian, Y. (2000). Comprehensive database for facial expression analysis. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 46–53. (page 70)

[58] Kazemi, V., Burenius, M., Azizpour, H., and Sullivan, J. (2013). Multi-view body part recognition with random forests. In *BMVC*, pages 1–11. (page 23)

[59] Kazemi, V. and Sullivan, J. (2014). One millisecond face alignment with an ensemble of regression trees. In *CVPR*. (page 8, 9, 31, 37)

[60] Kazmi, S., ul Ain, Q., Ishtiaq, M., and Jaffar, M. (2010). Texture analysis based facial expression recognition using a bank of bayesian classifiers. In *Information and Emerging Technologies (ICIET), 2010 International Conference on*, pages 1–6. (page 12)

[61] Khan, R., Meyer, A., Konik, H., and Bouakaz, S. (2012). Human vision inspired framework for facial expressions recognition. In *ICIP*, pages 2593–2596. (page 12, 77)

[62] Khan, R. A., Meyer, A., Konik, H., and Bouakaz, S. (2012). Exploring human visual system: study to aid the development of automatic facial expression recognition framework. In *CVPR*. (page 74)

[63] Khan, R. A., Meyer, A., Konik, H., and Bouakaz, S. (2013). Framework for reliable, real-time facial expression recognition for low resolution images. *Pattern Recognition Letters*, 34. (page 77)

[64] Kimmel, R., Elad, M., Shaked, D., Keshet, R., and Sobel, I. (2003). A variational framework for retinex. *International Journal of Computer Vision*, 52(1):7–23. (page 32)

[65] Komodakis, N. and Tziritas, G. (2007). Image completion using efficient belief propagation via priority scheduling and dynamic pruning. *Image Processing, IEEE Transactions on*, 16(11):2649–2661. (page 103)

[66] Kontschieder, P., Bulo, S., Pelillo, M., and Bischof, H. (2014). Structured labels in random forests for semantic labelling and object detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36:2104–2116. (page 23, 24)

[67] Kostrikov, I. and Gall, J. (2014). Depth sweep regression forests for estimating 3d human pose from images. In *BMVC*, pages 1–13. (page 23)

[68] Langner, O., Dotsch, R., Bijlstra, G., Wigboldus, D. H. J., Hawk, S. T., and van Knippenberg, A. (2010). Presentation and validation of the radboud faces database. *Cognition and Emotion*, 24(8):1377–1388. (page 72, 101)

[69] Lee, C.-C. and Shih, C.-Y. (2010). Facial expression recognition using contourlets and regularized discriminant analysis-based boosting algorithm. In *Computer Symposium (ICS), 2010 International*, pages 1–5. (page 12)

[70] Leistner, C., Saffari, A., Santner, J., and Bischof, H. (2009). Semi-supervised random forests. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 506–513. (page 42)

[71] Li, C., Zhou, K., and Lin, S. (2014). intrinsic image decomposition; reflectance models; human face priors. In *ECCV*, pages 218–233. (page 32)

[72] Li, H., Ding, H., Huang, D., Wang, Y., Zhao, X., Morvan, J.-M., and Chen, L. (2015a). An efficient multimodal 2d + 3d feature-based approach to automatic facial expression recognition. *Computer Vision and Image Understanding*, 140:83 – 92. (page 15)

[73] Li, H., Huang, D., Morvan, J.-M., Wang, Y., and Chen, L. (2015b). Towards 3d face recognition in the real: A registration-free approach using fine-grained matching of 3d keypoint descriptors. *International Journal of Computer Vision*, 113(2):128–142. (page 21)

[74] Li, Y., Wang, S., Zhao, Y., and Ji, Q. (2013). Simultaneous facial feature tracking and facial expression recognition. *IEEE Trans. on Image Processing*, 22(7):2559–2573. (page 77)

[75] Liao, C.-T., Chuang, H.-J., Duan, C.-H., and Lai, S.-H. (2013). Learning spatial weighting for facial expression analysis via constrained quadratic programming. *Pattern Recognition*, 46(11). (page 77)

[76] Libralon, G. and Romero, R. (2013). Investigating facial features for identification of emotions. In *Neural Information Processing*. (page 77)

[77] Liu, P., Han, S., Meng, Z., and Tong, Y. (2014). Facial expression recognition via a boosted deep belief network. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1805–1812. (page 15)

[78] Lörincz, A., Jeni, L. A., Szabó, Z., Cohn, J. F., and Kanade, T. (2013). Emotional expression classification using time-series kernels. *CoRR*, abs/1306.1913. (page 77)

[79] Lucey, P., Cohn, J., Kanade, T., Saragih, J., Ambadar, Z., and Matthews, I. (2010). The extended cohn-kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression. In *Workshop on CVPR*, pages 1–8. (page 70)

[80] Mariappan, M. B., Suk, M., and Prabhakaran, B. (2012). Facial expression recognition using dual layer hierarchical svm ensemble classification. In *ISM*. (page 77)

[81] Mian, A. and Pears, N. (2012). 3d face recognition. In Pears, N., Liu, Y., and Bunting, P., editors, *3D Imaging, Analysis and Applications*, pages 311–366. Springer London. (page 21)

[82] Mo, Z., Lewis, J., and Neumann, U. (2004). Face inpainting with local linear representations. In *BMVC*, page 1. (page 101)

[83] Moeini, A., Moeini, H., and Faez, K. (2014). Pose-invariant facial expression recognition based on 3d face reconstruction and synthesis from a single 2d image. In *ICPR*, pages 1746–1751. (page 21)

[84] Mohammad, T. and Ali, M. (2011). Robust facial expression recognition based on local monotonic pattern (lmp). In *Computer and Information Technology (ICCIT), 2011 14th International Conference on*, pages 572–576. (page 12)

[85] Moore, S. and Bowden, R. (2010). Multi-view pose and facial expression recognition. In *BMVC*. (page 12, 18, 87, 94)

[86] Moore, S. and Bowden, R. (2011). Local binary patterns for multi-view facial expression recognition. *Computer Vision and Image Understanding*, 115:541–558. (page 18, 82, 89, 96, 106)

[87] Murphy, K. P., Weiss, Y., and Jordan, M. I. (1999). Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 467–475. Morgan Kaufmann Publishers Inc. (page 65, 100)

[88] Nicolaou, M., Gunes, H., and Pantic, M. (2010). Audio-visual classification and fusion of spontaneous affective data in likelihood space. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 3695–3699. (page 12)

[89] Ojala, T., Pietikainen, M., and Harwood, D. (1994). Performance evaluation of texture measures with classification based on kullback discrimination of distributions. In *ICPR*, pages 582 – 585. (page 38, 79, 92)

[90] O'Toole, A. J., Phillips, P. J., Weimer, S., Roark, D. A., Ayyad, J., Barwick, R., and Dunlop, J. (2011). Recognizing people from dynamic and static faces and bodies: Dissecting identity with a fusion approach. *Vision Research*, 51(1):74 – 83. (page 104)

[91] Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers. (page 65, 100)

[92] Phillips, G. J., Scruggs, W. T., OToole, A. J., Flynn, P. J., Bowyer, K. W., Schott, C. L., and Sharpe, M. (2007). Frvt 2006 and ice 2006 large-scale results. Technical Report 7408, National Institute of Standards and Technology, NISTIR,. (page 30)

[93] Rudovic, O., Pantic, M., and Patras, I. (2013). Coupled gaussian processes for pose-invariant facial expression recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35:1357–1369. (page 22)

[94] Rudovic, O., Patras, I., and Pantic, M. (2010a). Coupled gaussian process regression for pose-invariant facial expression recognition. In *ECCV*, volume 6312, pages 350–363. (page 22)

[95] Rudovic, O., Patras, I., and Pantic, M. (2010b). Regression-based multiview facial expression recognition. In *ICPR*, pages 4121–4124. (page 11, 13, 22)

[96] Rudovic, O., Pavlovic, V., and Pantic, M. (2012). Multi-output laplacian dynamic ordinal regression for facial expression recognition and intensity estimation. In *CVPR*, pages 2634–2641. (page 22)

[97] Ruffer, B., Kellett, C., Dower, P., and Weller, S. (2010). Belief propagation as a dynamical system: the linear case and open problems. *Control Theory Applications, IET*, 4(7):1188–1200. (page 65)

[98] Russell, J. A. and Dols, J. M. F. (1977). *The psychology of facial expression*. Cambridge University Press. (page 9)

[99] Sadeghi, H., Raie, A.-A., and Mohammadi, M.-R. (2013). Facial expression recognition using geometric normalization and appearance representation. In *Machine Vision and Image Processing (MVIP), 2013 8th Iranian Conference on*, pages 159–163. (page 15)

[100] Saffari, A., Leistner, C., and Bischof, H. (2009). Regularized multi-class semi-supervised boosting. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 967–974. (page 42)

[101] Sakar, C., Kursun, O., Karaali, A., and Erdem, C. (2012). Feature extraction for facial expression recognition by canonical correlation analysis. In *Signal Processing and Communications Applications Conference (SIU), 2012 20th*, pages 1–3. (page 13)

[102] Sandbach, G., Zafeiriou, S., Pantic, M., and Yin, L. (2012). Static and dynamic 3d facial expression recognition: A comprehensive survey. *Image Vision Comput.*, 30(10):683–697. (page 13)

[103] Schulter, S., Leistner, C., and Bischof, H. (2015). Fast and accurate image upscaling with super-resolution forests. In *CVPR*, pages 3791–3799. (page 23, 24)

[104] Schulter, S., Leistner, C., Wohlhart, P., Roth, P., and Bischof, H. (2014). Accurate object detection with joint classification-regression random forests. In *CVPR*, pages 923–930. (page 23)

[105] Sha, T., Song, M., Bu, J., Chen, C., and Tao, D. (2011). Feature level analysis for 3d facial expression recognition. *Neurocomputing*, 74(12):2135–2141. (page 12, 13)

[106] Shao, J., Gori, I., Wan, S., and Aggarwal, J. (2015). 3d dynamic facial expression recognition using low-resolution videos. *Pattern Recognition Letters*, pages –. (page 14)

[107] Shen, J., Yang, X., Jia, Y., and Li, X. (2011). Intrinsic images using optimization. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3481–3487. (page 32)

[108] Shen, L., Yeo, C., and Hua, B.-S. (2013). Intrinsic image decomposition using a sparse representation of reflectance. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(12):2904–2915. (page 32)

[109] Shotton, J., Criminisi, A., and Konukoglu, E. (2011). Decision forests for classification, regression, density estimation, manifold learning and semi-supervised learning. In *Technical Report TR-2011-114, Microsoft Research Cambridge*, page 1. (page 24)

[110] Shotton, J., Johnson, M., and Cipolla, R. (2008). Semantic texton forests for image categorization and segmentation. In *CVPR*, pages 1–8. (page 24)

[111] Society, T. A. (2015). Facts and statistics. `http://www.autism-society.org/what-is/facts-and-statistics/`. (page 2)

[112] Soyel, H. and Demirel, H. (2007). Facial expression recognition using 3d facial feature distances. *Image Analysis and Recognition*, 4633:831. (page 12)

[113] Suk, M. and Prabhakaran, B. (2014). Real-time mobile facial expression recognition system – a case study. In *CVPRW*, pages 132–137. (page 14)

[114] Sun, J., Yuan, L., Jia, J., and Shum, H.-Y. (2005). Image completion with structure propagation. *ACM Trans. Graph.*, 24(3):861–868. (page 65, 100, 103)

[115] Tang, H., Hasegawa-Johnson, M., and Huang, T. (2010). Non-frontal view facial expression recognition based on ergodic hidden markov model supervectors. In *ICME*, pages 1202–1207. (page 12, 72, 89, 96, 106)

[116] Tariq, U., Yang, J., and Huang, T. (2012). Multi-view facial expression recognition analysis with generic sparse coding feature. In *ECCV*, pages 578–588. (page 13, 21, 82, 89, 96, 106)

[117] Tariq, U., Yang, J., and Huang, T. (2013). Maximum margin gmm learning for facial expression recognition. In *FG*, pages 1–6. (page 12, 82, 89, 96, 106)

[118] Tariq, U., Yang, J., and Huang, T. S. (2014). Supervised super-vector encoding for facial expression recognition. *Pattern Recognition Letters*, 46:89–95. (page 72, 82, 89, 96, 106)

[119] Timofte, R., De, V., and Gool, L. V. (2013). Anchored neighborhood regression for fast example-based super-resolution. In *ICCV*, pages 1920–1927. (page 41, 53)

[120] Tropp, J. and Gilbert, A. (2007). Signal recovery from random measurements via orthogonal matching pursuit. *Information Theory, IEEE Transaction on*, 53:4655–4666. (page 41, 51, 79)

[121] Turk, M. and Pentland, A. (1991a). Eigenfaces for recognition. *J. Cognitive Neuroscience*, 3(1):71–86. (page 29)

[122] Turk, M. and Pentland, A. (1991b). Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR '91., IEEE Computer Society Conference on*, pages 586–591. (page 29)

[123] Uddin, M. Z. (2015). Chapter 26 - a local feature-based facial expression recognition system from depth video. In Deligiannidis, L. and R., A. H., editors, *Emerging Trends in Image Processing, Computer Vision and Pattern Recognition*, pages 407 – 419. Morgan Kaufmann. (page 14)

[124] Viola, P. and Jones, M. (2004). Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154. (page 8, 30, 47, 73, 74)

[125] V.M., P. and Chellappa, R. (2013). *Sparse Representations and Compressive Sensing for Imaging and Vision*. Springer-Verlag New York. (page 41)

[126] Vo, A. and Ly, N. (2015). Facial expression recognition using pyramid local phase quantization descriptor. In *Knowledge and Systems Engineering*, pages 105–115. (page 12)

[127] Wang, X., Han, T., and Yan, S. (2009). An hog-lbp human detector with partial occlusion handling. In *ICCV*, pages 32–39. (page 38, 80)

[128] Ward, M. (2015). Facial action coding system. `http://web.cs.wpi.edu/~matt/courses/cs563/talks/face_anim/ekman.html`. (page 9)

[129] Yedidia, J. S., Freeman, W. T., and Weiss, Y. (2003). Understanding belief propagation and its generalizations. *Exploring Artificial Intelligence in the New Millennium*, 8:239–269. (page 67)

[130] Yin, L., Wei, X., Sun, Y., Wang, J., and Rosato, M. (2006). A 3d facial expression database for facial behavior research. In *FG*, pages 211–216. (page 72, 101)

[131] Zhang, D., Ding, D., Li, J., and Liu, Q. (2015a). Pca based extracting feature using fast fourier transform for facial expression recognition. In Yang, G.-C., Ao, S.-I., Huang, X., and Castillo, O., editors, *Transactions on Engineering Technologies*, pages 413–424. (page 13, 18)

[132] Zhang, D., Yang, M., and Feng, X. (2011). Sparse representation or collaborative representation: Which helps face recognition? In *ICCV*, pages 471–478. (page 41, 53)

[133] Zhang, S., Li, L., and Zhao, Z. (2012). Facial expression recognition based on gabor wavelets and sparse representation. In *Signal Processing (ICSP), 2012 IEEE 11th International Conference on*, volume 2, pages 816–819. (page 12)

[134] Zhang, W., Zhang, Y., Ma, L., Guan, J., and Gong, S. (2015b). Multimodal learning for facial expression recognition. *Pattern Recognition*, 48(10):3191 – 3202. (page 15)

[135] Zhang, Z., Wang, L., Zhu, Q., Chen, S.-K., and Chen, Y. (2015c). Pose-invariant face recognition using facial landmarks and weber local descriptor. *Knowledge-Based Systems*, 84:78–88. (page 12)

[136] Zhao, Q., Tan, P., Dai, Q., 0003, L. S., Wu, E., and Lin, S. (2012a). A closed-form solution to retinex with nonlocal texture constraints. *Pattern Analysis and Machine Intelligence (PAMI)*, 34(7):1437–1444. (page 64, 100)

[137] Zhao, Q., Tan, P., Dai, Q., Shen, L., Wu, E., and Lin, S. (2012b). A closed-form solution to retinex with nonlocal texture constraints. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 34(7):1437–1444. (page 32)

[138] Zhao, W. and Chellappa, R. (2005). *Face Processing: Advanced Modeling and Methods*. Academic Press, Inc., Orlando, FL, USA. (page 29)

[139] Zheng, W. (2014a). Multi-view facial expression recognition based on group sparse reduced-rank regression. *Affective Computing, IEEE Transactions on*, 5:71–85. (page 88)

[140] Zheng, W. (2014b). Multi-view facial expression recognition based on group sparse reduced rank regression. *Affective Computing IEEE Trans. on*, 5(1):71–85. (page 94)

[141] Zheng, W., Tang, H., Lin, Z., and Huang, T. (2009). A novel approach to expression recognition from non-frontal face images. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1901–1908. (page 18)

[142] Zheng, W., Tang, H., Lin, Z., and Huang, T. (2010). Emotion recognition from arbitrary view facial images. In *ECCV*, pages 490–503. (page 18, 72, 82, 89, 96, 106)

[143] Zhu, X., Lei, Z., Yan, J., Yi, D., and Li, S. Z. (2015a). High-fidelity pose and expression normalization for face recognition in the wild. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 787–796. (page 57)

[144] Zhu, X. and Ramanan, D. (2012). Face detection, pose estimation, and landmark localization in the wild. In *CVPR*, pages 2879–2886. (page 8, 9, 31, 37, 47)

[145] Zhu, X., Yan, J., Yi, D., Lei, Z., and Li, S. (2015b). Discriminative 3d morphable model fitting. In *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, volume 1, pages 1–8. (page 28, 30)

[146] Zhu, X., Yi, D., Lei, Z., and Li, S. (2014). Robust 3d morphable model fitting by sparse sift flow. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 4044–4049. (page 30)

[147] Zhuang, Y.-t., Wang, Y.-s., Shih, T. K., and Tang, N. C. (2009). Patch-guided facial image inpainting by shape propagation. *Journal of Zhejiang University SCIENCE A*, 10(2):232–238. (page 101)