THE UNIVERSITY OF
SYDNEY

TOWARDS REALISTIC FACIAL EXPRESSION RECOGNITION

A thesis submitted in fulfilment of the requirements for

the degree of Doctor of Philosophy in the

School of Information Technologies at

The University of Sydney

Kaimin Yu

October 2013

ABSTRACT

Automatic facial expression recognition has attracted significant attention over the past decades due to its importance in a wide range of applications such as human-computer interaction (HCI), image or video understanding and affective computing. Although substantial progress has been achieved for certain scenarios (such as frontal faces in strictly controlled laboratory settings), accurate recognition of facial expressions in realistic environments remains unsolved for the most part. The research presented in this thesis investigates solutions to three major issues: collecting a large amount of realistic training data from the Web, devising several novel facial features and a multi-modal recognition algorithm, and exploring facial expression recognition on image sequences.

The problem of lacking realistic training data is the first focus of the thesis. The majority of existing facial expression datasets are collected under controlled environments, which cannot represent the diverse set of variations found in the real world. Since it is often costly to collect a large amount of training examples, they also suffer from common limitations of being small in size. This thesis proposes to solve the problem by utilizing a web search based framework to collect realistic facial expression dataset from the Web. By adopting an active learning based method to remove noisy images from search results returned by commercial search engines, the proposed

approach minimizes the human efforts during the dataset construction and maximizes the scalability for future research.

Another focus is on developing robust facial feature extraction methods. Due to the high intraclass variations and interclass similarities, effective feature extraction is vital to facial expression recognition. The extracted features should represent different types of facial expressions in a way which is not significantly affected by age, gender, or appearance of the subjects. It is also desirable to have features which are robust to face localization errors and occlusions. In the thesis, three novel facial feature extraction methods are introduced, namely Multiscale-WLD (MWLD), Spatially enhanced Local Binary Pattern (SLBP) and Local Patch Pattern (LPP). The first two features exploit the spatial layout of local texture patterns. LPP is proposed to tackle the problem of facial expression recognition in unconstrained environments. Comprehensive experiments on a range of benchmark datasets demonstrate their effectiveness.

The problem of facial feature combination is solved using a novel spectral embedding based feature fusion framework. By assuming that facial expression features extracted from one type of expressions form a manifold embedded in a high dimensional feature space, a neighborhood graph is constructed to encode the structure of the manifold locally. After the Laplacian matrices associated with the neighborhood graph from each view are combined, a unified low dimensional feature space is obtained by performing spectral analysis of the combined matrix. The experimental results clearly demonstrate the effectiveness of the proposed feature fusion framework on realistic facial expression data.

Lastly, we systematically investigate how the number of frames of a facial expression sequence can affect the performance of facial expression recognition algorithms, since facial expression sequences may be captured under different frame rates in realistic scenarios. A facial expression keyframe selection method is proposed based on keypoint based frame representation. Experimental results indicate that the proposed keyframe selection method can reduce the number of frames without clearly compromising recognition accuracy.

To my family...

ACKNOWLEDGMENTS

First and foremost I want to express my gratitude and thanks to my advisors, Dr. Zhiyong Wang and Prof. Dagan Feng. I have learnt a lot from them as a researcher and also as a person par excellence. This thesis would not have been completed without their guidance and encouragement.

I would also like to thank my friends and colleagues in the School of Information Technologies, especially those from the Biomedical & Multimedia Information Technology (BMIT) Research Group. Their support has been invaluable throughout my PhD study, making my time both memorable and enjoyable.

Last but not least, I thank my family for their unconditional love and support without which this journey would not have been possible.

PUBLICATIONS

Journal Publications

Kaimin Yu, Zhiyong Wang, Li Zhuo, Jianjun Wang, Zheru Chi and Dagan Feng. Learning realistic facial expressions from web images. *Pattern Recognition*, 46(8):2144-2155, 2013.

Kaimin Yu, Zhiyong Wang, Markus Hagenbuchner, and Dagan Feng. Spectral Embedding based Facial Expression Recognition with Multiple Features. *Neurocomputing*. (Accepted on 21st September 2013).

Kaimin Yu, Zhiyong Wang, Lei Yue and Dagan Feng. Spatially enhanced local binary pattern. *Electronic Letters*, 48(25), 2012.

Conference Publications

Kaimin Yu, Zhiyong Wang, Genliang Guan, Qiuxia Wu, Zheru Chi and Dagan Feng. How Many Frames Does Facial Expression Recognition Require?. In: *Proceedings of the 2012 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, Pages 290-295, 2012.

Kaimin Yu, Zhe Li, Genliang Guan, Zhiyong Wang and Dagan Feng. Unsupervised Text segmentation using LDA and MCMC. In: *The Australasian Data Mining Conference (AusDM)*, 2012.

Genliang Guan, Zhiyong Wang, Kaimin Yu, Shaohui Mei Ming-yi Y. He and Dagan Feng. Video Summarization with Global and Local Features. In: *Proceedings of the 2012 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, Pages 570-575, 2012.

Kaimin Yu, Zhiyong Wang, Zhuo Li and Dagan Feng. Harvesting Web Images for Realistic Facial Expression Recognition. In: *DICTA '10 Proceedings of the 2010 International Conference on Digital Image Computing: Techniques and Applications*, Pages 516-521, 2010.

TABLE OF CONTENTS

Page

Abstract iii

Dedication vi

Acknowledgments vii

List of Publications viii

List of Tables xii

List of Figures xiv

Chapters:

1. Introduction 1

    1.1 Motivations 1
    1.2 Objectives 3
    1.3 Contributions 4
    1.4 Organization of the thesis 5

2. State-of-the-Arts in Facial Expression Recognition 7

    2.1 Face Detection and Registration 9
    2.2 Facial Feature Extraction and Selection 13
    2.3 Facial Expression Classification 19
    2.4 Summary 22

3. Harvesting Web Images for Realistic Facial Expression Recognition . . . 23

3.1 Introduction . . . 23
3.2 Related Work . . . 27
3.3 Text-based Web Image Search . . . 32
3.4 Content-based Refinement . . . 34
3.5 Multiscale-WLD Based Facial Expression Feature . . . 37
3.6 Experiments and Discussions . . . 42
3.7 Conclusion . . . 55

4. Facial Expression Features . . . 58

4.1 Spatially enhanced Local Binary Pattern . . . 58
4.2 Local Patch Pattern . . . 68

5. Spectral Embedding based Facial Expression Recognition . . . 88

5.1 Introduction . . . 88
5.2 Related Work . . . 91
5.3 Multiview Spectral Embedding . . . 92
5.4 Facial Expression Features . . . 99
5.5 Experimental Results and Discussions . . . 105
5.6 Conclusion . . . 111

6. Towards Video based Facial Expression Recognition . . . 113

6.1 Introduction . . . 113
6.2 Keypoint Based Keyframe Selection . . . 115
6.3 Facial Expression Recognition Method . . . 118
6.4 Experiments . . . 121
6.5 Conclusion . . . 128

7. Conclusion . . . 129

7.1 Main Contributions . . . 129
7.2 Future Work . . . 132

Bibliography . . . 134

LIST OF TABLES

Table **Page**

3.1 A list of keywords used for text based facial expression image search. 33

3.2 Statistics of the evaluation datasets used in the experiments. The columns of Male and female show the percentage of male subjects and female subjects respectively. 47

3.3 Confusion matrix for the classification results on our G_w. The classifiers are trained on G_w and evaluated on G_v. 49

3.4 Confusion matrix for the 5-fold cross validation results on the CK dataset. 49

3.5 Confusion matrix for the 5-fold cross validation results on the JAFFE dataset. 50

3.6 Comparison of recognition performance when training and testing happen on the same and different datasets. We use G_w as our training dataset and G_v as our evaluation dataset. 50

3.7 Benchmark results on BU-3DFE dataset with classifiers trained on G_w, $JAFFE$ and CK respectively. 52

3.8 A comparison of 5-fold cross validation results for different WLD and LBP based facial feature on different evaluation datasets. 54

4.1 Classification accuracy of our method compared with LBP. 66

4.2 Influence of rotation invariance. 66

4.3 Influence of rotation invariance. 67

4.4 Statistics of the evaluation datasets used in the experiments. The columns of Male and female show the percentage of male subjects and female subjects respectively. 81

4.5 Frontal facial expression recognition results on CK+ 84

5.1 Statistics of the evaluation datasets used in the experiments. The columns of Male and female show the percentage of male subjects and female subjects respectively. 106

5.2 Comparison of the multiview feature with single view features. 108

5.3 Comparison of different feature fusion techniques. 110

6.1 Comparison of recognition accuracy with keyframes, uniformly sampled frames, and the whole sequence frames. 123

6.2 Image based facial expression recognition results of using a single keyframe and the peak frame (i.e. the last frame). 127

LIST OF FIGURES

Figure **Page**

2.1 The earliest systematic investigation of facial expressions [1] 8

2.2 The three main components of a typical facial expression recognition system 9

2.3 A classifiers cascade with 3 levels [2] 11

2.4 Geometric Features [3] vs. Appearance Features [4] 13

2.5 ASM: The top row of images are manually labeled for training, the bottom row shows the point distribution model(PDM) learned by PCA [5] 14

2.6 Multistate Face Component Model [6] 16

2.7 Texture primitives represented by LBP [7] 18

2.8 Submanifold 20

3.1 Overview of the active learning based framework for constructing facial expression datasets. 25

3.2 Illustration of neighborhood pixels used for extracting the WLD descriptor of x_c. 39

3.3 Multiscale-WLD Facial Expression Feature. 39

3.4 The weighting scheme used in our experiments. 43

3.5 Images from our actively collected facial expression dataset G_w. . . . 45

3.6 Precision of images in G_w (blue and red, the left two bars) compared to first 100 Google image search results (green, the right bar). 46

3.7 Comparison of classification performance between active learning and passive learning for *happiness* expression in G_v. The classification accuracy is defined as the ratio of the number of correctly classified images to the total number of images in the test dataset. 56

3.8 Impact of parameter settings. The Y-axes denote the recognition accuracy. 57

4.1 Illustration of the spatial distribution of a same texture pattern in four different texture images. It demonstrates the importance of spatial information in describing textures. 60

4.2 A sample illustration of applying the original LBP operator to a pixel, where the circulation starts at the top-left position. 62

4.3 Illustration of a LBP coded face image. 63

4.4 Illustration of computing the shape context for the sampling point (*i.e.* the origin of the log-polar coordinate system). 64

4.5 Illustrations of the two types of head pose variations. The first row shows the head pose variations caused by in-plane rotations and the second row shows the out-plane variations. 69

4.6 Workflow of the BOW based framework for facial expression recognition. The red crosses in (b) refer to the densely sampled keypoint locations, and the different shapes in (c) refer to different visual words obtained using KNN. 74

4.7 Illustration of the LPP descriptor. The red dot in the center indicates the keypoint. 78

4.8 Impact of parameter settings. The y-axis denotes the recognition accuracy. The red solid line corresponds to the results obtained using Histogram Intersection kernel and the blue dotted line corresponds to the results obtained using the RBF kernel. 83

4.9 Multiview Facial expression recognition results on BU-3DFE. Figure 4.9(a) corresponds to the results obtained using the RBF kernel and Figure 4.9(b) corresponds to the results obtained using the Histogram Intersection kernel. (The figure is best viewed in color.) 86

5.1 Illustration of conventional and supervised neighborhood graph construction methods. Assume the only available distance function is based on the face shape, the data points representing different facial expression form a neighborhood in 5.1(a) which is not intended. In contrast, the supervised graph illustrated in 5.1(b) is preferred as points depicting same type of expressions are connected by utilizing the label information (*i.e.* color). 96

5.2 Multiscale-WLD Facial Expression Feature. 100

5.3 Illustration of AAM shape registration. 104

5.4 Impact of parameter settings. The y-axis denotes the recognition accuracy. 112

6.1 Inter-window chaining of keypoints. 116

6.2 Intra-window chaining of keypoints. 117

6.3 Basic LBP operator. 119

6.4 Three Orthogonal Planes, XY, XT and YT. 120

6.5 Sample facial expression images from the CK dataset. In clockwise order, *happiness*, *sadness*, *fear*, *disgust*, *surprise* and *anger*. 121

6.6 Keyframe selection results for s052_004. The original sequence (the upper part) is organized in temporal order from left to righ and top to bottom. The selected keyframes are highlighted with red rectangles. The overlay numbers indicate the selection order. Below each original expression sequence, the keyframes are organized in the selection order. The numbers at the bottom are the frame numbers of the original sequence. 123

6.7 Keyframe selection results for s074_001 125

6.8 Comparison of recognition results with keyframes and with uniform sampling under different numbers of target frames. Since the number of frames obtained through keyframe selection algorithm under full coverage is fewer than 9, the results of Keyframe-based approach is not available at 9 and 10. 126

CHAPTER 1

INTRODUCTION

This thesis investigates the problem of automatic facial expression recognition, with the ultimate goal to identify facial expressions in unconstrained environments. The chapter introduces the motivations behind our work and its objectives. Then the challenges involved in automatic facial expression recognition are discussed. Finally, it is concluded with a discussion of the major contributions and an outline of the structure of the thesis.

1.1 Motivations

Facial expression is the most expressive way for human to communicate emotions and signal intentions, which conveys non-verbal communication cues in human face to face interactions. Previous studies [8] have demonstrated that faical expression accounts for more than 50 percent to the effect of a spoken message. Facial expression recognition aims to identify emotional states of humans from faces. It is a main component of the emerging affective computing, which focuses on understanding the affective states of users and responding accordingly to the affecting signals. Automatic facial expression recognition systems also play an important role for the next generation human computing interaction where the user interfaces are argued to

be human centered, and they are capable of sensing and reacting to user's affective feedbacks in a more natural way.

Although recognizing facial expressions is a relatively effortless task for the majority of human beings, it is a very challenging task for a computer. One reason is that it has been observed that the variations among the images depicting the same expressions due to the change of illumination and view directions are almost always larger than variations from the change in facial expressions. These variations are increased by additional factors such as occlusion, gender, and even ethnic origins. Such appearance variations make it difficult to locate facial regions and extract the inherent facial expression features. Under unconstrained environments, these variations become even greater thus harder to model than under well controlled environments.

Feature extraction is a crucial step in facial expression recognition and largely determines the effectiveness of the performance. Therefore, different types of features and selecting a suitable type for facial expression representation play an essential role in designing facial expression recognition systems. During the past decades, a large number of facial expression features have been proposed. However, only relatively few recent studies consider the combination of different features. In addition, most of these comparisons are evaluated based on posed datasets rather than realistic data which leads to our next point.

Most datasets used in previous studies are collected under highly constrained laboratory environments. The resulting facial expression data cannot fully reflect the variations found in real world, such as illumination, intensity and poses. Therefore, most existing datasets are not optimal for evaluating facial expression recognition systems toward real world applications.

1.2 Objectives

The ultimate goal of this thesis is to investigate facial expression recognition in unconstrained environments. Despite the effort of researchers over a couple of decades, this problem has remained unsolved for the most part. Toward the goal, the objectives of this thesis can be split into three parts which will be pursued separately.

The first objective is to deal with the problem of lacking training data. The majority of the existing facial expression datasets were collected under controlled environments which can not represent the diverse set of variations found in the real world. Since it is often costly to collect a large amount of training examples, they also suffer from common limitations of being small in terms of both the number of human subjects and images which can lead to over-fitting problems in many learning algorithms and result in poor recognition performance. In this thesis, we aim to develop method that is able to construct large scale facial expression image dataset from web images with minimum human efforts.

The second objective is to investigate novel feature extraction methods. Due to the high intraclass variations and interclass similarities, effective feature extraction is vital to facial expression recognition. The extracted features should represent different types of facial expressions in a way which is not significantly affected by age, gender, or appearance of the subject. It is also desirable to have features which are robust to face localization errors and occlusions.

The third objective is to investigate feature selection and combination methods for facial expression recognition. It is commonly acknowledged that the performance of facial expression recognition can benefit from a combination of multiple features. However there is often no obvious way to select and combine different types of features.

If redundant or noisy features are chosen at the expense of discriminant features, the recognition performance can be adversely affected. To make the matter worse, facial expression features are typical of very high dimension. A simple concatenation of different features may greatly increase the computation cost and lead to inferior results. Therefore the third objective of this thesis is to investigate methods that can combine different facial expression features to form a more descriptive representation.

1.3 Contributions

The main contributions of this thesis can be summarized as follows:

- A web search based framework is proposed to build a realistic facial expression dataset to solve the problem of lacking training and testing data. The dataset contains a diverse set of human subjects and imaging environments. By adopting an active learning based method to remove noisy images from the search results returned by commercial search engines, the proposed approach minimizes the human efforts during the dataset construction and maximizes the scalability for future research. To the best of our knowledge, there exist very limited studies in the literature on building general facial expression datasets using web images.

- Development of three novel facial expression features, namely Multiscale Weber Local Descriptor (MWLD), Spatially enhanced Local Binary Patterns (SLBP) and Local Patch Patterns (LPP). The first two features are developed to encode the spatial layout of local texture patterns. LPP is proposed to tackle the problem of unconstrained facial expression recognition by combining local feature descriptors extracted from neighboring patches to form a second order

representation. Experiments on a range of benchmark datasets demonstrates their effectiveness.

- A spectral embedding based feature fusion framework is proposed to tackle the problem of facial expression feature selection and combination. By assuming that facial expression features extracted from one type of expressions form a manifold embedded in a high dimensional feature space, a neighborhood graph is constructed to encode the structure of the manifold locally. After the Laplacian matrices associated with the neighborhood graph from each view are combined, a unified low dimensional feature space is obtained by performing spectral analysis of the combined matrix.

- Systematic investigations are performed on how the number of frames of a facial expression sequence can affect the accuracy of facial expression recognition. A facial expression keyframe selection method is proposed based on keypoint based frame representation.

1.4 Organization of the thesis

The reminder of this thesis is organized as follows.

Chapter 2 presents a general overview of different aspects of automatic facial expression recognition and describes the state of art in this area.

Chapter 3 proposes a search based framework to harvest facial expression images from the web to address the problem of lacking a large-scale facial expression dataset that is collected under real world conditions. This chapter includes details about the implementation of the method as well as comprehensive experiments demonstrating the benefits of adapting realistic images for training the facial expression algorithms.

Chapter 4 presents two novel facial feature extraction algorithms to handle facial expression images that are collected in highly unconstrained conditions using the framework discussed in the previous chapter. The first feature incorporates the spatial contextual information into the famous LBP descriptor using a shape context based representation. The other one combines local feature descriptors extracted from neighboring patches to form a second order representation. Both features are more descriptive and robust in the presence of noise, which are commonly observed in practical environments.

Chapter 5 focuses on the feature selection and combination methods. Due to the high dimensionality, direct manipulation on facial expression feature descriptors is highly computationally expensive and often results in suboptimal recognition performance. Inspired by approaches from related domains, we propose a framework that treats feature selection and fusion as a multiview dimension reduction problem and aim to find a unified low dimensional subspace that captures information from all sources (*i.e.* different feature spaces) by preserving local geometric properties of the original features.

Chapter 6 presents the frame selection method that is able to choose a small subset of frames to represent a facial expression sequence, such that the computational cost can be greatly reduced without sacrificing the recognition accuracy.

Finally, Chapter 7 summarizes our work and key findings, and provides some suggested directions for future research.

CHAPTER 2

STATE-OF-THE-ARTS IN FACIAL EXPRESSION RECOGNITION

This chapter reviews the state-of-the-arts in automatic facial expression recognition, with particular attention to the works relevant to our own investigations. Facial expression has been systematically studied for over 150 years. To our knowledge, the earliest publication with regard to facial expression analysis can be traced to 1862 [1] (see Figure 2.1). The interest of using facial expression as a key clue to human emotions is renewed since 1960s, when Ekman linked expression to a group of universal emotions shared by all human beings [9]. The ideas of using facial expressions to measure human emotions become the mainstream for the past 30 years in psychology research. In 1978, Suwa *et al.* [10] proposed the first approach to automatic facial expression analysis by tracking the motion of 20 identified spots on image sequences. Since then, numerous systems have been developed to automatically analysis facial expressions from static images and dynamic image sequences. The early works have been summarized by Samal and Lyengar [11], Pantic and Rothkrantz [12], Fasel and Luettin [13], Tian et al. [14]. Recent advances are surveyed by Zeng et al. [8].

Figure 2.1: The earliest systematic investigation of facial expressions [1]

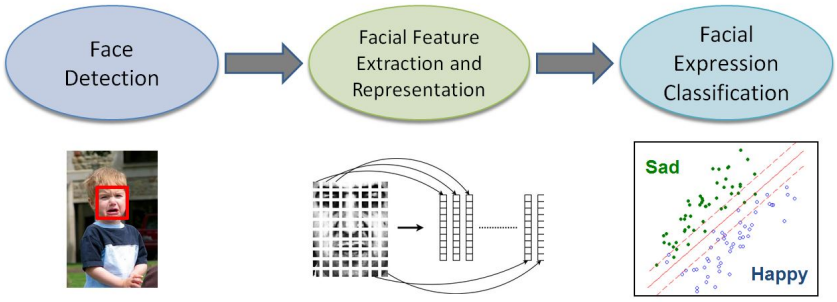


Figure 2.2: The three main components of a typical facial expression recognition system

The general approach to automatic facial expression recognition can be divided into three components as shown in Figure 2.2. The first is face detection and registration. It involves the process of locating face regions from input data, and align the faces to a common coordinate system. The second component is facial feature extraction and representation which is responsible for extracting and representing the facial changes caused by facial expressions. The last component is facial expression classification. The facial changes can be identified as facial action units or prototypic emotional expressions. In the following sections, we review the relevant work according to these three components.

2.1 Face Detection and Registration

The face detection and registration problem involves identifying the presence of faces in an image and determining the locations and scales of the faces. The accuracy of face detection and registration is particularly important in realistic conditions, where the presence of face in a scene and the global locations of the face are not

known a priori. This section reviews the eye finding and face normalization tasks. Emphasis is placed on the most recent advances in the field. Readers can refer to [15] for previous works.

A typical face detection algorithm [2, 16, 17, 18] performed the detection in the following steps. Given a set of training images acquired in a fixed pose (e.g. frontal or near-frontal), histogram equalization or standardization is performed to minimize the effects of illumination. After this pre-processing step, certain face patterns are extracted with knowledge based or learning based methods. Here the knowledge based methods model the face patterns by some explicit rules, such as facial components, face textures or skin color; the learning based methods model the face patterns by learning from a set of data with some discriminant functions. With the extracted face patterns, the system scans through the entire image to locate the faces. This scanning process can be repeated at various coarser scales of the original image in order to find faces at different scales. The raw detected face regions are further processed to remove the overlapping matches. This section will focus on the cascade based face detectors due to its dominance in the recent literature.

The AdaBoost based face detector by Viola and Jones [2] is arguably the most commonly used face detector in automatic face recognition and expression analysis. The basic idea of the approach is to train a cascade classifier (see Figure 2.3) for haar-like rectangular features. The detector then scans an image at different scales and positions by a sub-window, and the regions accepted by the classifier are declared as faces. The haar-like rectangular features can be efficiently computed with integral images [2], which allows the approach to achieve real time detection speed. To further increase the detection speed while retaining the accuracy, AdaBoost [19] was used to

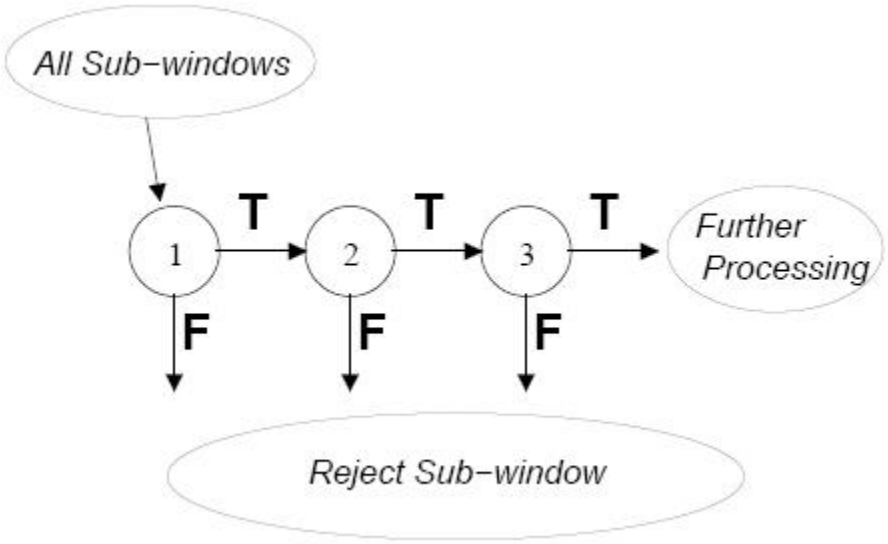


Figure 2.3: A classifiers cascade with 3 levels [2]

select the representative haar-like features. Moreover instead of training a single strong classifier, a number of weak classifiers are constructed. The weak classifiers are combined into a cascade. The motivation is that simple classifiers at the beginning of the cascade can efficiently rejects non-face regions, while stronger classifiers later in the cascade simply need to classify the more face-like regions. The final face detector with 38 layers achieves impressive accuracy and very rapid detection speed.

Several extensions have made to [2] for detecting faces from different views (e.g. frontal and profile, faces with in plane rotations) [20, 21, 22]. In [20], separator Viola-Jones face detectors [2] were trained for different fixed views of the faces. The face detection is performed by estimating the view of each detection window with a decision tree constructed using features described in [2]. Based on the predicted view, the detection windows are passed to the corresponding Viola-Jones face detectors. The

union of all detected regions are reported as faces. Li et al. [21] proposed a modified AdaBoost algorithm for learning face and non-face classifiers. To efficiently detect multi-view faces, the classifiers are organized in a coarse-to-fine, simple-to-complex pyramid-like structures. In [22], a width-first-search tree structure for constructing face detector was proposed which is reported to obtain significant improvements in speed and accuracy for multi-view face detections. In [20, 21, 22, 23, 24], the original four type of Haar-like features were extended to represent complex or multi-view face patterns.

Despite the excellent detection speed, Viola-Jones face detector has the drawback of long training time. Some methods have been proposed to address this issue. In [2], the classification learning algorithm is AdaBoost, which does feature selection and classifier training simultaneously. Wu et al. [25] used a greedy feature selection algorithm to determine the set of features before training the cascade classifier, they reported that the training efficiency can be greatly increased with this approach. In [26], Pham and Cham proposed an online learning algorithm that learns asymmetric boosted classifiers with significant improvements in training time.

There are a few other face detection approaches besides the cascade face detectors in the recent literature, including the component-based face detector using Naive Bayes classifiers [18], the face detectors using support vector machines [27], the face detectors trained with positive image only [28], and the energy-based method that simultaneously detects faces and estimates the poses [29].

2.2 Facial Feature Extraction and Selection

After the presence and location of a face are detected, the next step is to extract the representative features about the shown facial expression. Obtaining effective facial expression features from the detected face image is crucial for successful facial expression recognition. The optimal features should minimize within-class variations of expression while maximizing across-class variations. The common facial expression features can be divided into two groups: geometric features and appearance features. Geometric features represent the shape and location of facial components or predefined facial feature points (see Figure 2.4), which are extracted to form a feature vector to represent the face geometry. Appearance features represent the skin texture changes of the face, such as wrinkles and furrows, which are normally obtained by applying image filters (such as Gabor wavelets) to a face image.

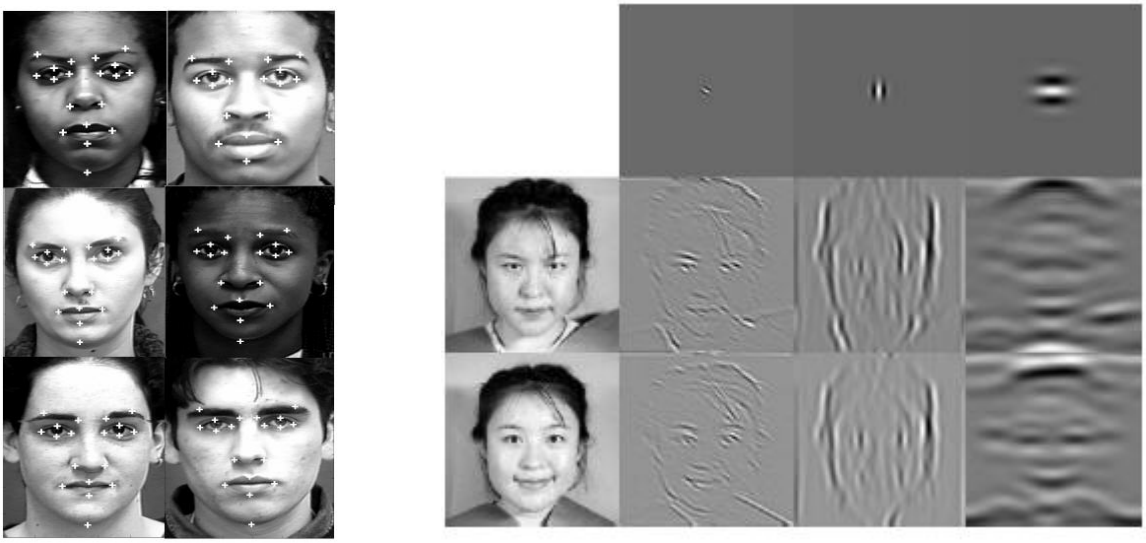


Figure 2.4: Geometric Features [3] vs. Appearance Features [4]

Geometric Feature Extraction

Geometric facial features have been widely used for facial representation [6, 30, 31, 32, 33, 34], where shapes and locations of facial components or facial feature points are extracted to represent the face geometry. In [30], 34 fiducial points are used to represent a face image. The fiducial points are manually selected at facial landmarks (eg. corners of mouth, inner eye canthus), and the image coordinates of these points are used as features. Hence each face image is represented by a vector of 68 elements.

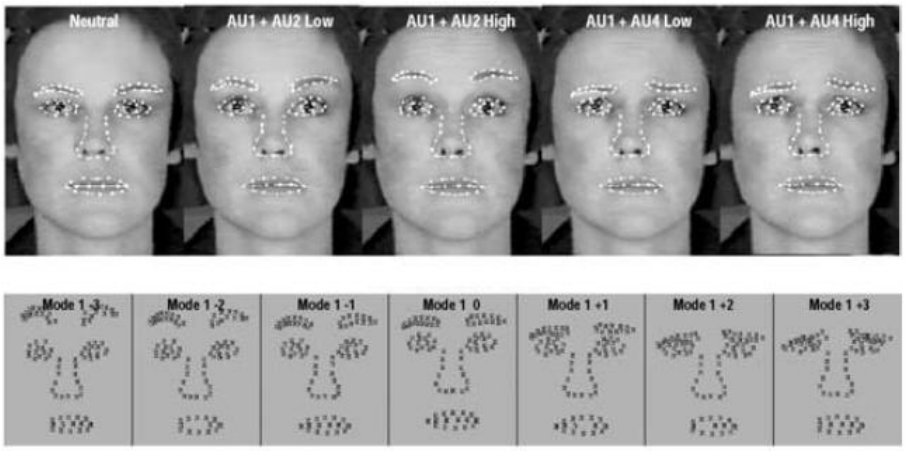


Figure 2.5: ASM: The top row of images are manually labeled for training, the bottom row shows the point distribution model(PDM) learned by PCA [5]

Active shape models (ASM) (see Figure 2.5) proposed by Cootes et al. [35] represent the shape of an object using a number of landmark points, and capture the shape variations of the object with a point distribution model (PDM) which is constructed by principal component analysis (PCA). Active shape models allow to simultaneously determine shape, scale and pose by fitting appropriate point distribution models to

objects of interest. Huang et al. [36] used a point distribution model to represent the shape of a face, where shape parameters were estimated by employing a gradient-based method. Most of ASM-based facial expression approaches use a single ASM face model to represent one type of expressions. Naturally the models learnt from face images in a specific view(eg. frontal view) cannot work on the other views(eg. profile view). Wan et. al [37] extends the ASM to represent faces in different views by modelling the shape of face counters and facial components separately. The basic idea is that the shape of face contours will be much more similar than the individual face components in different views. To represent the facial components, three models are used depending on the face's orientation: frontal view model, left profile view model and right profile view model.

Tian et al. [6] proposed a Multistate Face Component Model to detect and track changes of facial components in near front images, see Figure 2.6. They used a three-state lip model to describe the lip states: open, closed and tightly closed. A two-state model is used for each of the eyes: open or closed. Each brow and cheek has a one-state model. This model is used to represent facial movements in an image sequence by measuring the states transition of corresponding facial components.

In an image sequence, the facial movements can be modeled by measuring the geometrical displacement of facial feature points between the current frame and the initial frame. In [38], 20 facial points were manually selected and the facial movements are represent by features calculated from the tracked facial points. In [39], they extended the approach by proposing a fully automatic facial movement detection system that can automatically localize facial points in the initial frame and recognize the facial movements using the most representative features selected by AdaBoost.

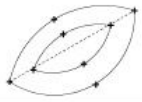
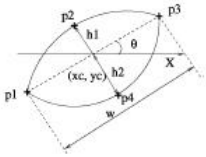
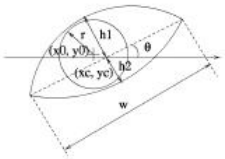
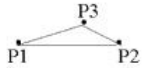
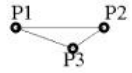
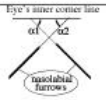
Component	State	Description/Feature
Lip	Open	
	Closed	
	Tightly closed	
Eye	Open	
	Closed	
Brow	Present	
Cheek	Present	
Furrow	Present	
	Absent	

Figure 2.6: Multistate Face Component Model [6]

One may note here that, to extract geometric features, it usually requires accurate and reliable facial feature detection and tracking. The automatic detection and tracking of facial features is still an open problem in many real life situations, and relying on manual labor on such task is very time expensive and error prone. This motivates the use of appearance based features for facial expression analysis.

Appearance Feature Classification

The appearance feature models the appearance change of faces. Holistic spatial analysis including Principal Component Analysis (PCA)[40], Linear Discriminant Analysis (LDA) [41], Independent Component Analysis (ICA) [42], and Gabor wavelet [43] have been applied to the whole face or specific face regions to extract the facial appearance changes. Different techniques are explored by Donato *et al.* [44] to represent face images for facial movement recognition, including PCA, ICA, LDA, Local Feature Analysis and local schemes such as Gabor wavelet representation and local principal components. They reported that best performance can be achieved with Gabor wavelet representation and Independent Component Analysis.

Gabor filters are widely used to extract the facial appearance changes as a set of multiscale and multiorientation coefficients. Ford [45] applied a family of Gabor wavelets at five spatial frequencies and eight orientations to the whole face image. In order to provide robustness to lighting conditions and to image shifts, they used a representation in which the outputs of two Gabor filters in quadrature are squared and summed. This representation is known as Gabor energy filters which models complex cells of the primary visual cortex [46]. The Gabor filters can also be applied to specific locations on a face [4, 30, 47, 48].

Since the computation of Gabor-wavelet representation is both time and memory intensive, Ojala et al. [49] proposed the Local Binary Pattern (LBP) as a computational effective texture description. The original LBP operator labels the pixels of an image by thresholding a 3 x 3 neighborhood of each pixel with the center value and considering the results as a binary number. The derived binary numbers can be used to represent texture primitives, see Figure 2.7. In order to capture dominant features

with large scales, the original LBP operator was later extended to use neighborhood of different sizes [50]. Using circular neighborhoods and bilinearly interpolating the pixel values allow any radius and number of pixels in the neighborhood. The LBP histogram contains information about the distribution of the local micro-patterns, such as edges and spots, over the whole image. Therefore, face images can be effectively represented by LBP histograms as shown in [7, 51, 52]. Shan et al. [53] performed a comprehensive study on facial expression recognition using LBP features. Different machine learning methods are exploited to classify expressions on several databases, including Support Vector Machines (SVM), Linear Discriminant Analysis (LDA) and linear programming. They also tested the LBP features for low-resolution facial expression recognition.

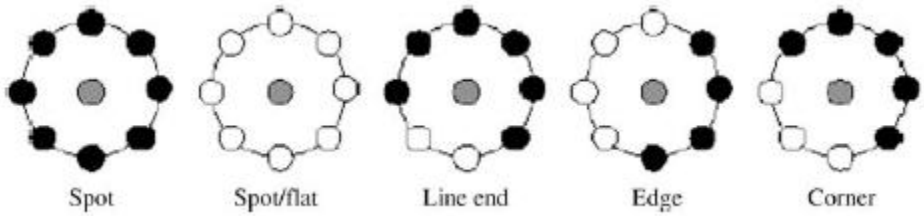


Figure 2.7: Texture primitives represented by LBP [7]

Alternatively, Wang et al. [54] used four Haar-like rectangle features for facial expressions recognition. These features are originally proposed by Viola et al. [2] for face detection. Hence the ability for distinguishing different facial expression is limited. Jung et al. [55] present new types of Haar-like rectangle features that are suitable for facial expression recognition.

2.3 Facial Expression Classification

The facial expression recognition component is responsible for interpreting the extracted facial features and mapping the face images to some pre-defined categories, which is essential a classification task. Different classifiers has been applied to tackle this task, including Neural Network [56], Bayesian Network (BN) [57], Support Vector Machine (SVM) [58], rule-based classifiers [34], Hidden Markov Models(HMM) [59]. The approaches can be divided into two groups: frame based recognition which only relies on a single frame with or without a reference frame; image sequence based approaches exploited the temporal behaviors of facial expressions.

Expression representation

There are generally two models for expression representation: *discrete category model* and *Facial Action Coding System (FACS) model*. As its name suggests, discrete category model describes expressions in terms of discrete categories. The most popular example of this description is the prototypical (basic) emotion categories, including happiness, sadness, surprise, anger, fear, and disgust [9, 60, 61]. All emotion related expression can be described by one of the prototypical expressions and Ekman et al. [9] claim that these six prototypical expressions can be perceived in the same way across all human ethnicities and cultures. A large percent of automatic facial expression analysis systems focus on recognizing these prototypical emotions.

The Facial Action Coding System (FACS) [62] is designed to detect subtle changes in terms of Action Units (AU) in facial features. Any types of facial expression can be represented by a combination of Action Units. The Facial Action Coding System (FACS) is comprised of 44 Action Units, see Figure 2.8 for some example Action

Units. As actions units are independent of interpretation, they can be used for any high level decision making process, such as facial expression recognition.

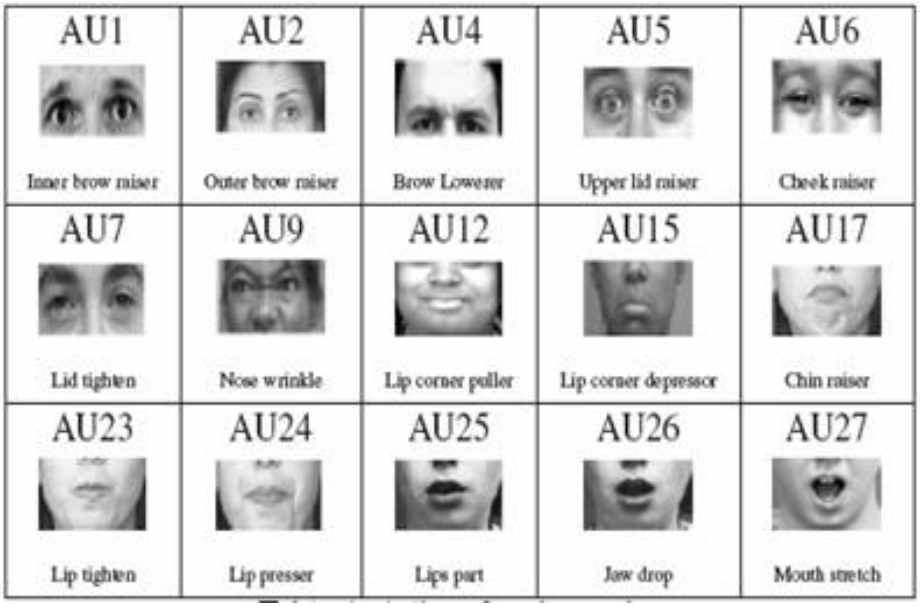


Figure 2.8: Sample Action Units [62]

Frame based recognition

Frame based recognition relies on a single frame with or without a reference frame for expression recognition. In general, any machine learning algorithm is applicable for this task. In [43], the principal components of the feature vectors from training images were analyzed by LDA to form discriminant. A test face image was classified by projecting the input vector of the image along the discriminant vectors. The proposed method was trained and tested on JAFFE [43] dataset. The recognition rate was reported to be 92%.

Bartlett et al. [58] performed systematic comparison of different techniques including AdaBoost, SVM and LDA for facial expression recognition. Best results were obtained by selecting a subset of Gabor filters using AdaBoost and training SVM on the outputs of the selected filters. Shan et al. [53] compared the different learning algorithms using the LBP features and achieved the best performance with SVM. LBP features were further compared to Gabor wavelet features using SVM on the Cohn-Kanade [63] dataset, and they reported that the recognition rate of LBP + SVM was slightly higher than Gabor + SVM.

Tian et al. [6] proposed to use a three-layer neural network with one hidden layer to recognize Action Units. Separate networks are used for the upper and lower face. When Action Units are occur in combination, multiple output nodes are excited. Pantic and Rothkrantz [64] proposed to use rule-based reasoning to recognize action units and their combinations.

Image Sequence based recognition

Hidden Markov Models (HMMs) have been widely used to model the temporal information. Cohen et al. [65] proposed a multilevel HMMs classifier for recognizing facial expression and segmenting long image sequence to different expression segments. The first level is comprised of independent HMMs related to different emotions. The output state sequence is used as the input of the higher level HMM which enables the segmentation.

Dynamic Bayesian Networks (DBN) have also been exploited for image sequence based expression recognition [32, 66]. Kaliouby and Robinson [32] propose to use multi-level DBN classifier to model complex mental states as a number of interacting facial and head displays. Zhang and Ji [66] propose to use DBN and multi-sensory

information fusion technique to model the temporal information of facial expression in image sequences.

Lee et al. [67] proposed a framework to learn a decomposable generative model to represent and analysis facial motions. The learned model can generate different dynamic facial appearance for different people and for different emotions.

2.4 Summary

As disscused in this chpater, numerous approaches to automatic facial expression recognition have been proposed. However, only relatively few recent studies consider the problem in unconstrained environments, and the problem has remained unsolved for the most part. As one major problem faced by the litearture is the lack of realistic training and testing data, we tackle the problem by starting developing methods that are able to construct large scale facial expression image datset form web images with minimum human efforts. Various novel facial expression features are then proposed to address the challenges imposed by the newly collected dataset. Finally, a feature cmbination method is propsed to combine the proposed facial expression features to form a more descriptive representation.

CHAPTER 3

HARVESTING WEB IMAGES FOR REALISTIC FACIAL EXPRESSION RECOGNITION

3.1 Introduction

Facial expression image dataset is essential for the research of automatic facial expression analysis, as it is required to learn facial expression models and to evaluate recognition algorithms. Over the past decades, several facial expression datasets such as CK [63, 68] and JAFFE [43] have been made available to the public [4, 64]. They have played a significant role to address the issues of lacking facial expression data. However, these datasets are generally of a small scale comprising mostly of photos collected under controlled environments from a very small number of human subjects, which results in the fact that they are not ideal for evaluating expression recognition algorithms in realistic settings due to the following two reasons:

- They suffer from common limitations of being small in terms of both the number of subjects and images and fail to represent the diverse set of variations found in the real world [69]. Consequently, despite the promising performance reported [53], facial expression recognition algorithms trained on such datasets cannot be applied in practice.

- The datasets are not sufficiently challenging to unveil the capabilities of comprehensive facial expression recognition algorithms. For example, some rudimentary algorithms are able to match the performance of many state-of-art systems on these datasets due to the well controlled data acquisition environments [51]. The benefits of utilizing a more advanced algorithm may not be thoroughly studied.

There is a clear demand for a comprehensive large-scale facial expression dataset that is collected under real world conditions [8, 69]. Ideally, the data need to be collected manually to ensure the quality. However, obtaining such data is a tedious and time-consuming task that requires tremendous efforts which are proportional to the dataset size. Nowadays, the prosperity of the Internet and Web technologies brings us a large quantity of web images containing faces. These face images are taken by people all over the world, hence typically span a wide range of image settings and cover a large number of human subjects with different ages, genders and ethnic groups. Moreover, a large number of web images are associated with related textual descriptions, such as surrounding texts, title, and URL. Together these form a solid foundation on which high quality facial expression image datasets can be obtained with limited user supervisions.

Motivated by the above observations, in this chapter we address the demand on realistic facial expression image datasets and propose a search based framework to harvest realistic facial expression images from the Web. Specifically, our framework consists of two major components as shown in Fig. 3.1: initial keyword based search and active learning based refinement of search results. Firstly, given emotional keywords corresponding to one facial expression, we use image search engines to obtain a

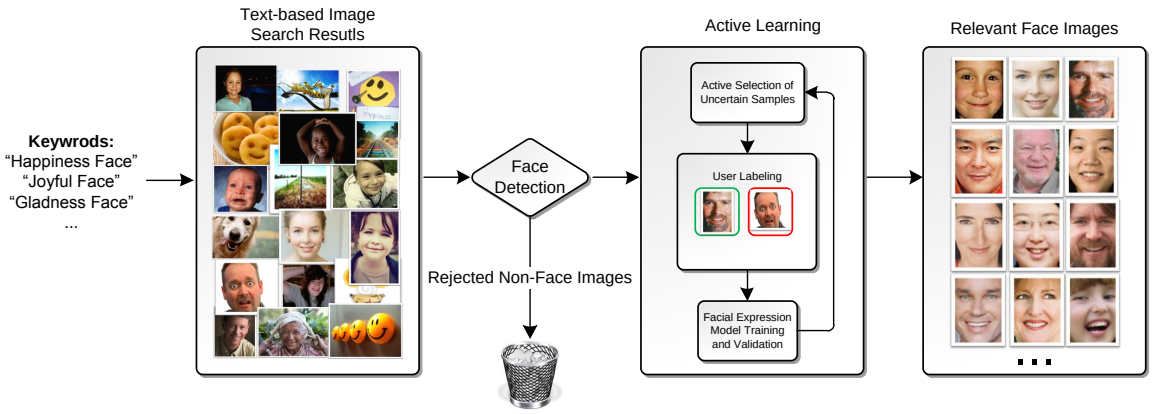


Figure 3.1: Overview of the active learning based framework for constructing facial expression datasets.

raw dataset S_{raw} consisting of images that are potentially relevant to the expression. In our experiment, Google is used for simplicity. However, it should be noted that our framework is flexible for using multiple and differnt image search engines. As most of the commercial search engines handle image search based on text analysis while ignoring the actual visual content, S_{raw} is too noisy to be used directly for training the facial expression recognition methods. Hence rather than treat all images in S_{raw} as positive samples for the class of interest, we secondly use a binary Support Vector Machine (SVM) classifier as a post-detector to select images visually relevant to the query expression (keyword). The SVM classifier is learned from a training set that is constructed by pool-based active learning [70]. The goal of the classifier is, for each input image, to predict the presence of the facial expression of interest. Since the active learning algorithm selects examples to be labeled, it requires much less human effort during the classifier training. The final facial expression dataset is composed of images selected by the SVM classifier.

The newly collected facial expression dataset is very challenging for the Local Binary Pattern (LBP) [53] and Weber Local Descriptor (WLD) [71] based facial expression descriptors due to the level of variation in the data. Therefore, we propose a novel facial expression feature based on WLD and histogram contextualization [72] for multiscale analysis of facial expressions.

In summary, the main contributions are as follows:

1. We propose a web search based framework to build a realistic facial expression dataset from the Web that contains a diverse set of human subjects and imaging environments. Our approach is designed to minimize the human efforts during the dataset construction and to maximize the scalability of the dataset for future research. To the best of our knowledge, there are very limited studies in the literature about building general facial expression datasets from web images.

2. We adopt an active learning based method to remove noisy images from the search results returned by commercial search engines.

3. We propose an efficient facial expression feature based on the recent WLD descriptor for multiscale analysis of faces, namely Multiscale-WLD. In addition, spatial context is taken into account while histogram features are formed. Various experimental results demonstrate that Multiscale-WLD is more robust than other widely used descriptors such as conventional WLD and LBP.

4. We conduct comprehensive experiments to demonstrate that our facial expression dataset outperform other existing datasets in terms of generalization capabilities.

The rest of the chapter is organized as the follows. Section 3.2 briefly reviews the related works. Sections 3.3 and 3.4 explain in detail the text based image collection and content based image refinement approaches. Section 3.5 presents our Multiscale-WLD based facial expression feature. Section 3.6 discusses the experimental settings and results. Finally, concluding remarks are addressed in Section 3.7.

3.2 Related Work

In this section, we review the relevant literature on image understanding and facial expression image feature extraction.

3.2.1 Image Understanding from Web Images

Training data acquisition is a key challenge in the development of large scale computer vision and pattern recognition applications. In most cases, the cost of manual data labeling is prohibitive due to the amount and range of the data. Therefore in the recent years, there have been emerging interests in harvesting data or mining knowledge from the Web.

A few approaches in this domain are related to our framework. Fergus *et al*. [73] utilized images returned by Google image search to learn object categories automatically. In their work, images were modeled as a mixtures of latent topics, which were learned from Google image search results by an extended probabilistic Latent Semantic Analysis (pLSA) model. Then top ranked image search results were used to select a subset of topics corresponding to the object category of interest, and an object classifier was built using these topics. However its performance might be strongly affected by the accuracy of top ranked Google image search results, as it relies on these images to select representative latent topics for each category. Li *et al*. [74] also

aimed to collect an object category dataset from images returned by search engines and learn the object category models simultaneously. They employed a Hierarchical Dirichlet Process (HDP) model that was learned via an incremental learning process, which gave their framework ability to incorporate novel images without being fully re-trained. Hence the approach was more scalable than the work of Fergus *et al.* [73]. Similarly, Collins *et al.* [75] applied a boosting based classifier to select relevant images from image search results. To minimize the required number of supervised training examples, the classifier utilized active learning and online learning to update its model during the training process.

While the above works focus on utilizing the visual information, some other works also exploit the meta data surrounding the web images. Berg and Forsyth [76] aimed to build an animal dataset using Google search. They applied Latent Dirichlet Allocation (LDA) model to learn a set of latent topics from surrounding texts of images returned by Google image search, and select visual exemplars (images) for each topic based on the nearby words of the images. Visual information obtained from visual exemplars was then incorporated with textual information to build a classifier, which can be used to determine whether a new image depicts an animal. This method requires users to label visual exemplars as relevant or background. To completely automate the data collection and cleaning tasks, Schroff *et al.* [77] proposed to use a simple Bayesian posterior estimation to re-rank the image search results based on textual information. The top ranked images were selected to build a training set. A SVM classifier, learned from the training set, was adopted to re-rank the image search results based on the visual information. Other recent work aiming at building datasets from web images require manual data collection and cleaning [78, 79].

As in the domain of facial expression analysis, to our best knowledge, there exists only two such datasets, GENKI dataset [69] and Static Facial Expressions In The Wild (SFEW) dataset [80]. The GENKI dataset was developed for smile detection, therefore it only provides the ground-truth data for one type of facial expression. Also the images had to be manually labeled. These limitations make the GENKI dataset impractical for the research of general purpose facial expression analysis. The SFEW dataset [80] was collected from movies and covered six basic expressions *angry*, *disgust*, *fear*, *happy*, *sad*, *surprise* and *neutral*. However, it also required human annotators to manually label all the data. To address this issue, here we attempt to construct a semi-automatic framework by harvesting facial expression images from the Web that can be scaled to support diverse types of facial expressions.

3.2.2 Facial Feature Extraction

Face representation has been studied intensively for automatic expression recognition over the past decades, and a variety of approaches have been presented. In general, these approaches can be divided into two groups: geometric based approaches [6, 30, 38, 39] and appearance based approaches [42, 53, 81, 82, 83].

Geometric based facial expression feature extracts the shape and locations of facial components to represent the face geometry. In an early work by Zhang *et al.* [30], 34 fiducial points were utilized to represent a face image. The fiducial points were manually selected at facial landmarks and the image coordinates of these points were used as features, which results in a 68-dimension feature vector. Tian *et al.* [6] proposed a Multistate Face Component Model to detect and track changes of facial components in near frontal face images. This model represented facial movements

by measuring the state transitions of corresponding facial components. In an image sequence, the facial movements could be modeled by measuring the geometrical displacement of facial feature points between the current frame and the initial frame. Valstar *et al*. [38] manually selected 20 facial points and recognized Facial Action Units (AUs) by classifying features calculated from tracked facial points. Their experiments demonstrated that the facial representation based on tracked facial points was well suited for facial expression analysis. This approach was further extended by adopting a fully automatic facial movement detection system that could automatically localize facial points in the initial frame and recognize the facial movements using the most representative features selected by AdaBoost [39]. However, extracting geometric features usually requires accurate and reliable facial feature detection and tracking. The automatic detection and tracking of facial features is still an open problem in many real world situations, and relies on manual labor which is very time expensive and error prone. Therefore, appearance based features for facial expression analysis have also been investigated.

Appearance based facial expression features model the appearance change of faces, such as wrinkles and furrows, by directly utilizing pixel values. Holistic spatial analysis including Principal Component Analysis (PCA) [40, 84], Linear Discriminant Analysis (LDA) [41], and Independent Component Analysis (ICA) [42] has been applied to the whole face or specific face regions to extract the facial appearance changes. Typically these methods project face images onto a subspace, find a set of basis images, and represent faces as a linear combination of those basis images. The Active Appearance Models (AAM) [85] was also applied to facial expression recognition,

which used PCA to model both shape and texture variations. The models can be fitted to new images by varying the shape and texture parameters within limits learned from a training set. Abboud *et al.* [86] applied LDA to the AAM parameters to obtain the most discriminative features and represent facial expression images. Sung *et al.* [87] combined the AAM with Active Shape Models (ASM) [35] to reduce the average model fitting errors. Ashraf *et al.* [88] utilized AAM derived representations for recognizing facial expression of pain.

In recent years, researchers have turned toward local descriptor based facial expression features as local descriptors have been shown to be more robust to occlusion, misalignment and moderate pose changes than traditional holistic methods [81, 82, 83, 89]. Ojala *et al.* [49] proposed the Local Binary Pattern (LBP) as a computational effective texture descriptor. The original LBP operator labels the pixels of an image by thresholding a 3×3 neighborhood of each pixel with the center value and considering the results as a binary number, and use the derived binary number to represent texture primitives. In order to capture dominant features at a large scale, the original LBP operator was later extended to use neighborhood of different sizes. Using circular neighborhoods and bi-linearly interpolating the pixel values allow any radius and number of pixels in the neighborhood [50]. The LBP histogram contains information about the distribution of the local micro-patterns. Thus face images can be effectively represented by LBP histograms as shown in these works [53, 90]. Shan *et al.* [53] performed a comprehensive study on facial expression recognition using LBP features. Different machine learning methods have been employed to classify expressions on several datasets, including SVM, Linear Discriminant Analysis (LDA) and linear programming. It is argued that LBP is more robust and efficient than

Gabor wavelet features. By combining Gabor filtering with LBP, Local Gabor Phase Patterns (LGBP) [66] was proposed to extended LBP to multiple resolutions and orientations. Xie *et al*. proposed Local Gabor XOR Pattern [91] to exploit the Gabor phase information. Recently, Chen *et al*. [71] proposed the Weber Local Descriptor (WLD) for face detection. WLD characterizes texture information of an image by considering the ratio of changes in pixel intensity. Different to LBP, it uses the gradient orientations to describe the direction of edges. In addition to LBP and WLD, some other local descriptors originally proposed for object recognition tasks were also used for facial representation, such as SIFT [92], Histogram of Oriented Gradients (HOG) [93], and Haar-like rectangle features [54].

3.3 Text-based Web Image Search

In this section, we describe the procedures for collecting the initial pool of weakly labeled images from the Web. We rely on text based image search engines to obtain the initial image set which is motivated by the fact that the human face related images are usually accompanied with emotional text information. Although it is often the case that the retrieved images are not semantically relevant to the query keyword, there still exist a considerable number of images with correct information on facial expressions. For the scope of this chapter, interests are placed on the seven basic facial expression categories including *happiness*, *sadness*, *surprise*, *fear*, *disgust*, *anger* and *neutral* [94]. The initial list of query keywords are intuitively formed by these category names.

Most of image search engines restrict the number of images to be returned. For example, Google only returns the top 1,000 images for each query, in which there also

exist quite a number of dead links. If one keyword is used for one facial expression, we can only collect a very limited number of training images. In order to overcome this restriction, we formulate emotionally related text queries using an affective-based lexical datasets - WordNet-Affect [95] which models the affective words or synsets in a hierarchical structure. Under a category parent in the hierarchy, the affective words are semantically and emotionally similar. For example, *joy*, *gladness*, and *cheerfulness* are affective words under a same category, and share very similar affective meaning of *happiness*. Hence these affective words are also included in the query keywords list as shown in Table 3.1. To further increase the number of potential training images, the queries can also be performed on different image search engines and online image sharing web sites in the future. Since only the human face related images are of our interest, the above query keywords have been expanded with face related terms such as *face* and *expression*. As a result, a *raw* dataset, denoted as S_{raw}, is created from straightforward web image search.

Table 3.1: A list of keywords used for text based facial expression image search.

Original Keyword	**Extended Keywords**
Anger	fury, infuriation, umbrage, indignation, annoyance, huffiness, dander
Disgust	shame, dislike, repugnance, nausea
Fear	scare, panic, horror, creeps, apprehension
Happiness	joy, amusement, gladness, rejoicing, cheerfulness, exhilaration, elation
Sadness	cheerlessness, sorrow, misery, weepiness, depression, forlornness, melancholy
Surprise	astonishment, amazement, shock

3.4 Content-based Refinement

3.4.1 Face Registration

Despite the efforts described in Section 3.3, many images in S_{raw} still do not contain any face or are of low quality for training purpose (*e.g.* lack of frontal face). The well known Viola-Jones face detector [2] is utilized to remove these noisy images.

After faces are detected, we firstly perform automatic eye localization on the detected face region in order to align different face image data into a common coordinate system based on eye locations. For this task, we adopt the Average of Synthetic Exact Filters (ASEF) [96] which is a class of correlation filters. Then we align and normalize the faces based on the detected eye locations and the distance between the two eyes. Similar to the practice of Shan *et al.* [53], facial images of 110×115 pixels are cropped from the original frames and are used to construct the initial facial expression dataset denoted as S_{face}.

3.4.2 Active Learning based Refinement

The face registration process described in Section 3.4.1 removes most of images not containing frontal faces. However, a fairly large proportion of the remaining face images are still not related to the query expression. For example, the search results may contain *sad* face images for the query of *happiness*. In order further improve the quality of S_{face}, we have to further select images that are semantically relevant to the query expression. Thus for each category of facial expression of interest, we apply a binary SVM classifier that is learned from a training set constructed by a pool-based active learning method.

Support Vector Machine

Given a training dataset $D = \{(x_i, y_i)\}_{i=1}^{l}$, where l is the size of the dataset, $x_i \in R^n$ is a feature vector and $y_i \in \{-1, +1\}$ is the class label of x_i. Support Vector Machine (SVM) finds a hyperplane in the form of $w \cdot x + b = 0$ that maximizes the margin of separation between two data classes -1 and $+1$. Maximizing the margin is a problem of constrained optimization that can be solved by Lagrange method. Thus w can be solved as [97]:

$$w = \sum_{i=1}^{l} \alpha_i y_i x_i, \tag{3.1}$$

where the coefficients α_i is a Lagrange multiplier. A non-zero α_i indicates that x_i associated with α_i is a support vector. New data point x_{new} can now be classified by the decision function:

$$f(x_{new}) = sign\left(\sum_{i=1}^{l} \alpha_i y_i (x_i \cdot x_{new}) + b\right), \tag{3.2}$$

where positive (negative) utput means x_{new} belongs to the positive (negative) class.

It is often the case that two classes are not linearly separable, therefore SVM uses a kernel function to perform non-linear mapping of the data into a higher dimensional feature space, and finds a linear separating hyperplane with the maximum margin to separate the data in this high dimensional space. The radial basis function (RBF) kernel given in Eq. 3.3 is used in our experiments.

$$k(x, y) = exp(-\gamma \|x - y\|^2) \tag{3.3}$$

And the decision function in Eq. 3.2 can be rewritten to:

$$f(x_{new}) = sign\left(\sum_{i=1}^{l} \alpha_i y_i k(x_i, x_{new}) + b\right), \tag{3.4}$$

where the dot product is substituted by the kernel function $k(x_i, x_{new})$. In our experiment, grid searching is used to find the optimal parameters of SVM.

Pool-based Active Learning with SVM

Active learning aims to reduce the number of labeled examples required to train a classifier. This is achieved by selecting the most informative unlabeled examples to require human labeling. The key challenge here is how to select the next unlabeled instances to interact with users. Following the settings in [98], we use an uncertainty sampling based approach that chooses the unlabeled examples with least classification certainty. Though SVM does not give the probability of prediction directly, the probability can be estimated by using a Sigmoid function [99]. In a binary classification problem, the method is equivalent to find the data points with the smallest margin to the decision hyperplane.

Given a seed training dataset D that contains labeled expression images, a unlabeled dataset P, and a validation dataset V. Initially, D only contains a small number of randomly selected examples. The workflow of the algorithm is as follows:

1. Train an SVM classifier with D;

2. Perform the classification on P and compute the class membership probability estimates;

3. Remove the images with the lowest classification certainty to query the user, and add the actively selected images along with user provided labels to D. In our experiments, five images are returned to the user for manual labeling;

4. Evaluate the model by performing classification on the manually labled validation dataset V;

5. Go back to step (1) and repeat until the user is satisfied or the images have been exhausted.

3.5 Multiscale-WLD Based Facial Expression Feature

Previous studies have shown that facial images can be effectively described as a composition of micro-texture patterns, such as edges, spots and flat areas [53, 100]. Hence in this work, we propose to represent a facial image by its local textures and the spatial layout of the textures. The spatial layout is captured by partitioning a facial image into grids (as shown in Fig. 3.3) and each grid is represented with WLD to capture its local texture. The local descriptors are then concatenated to form a global description of the face. In the following sub-sections, we will describe our facial feature in detail.

3.5.1 Weber Local Descriptor

The Weber's law [101] states that the smallest change in the intensity of a stimulus capable of being perceived is proportional to the intensity of the original stimulus. This implies that the ratio of the change in the intensity of the stimulus reflects the magnitude of human perception of the stimulus. Based on such motivation, the Weber Local Descriptor (WLD) was recently developed by Chen *et al.* [71] to characterize texture information of an image by considering the ratio of changes in pixel intensity which can be considered as stimulus information for visual perception.

WLD is comprised of two components, differential excitation and orientation. Differential excitation $\xi(x_c)$ measures the ratio of change in pixel intensity between a center pixel x_c against its neighbors. It is computed in the following way [71]:

$$\xi(x_c) = \arctan\left[\sum_{i=0}^{p-1}\left(\frac{x_i - x_c}{x_c}\right)\right], \tag{3.5}$$

where x_i denotes the ith neighbors of x_c and p is the number of neighbors (8 in the case of 3×3 neighborhoods). The arctan function is applied to smooth out the results. Differential excitation $\xi(x_c)$ captures the local salient visual patterns. For example, a high $\xi(x_c)$ value indicates that x_c potentially belongs to an edge or a spot as there is a strong difference in pixel intensity between x_c and its neighbors.

The orientation component $\theta(x_c)$ of WLD is the gradient orientation of the pixel x_c. It is computed as [71]:

$$\theta(x_c) = \arctan\left(\frac{x_7 - x_3}{x_5 - x_1}\right), \tag{3.6}$$

where x_1, x_3, x_5 and x_7 are neighbors pixels of x_c as shown in Fig. 3.2. The orientation component is then quantized into T dominant orientations.

After labeling the image with WLD, a $2D$ WLD histogram of the labeled image can be defined as:

$$H_{wld}(c,t) = \sum_{i=0}^{I-1}\sum_{j=0}^{J-1} I\Big(\xi\left(x_{i,j}\right) = c\Big) I\Big(\theta\left(x_{i,j}\right) = t\Big), c \in C, t \in T \tag{3.7}$$

where $I \times J$ is the dimensionality of the image, $x_{i,j}$ is the pixel at location (i,j) in the image coordinates, T is the number of dominant orientations as mentioned above, C is the number of bins of the differential excitation histogram in each dominant orientation θ_t, and

$$I(A) = \begin{cases} 1 & \text{if A is true,} \\ 0 & \text{otherwise.} \end{cases} \tag{3.8}$$

x_0	x_1	x_2
x_7	x_c	x_3
x_6	x_5	x_4

Figure 3.2: Illustration of neighborhood pixels used for extracting the WLD descriptor of x_c.

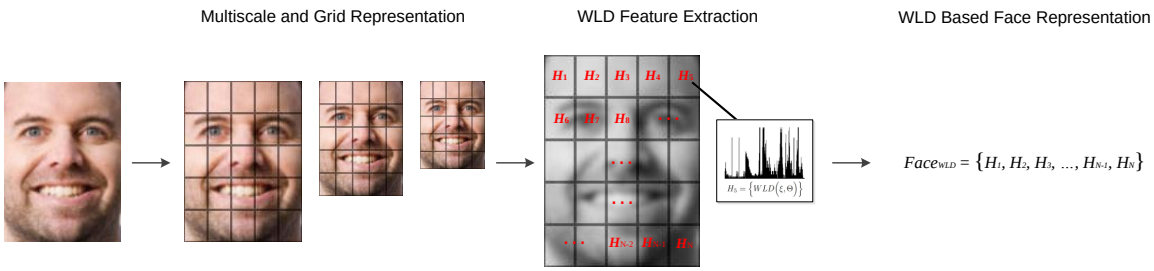


Figure 3.3: Multiscale-WLD Facial Expression Feature.

Note that in this $2D$ histogram, each column corresponds to a dominant orientation, and each cell $H_{wld}(c,t)$ corresponds to the frequency of a certain differential excitation interval on a dominant orientation θ_t. The size of the interval is controlled by two user defined parameters M and S. The $2D$ histogram is further encoded into a 1D histogram by concatenating all of the cells $H_{wld}(c,t)$. Therefore the size of the final descriptor is $T \times M \times S$.

3.5.2 Multiscale-WLD Based Face Representation

As faces can be seen as a composition of micro texture patterns [53, 100], it is intuitive to use WLD to represent facial images. However, a WLD histogram computed over a global face image does not capture the spatial locations of the micro texture patterns since the patterns tend to be averaged over the whole image area which will reduce the discriminative power of the WLD descriptor for facial expression recognition. Hence in order to overcome this issue, we divide a face image equally into N rectangular regions $R_1, R_2, ..., R_N$ (Fig. 3.3), and a histogram $H_n (n = 1, 2, ..., N)$ is computed independently for each sub-region.

When perceiving facial expressions, human beings pay more attention to some face regions (*e.g.* eyes and mouth) than others [53]. In order to take this observation into account, each region can be assigned a weight $w_n (n = 1, 2, ..., N)$ according to the importance of the region in human perception. The resulting N histograms are then concatenated into a single spatially enhanced histogram:

$$H = \Big\{ w_n H_n \Big\}, \text{ where } n = 1, 2, ..., N. \tag{3.9}$$

The histogram H effectively encodes both the local texture appearance and the global spatial relationships among facial regions.

Multiscale analysis is achieved by down-sampling the original face image to form an image pyramid followed by applying a WLD operator with fixed neighborhood size of 3×3 pixels. A spatially enhanced histogram is computed at each level of the pyramid. The final Multiscale-WLD feature vector is formulated by concatenating the histograms extracted at different scales. Since encoding not only the micro structures of a face image but also the macro structures which provides a more extensive

description than the basic WLD operator, the Multiscale-WLD is more robust. The distance between two Multiscale-WLDs reflects the extent to which two facial images contain similar textures within corresponding spatial regions.

In summary, our Multiscale-WLD based facial expression feature is formed as follows:

$$H_{M-WLD} = \Big\{H_1, H_2, ..., H_{R-1}, H_R\Big\}, \tag{3.10}$$

where R is the number of different scales and H_r is a spatially enhanced histogram computed using (3.9). The dimension of the Multiscale-WLD based descriptor equals to $R \times N \times M \times T \times S$, where R denotes the number of scales that a face image will be analyzed at, N denotes the number of rectangular regions that a face image will be divided into, M, T, and S are WLD parameters, in which T determines the number of dominant orientations of the WLD orientation components, and M and S control the size of the interval of the WLD differential excitation components on a certain dominant orientation.

3.5.3 Contextualized Multiscale-WLD Based Face Representation

As a histogram based feature, the Multiscale-WLD based face representation lacks the ability to capture the spatial contextual information among the patterns. In order to address this issue, we extend the Multiscale-WLD by utilizing contextualized histogram [72] to encode spatial context in our face feature. The contextualized histogram CH is constructed by comparing an image indexed by the bins of a histogram H with a set of predefined structures, followed by counting the occurrence of different structures for each bin of H. As in [72], 30 predefined structures are used in this chapter where each structure defines a homogeneity pattern and a different shape.

Hence, the dimension of the contextualized Multiscale-WLD is becoming to 30 times of that of the original Multiscale-WLD descriptor. PCA is then applied to reduce the dimensionality for better classification performance.

3.6 Experiments and Discussions

3.6.1 Experimental Settings

Our experiments were conducted on 7 categories of universal facial expressions [94]: *happiness*, *sadness*, *anger*, *fear*, *disgust*, *surprise* and *neutral*. After removing non-face images, the number of images in each category varies from 2000 to 2500 images in image set S_{face}. For each category, a facial expression classifier is trained from S_{face} using SVM with active learning where the following three datasets were utilized:

1. Validation Set G_v: This will be used as the verification dataset to determine the stopping criteria for active learning, as well as a *realistic* facial expression image dataset to be compared with other well established facial expression datasets in Section 3.6.3. In total, 350 images (50 images for each of the 7 categories) are randomly drawn from the initial face dataset S_{face} and manually labeled.

2. Seed Training Sets: Seed training sets are the initial training data used in active learning (see Section 3.4.2). Each facial expression category has a seed training set. The positive examples are manually labeled and the negative examples are uniformly-randomly drawn from other seed positive training sets. The number of positive examples and negative examples are equal, which is set to 20 images for each category in our experiment.

3. Active Learning Pools: It is comprised of all remaining images in S_{face}, one for each category of facial expressions. The number of images in each category varies from 1900 to 2400.

In our experiments, the facial expression images are represented by Multiscale-WLD features. We denote the Multiscale-WLD feature as $M-WLD_{R,N}^{M,T,S}$. Here, N is experimentally set to 5×5, R is set to 3 and each face image is down-sampled at scale $\sigma = 0.6$, M is set to 2, T is set to 6, and S is set to 4. The weight value for each region is empirically set according to our observation. The weighting scheme is symmetric with respect to the center y-axis as shown in Fig. 3.4. This yields a 3600-dimensional feature vector for each face image. In Section 3.6.7, we study the impact of different parameter settings. The dimension for Contextualized Multiscale-WLD facial feature (CM-WLD) is 10800, and PCA is applied to reduce the dimension to 400. The effectiveness of CM-WLD is analyzed in Section 3.6.6.

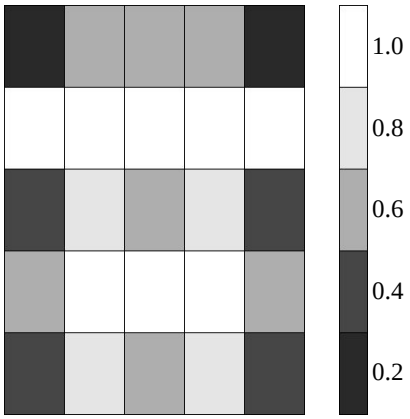


Figure 3.4: The weighting scheme used in our experiments.

We also compare the proposed Multiscale-WLD facial feature with LBP descriptor, which has been widely used in facial expression recognition. Following [90], we use the 59-bin $LBP^{u2}_{8,2}$ operator. Similar to WLD, each facial image is segmented into a grid of 5×5 regions. We compute a $LBP^{u2}_{8,2}$ operator for each of the 25 regions, yielding a 1475-dimensional feature vector (59×25) for each face image. Multiscale-LBP is realized by down-sampling the images into three resolution at scale $\sigma = 0.6$ which gives a 4425-dimensional feature vector ($3 \times 59 \times 25$).

3.6.2 Quality of Our Dataset

In order to evaluate the quality of the data collected with our method, we produce a facial expression dataset G_w. Specifically, we apply the *actively* learned facial expression classifiers to Google image search results, and select the top-ranked 100 images for each expression according to the estimated classification certainty. Some true and false positive images from the resulting dataset are shown in Fig. 3.5. The precision of the 100 images for each expression is compared with that of the top 100 images returned by Google image search. As can be seen in Fig. 3.6, our dataset significantly outperforms Google image search for all 7 categories of facial expressions.

3.6.3 Diversity of Our Dataset

We demonstrate the diversity of our dataset by comparing the performance of facial expression recognition algorithms trained with our dataset to the performance of the recognition algorithms trained with other well established facial expression datasets, JAFFE[4] and Cohn-Kanade DFAT (CK) [63]. For this purpose, we use the actively collected facial expression dataset G_w (see Section 3.6.1) as our *realistic*

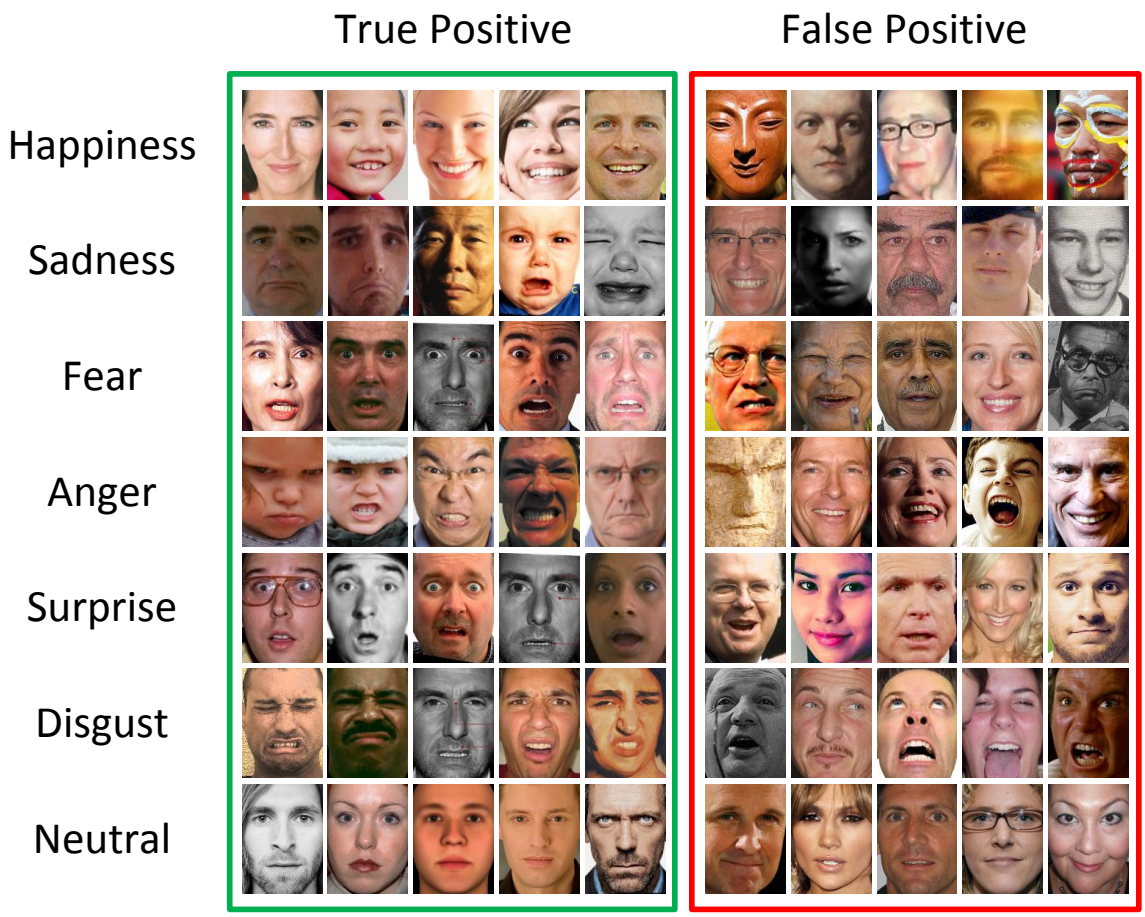


Figure 3.5: Images from our actively collected facial expression dataset G_w.

training dataset and G_v as our *realistic* verification dataset. The JAFFE dataset contains 213 images of the seven basic facial expressions which were posed by 10 Japanese females. It is the most trivial dataset over the three, and serves as the baseline in the experiments. The Cohn-Kanade DFAT dataset consists of approximately 500 images from 100 subjects ranged in age from 18 to 30 years, of which 65% are female. The distribution of the ethnic groups is: 81% Euro-American, 13% Afro-American and 6% other groups. An overview of the evaluation datasets is shown in Table 3.2. All

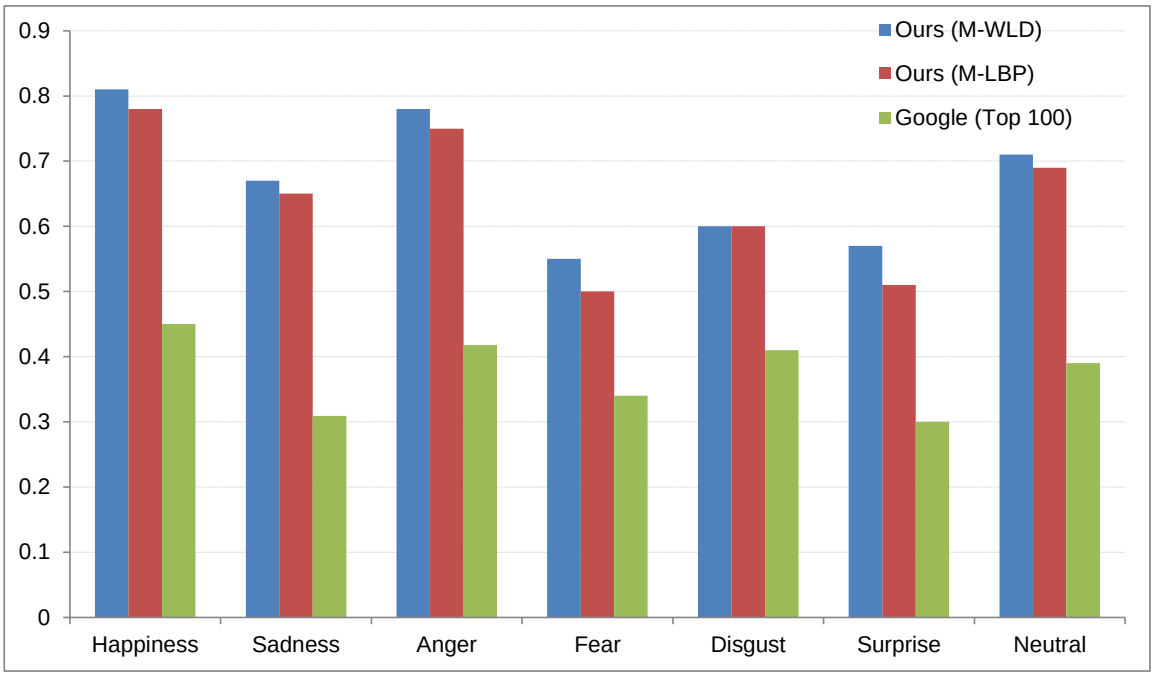


Figure 3.6: Precision of images in G_w (blue and red, the left two bars) compared to first 100 Google image search results (green, the right bar).

face images are resized to a fixed size of 110 x 150 pixels. Histogram equalization is performed to remove the illumination effect in the images.

Here we provide a walk through of the facial expression recognition algorithm used for evaluating the facial expression datasets. Given a set of training images, we first extract the local texture features to represent the facial expressions. Then we train an SVM classifier for each facial expression. Since SVM was originally developed for binary classification, in order to extend SVM for multi-class classification, we use the One-Versus-All approach, which trains a binary classifier to classify one class of interest (positive) versus all other classes (negative). These independent SVM classifiers are used to provide seven predictions of the presence or absence of the facial

Table 3.2: Statistics of the evaluation datasets used in the experiments. The columns of Male and female show the percentage of male subjects and female subjects respectively.

	Images	Subjects	Female	Male
G_v	350	328	59.2%	40.8%
$JAFFE$	213	10	100%	0%
CK	500	100	65%	35%

expression in unseen face images and the class with the greatest class-membership probability estimation value is output as the recognized facial expression.

In our experiments, we perform the 5-fold cross validation for CK and JAFFE. For our dataset, the classifiers are trained on G_w and evaluated on G_v. The confusion matrix of the classification results are reported in Tables 3.3, 3.4 and 3.5, where each row represents a set of images corresponding to a type of expression and each column represents the percentage of the images that is classified into a type of expression. It is observed that the results have some key similarities across the three datasets, but also some interesting differences. The *happiness* and *neutral* expressions are consistently better recognized than the rest types of the expressions. However they are mostly misclassified into each other which suggests that these two expressions share some appearances to a certain extent. It is also noted that the classifiers are mostly confused by the following three types of expressions: *sadness*, *fear* and *anger*. Interestingly, these three types of expressions are relatively difficult to be distinguished by human beings [102]. Another observation is that the *disgust* and *surprise* expressions are comparatively well classified on the CK and JAFFE datasets, but not so well on

our dataset. It is believed that these two expressions are over exaggerated in the posed facial expression datasets (CK and JAFFE). In a real world environment which our dataset G_w is trying to resemble, the difference among them is subtle, which makes them difficult to be well recognized. Finally, it is noted that there is a strong connection between G_w and the classification performance. The better classification results are achieved when there is less noise in the training dataset.

The cross-dataset classification results are reported in Table 3.6. As can be seen, the recognition algorithm performs well for the JAFFE and CK dataset with over 90% classification accuracy rate, but the performance is not ideal on our dataset G_w. This suggests that G_w is much more challenging to the facial expression recognizer compared to the other two datasets since it is collected in much more diverse imaging conditions and contains a much larger variety of subjects (*e.g.* 350 face images from 328 different people). We then perform the cross dataset experiments. Specifically, we train the classifiers using the Multiscale-WLD features obtained from face images belong to one dataset and test the classifiers on the other two datasets. As shown in Table 3.6, the recognition performance of the classifiers trained with our web facial images does not vary significantly across different datasets. Because we preprocess the images in the same way, the only difference between them is that they were collected under different controlled environments. The poor generalization ability of JAFEE and CK datasets suggests that the facial expression classifiers trained on a dataset with uniformly controlled environment only works well for the same dataset. On the other hand, it shows that our dataset is much more diverse compared to JAFFE and CK as the decrease of classification performance is minimal for our dataset. Therefore, it is more difficult to train the classifiers with our dataset, which may explain that

the classification performance are similar for all three datasets even though JAFFE and CK datasets are relatively easy to handle compared to G_w.

Table 3.3: Confusion matrix for the classification results on our G_w. The classifiers are trained on G_w and evaluated on G_v.

	Anger	Disgust	Fear	Happiness	Neutral	Sadness	Surprise
Anger	**51.2%**	13.5%	10.7%	6.5%	0.0%	8.0%	10.2%
Disgust	17.9%	**43.8%**	8.9%	6.0%	2.0%	13.7%	7.7%
Fear	14.4%	10.0%	**37.0%**	4.7%	4.0%	8.2%	21.8%
Happiness	5.0%	5.6%	3.6%	**67.5%**	11.8%	5.6%	1.0%
Neutral	4.5%	3.3%	5.0%	12.0%	**54.4%**	16.7%	4.1%
Sadness	14.7%	5.4%	8.2%	11.5%	11.2%	**43.9%**	5.3%
Surprise	9.5%	10.7%	17.4%	7.4%	3.1%	8.2%	**43.7%**

Table 3.4: Confusion matrix for the 5-fold cross validation results on the CK dataset.

	Anger	Disgust	Fear	Happiness	Neutral	Sadness	Surprise
Anger	**95.0%**	3.2%	0.0%	1.0%	0.8%	0.0%	0.0%
Disgust	1.0%	**97.4%**	1.1%	0.0%	0.0%	0.0%	0.6%
Fear	0.6%	0.0%	**86.1%**	4.7%	6.9%	1.6%	0.0%
Happiness	0.0%	0.0%	0.0%	**98.8%**	1.4%	0.0%	0.0%
Neutral	0.0%	0.0%	0.0%	1.0%	**98.6%**	0.4%	0.0%
Sadness	1.6%	0.0%	0.0%	0.0%	2.5%	**95.7%**	0.0%
Surprise	0.0%	0.0%	2.0%	0.4%	0.0%	0.0%	**97.5%**

3.6.4 Active Learning vs Passive Learning

In this section, we perform experiments to investigate the reduction in number of training examples required for active learning to obtain similar classification accuracy

Table 3.5: Confusion matrix for the 5-fold cross validation results on the JAFFE dataset.

	Anger	Disgust	Fear	Happiness	Neutral	Sadness	Surprise
Anger	**83.5%**	6.2%	2.0%	1.5%	3.9%	1.0%	1.9%
Disgust	5.5%	**86.5%**	0.0%	4.5%	2.1%	1.2%	0.0%
Fear	4.0%	2.5%	**82.4%**	7.0%	4.0%	0.0%	0.0%
Happiness	0.0%	0.0%	1.6%	**92.5%**	5.8%	0.0%	0.0%
Neutral	4.0%	0.0%	0.0%	5.5%	**89.4%**	0.0%	1.0%
Sadness	7.2%	4.8%	5.0%	0.0%	2.6%	**79.6%**	1.0%
Surprise	0.0%	3.2%	5.8%	5.0%	0.0%	0.0%	**86.0**%

Table 3.6: Comparison of recognition performance when training and testing happen on the same and different datasets. We use G_w as our training dataset and G_v as our evaluation dataset.

Training \ Testing	Our G_v	CK	$JAFFE$
Our G_w	48.8%	49.3%	45.1%
Cohn-Kanade (CK)	26.4%	95.6%	35.3%
$JAFFE$	24.2%	35.4%	85.7%

as passive learning. The passive learning is performed by randomly selecting 5 images from the learning pool, in contrast to the active learning where images are "actively" selected based on their rankings. We focus the study on the category of *happiness* facial expression images for illustration purposes. For both learning approaches, we begin with a pool of 25 randomly selected labeled examples. At each round of learning, we select 5 images to query user for labels based on the estimated classification

certainty for the active learning approach, while selecting 5 random images for passive learning. The learned classifiers is then evaluated on G_v. We report the results of classification accuracy after each iteration of learning in Fig. 3.7. It can be noted that there is no much difference in classification accuracy during the first few rounds of learning, which is mainly because we start with the same seed training set for both approaches and the proportion of actively selected examples is far fewer than the number of randomly selected examples. However, as the number of learning rounds increases, it becomes evident that active learning reduces the significant number of training examples required to obtain similar classification accuracy. In particular, the classification accuracy of active learning with 150 images is better than the passive learning with roughly 300 images as shown in Fig. 3.7.

3.6.5 Benchmarks with Near Frontal Face Images

Facial images in JAFFE and CK datasets are captured strictly at frontal view. However, the images in our web dataset G_w are varied in pose with approximately of $\pm 15^\circ$ against the front view. This motivates us to study its impacts on near-frontal facial expression recognition. The experiment is performed by training three sets of facial expression recognition classifiers for the three datasets (JAFFE, CK and G_w) respectively, and evaluating each set on a benchmark dataset. We use the same facial expression recognition algorithms as discussed in Section 3.6.2. And BU-3DFE is used as the benchmark dataset, which is a 3D facial expression dataset [103]. The BU-3DFE dataset contains 100 subjects, of which 56% are female and 44% are male, ranging from 18 years to 70 years old with a variety of ethnic backgrounds. For each subject, the dataset captures six universal expressions (*happiness*, *disgust*, *fear*,

Table 3.7: Benchmark results on BU-3DFE dataset with classifiers trained on G_w, $JAFFE$ and CK respectively.

	G_w	$JAFFE$	CK
Classification Accuracy	**58.2%**	34.7%	38.9%

angry, *surprise* and *sadness*) with four levels of intensity plus the *neutral* expression. In our experiment, we generate 9 facial images for each subject by rotating and projecting the 3D expression models with the strongest intensity. Each facial image corresponds to one near-frontal facial view with 3 yaw angles ($-15°$, $0°$, $+15°$) and 3 pitch angles ($-15°$, $0°$, $+15°$). Totally we have 6300 facial images. We believe that the combination of yaw and pitch angels is able to resemble the near-frontal facial views typically found in realistic environments.

As shown in Table 3.7, the classifiers trained on G_w outperform the other two counterparts by a great margin (more than 20%). The result demonstrates the benefits of training facial expression algorithms using datasets with large variations of imaging conditions. It also suggests that a diverse dataset be required to further develop facial expression recognition system that will be practical in realistic environments.

3.6.6 Effectiveness of Multiscale-WLD Based Facial Feature

In this section, we compare our Multiscale-WLD face descriptor with the original Singlescale-WLD descriptor. As described in Section 3.5, the Multiscale-WLD descriptor is extracted at three image scales by factors of 1 (original resolution), 0.6, and 0.3 in both horizontal and vertical dimensions. Meanwhile, the Singlescale-WLD

descriptor is obtained from the image at its original resolution. We use the same WLD parameter settings for both descriptors, and no weighting for local region is used for this experiment. We perform the test on all three evaluation datasets and the result is reported in Table 3.8. Multiscale-WLD yields clearly higher recognition rates than the original approach, which provides evidence that the discrimination power of the Multiscale-WLD facial feature outperforms the single scale based descriptor due to its ability in capturing and encoding textures with different sizes from facial images. The Multiscale-WLD descriptor is also compared with the Contextualized Multiscale-WLD descriptor. It is noted that the Contextualized Multiscale-WLD descriptor outperforms its conventional counterparts on G_v dataset while underperforming on CK and $JAFFE$. The result suggests that incorporating the local spatial relationships between the patterns into the descriptor can help reduce the issues caused by inconsistent face alignment which is commonly found in our dataset. However, the benefits cannot be readily determined when the face images are collected in a well controlled environments (*i.e.* JAFFE and CK).

To gain better understanding on the WLD based facial representation, we compare it with Local Binary Pattern (LBP) descriptor. The results shown in Table. 3.8 indicate that the WLD based representations are more robust than LBP based representations. Moreover, compared with their single-scale counterparts, both Multiscale-WLD and Multiscale-LBP achieves better recognition performance.

3.6.7 Impact of Parameter Settings

As discussed in Section 3.5, there are six parameters that should be selected to optimize the performance of Multiscale-WLD based facial representation. Of them,

Table 3.8: A comparison of 5-fold cross validation results for different WLD and LBP based facial feature on different evaluation datasets.

	S-WLD	M-WLD	CM-WLD	S-LBP	M-LBP
G_v	57.6%	59.9%	**60.5%**	46.3%	48.8%
$JAFFE$	83.1%	**85.7%**	84.9%	83.5%	84.9%
CK	92.6%	**95.7%**	91.3%	92.0%	92.5%

three original WLD parameters (M, T, S) control the dimensions of a WLD descriptor, N determines the number of regions that a facial image will be divided into, R denotes the number of scales that a facial image will be analyzed at and σ is its scale factor. In this section, we study the impact of each parameter by varying their values one at a time while fixing the other ones. For instance, when N is the interest of study, we only vary the value of N while fixing the values of all other parameters. The experiment is performed using the dataset G_v in a 5-fold cross-validation manner.

The parameters M, T, and S determines the discriminability and statistical reliability of the WLD descriptor [71]. One should note that the experiment is performed at original image resolution. The experimental results in Figs. 3.8(a), 3.8(b) and 3.8(c) show that the overall performance is not necessarily affected, though the changes of these parameters result in significant difference in the dimension of the WLD descriptor. This suggests that WLD based facial representation is very robust with respect to the change of parameters.

The size N is a trade-off between computational complexity, discriminative power and tolerance against face localization errors. Fig. 3.8(d) shows its effects. As N increases, both the discriminative power and computational complexity increase. This

trend continues to a point that the proposed method starts to become sensitive to a small change in face localization (25 in G_v), from where the discriminability starts to decline. This is because the proposed methods computes histograms over local regions so a small change in face registration relative to the grid only causes changes in the labels on the borders of the local regions. Therefore as the number of region N increases, the method becomes more prone to the localization errors.

Varying the size of R and σ enables Multiscale-WLD to deal with textures at different scales. A large value can increase the redundant information while a small value can cause loss of important information. Figs. 3.8(c) and 3.8(f) illustrate their minor impacts to the recognition accuracy.

3.7 Conclusion

In this chapter, we present a semi-automatic and scalable framework for harvesting facial expression images from the Web. The use of active learning minimizes human efforts required for data collection and cleaning. In addition, a novel facial feature based on Weber Local Descriptor (WLD) and histogram contextualization is proposed for multi-resolution analysis of faces. Our comprehensive experimental results on several benchmark datasets demonstrate that the proposed facial feature is robust, and the framework is capable of constructing diverse and high quality facial expression datasets. Compared to other popular datasets collected in controlled laboratory environments, the state-of-the-art facial expression recognition algorithm trained using our dataset shows better generalization. Our work is one step further toward more advanced affective analysis of realistic multimedia data.

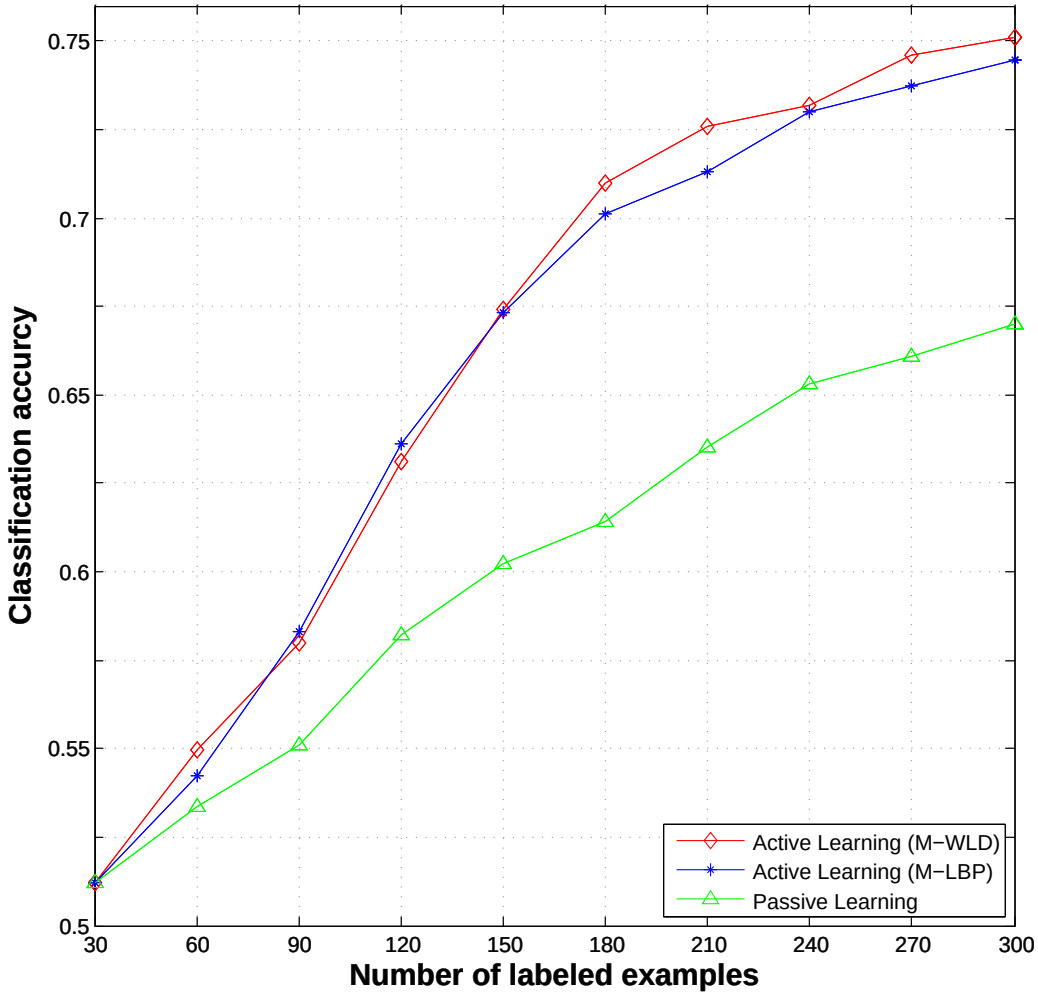
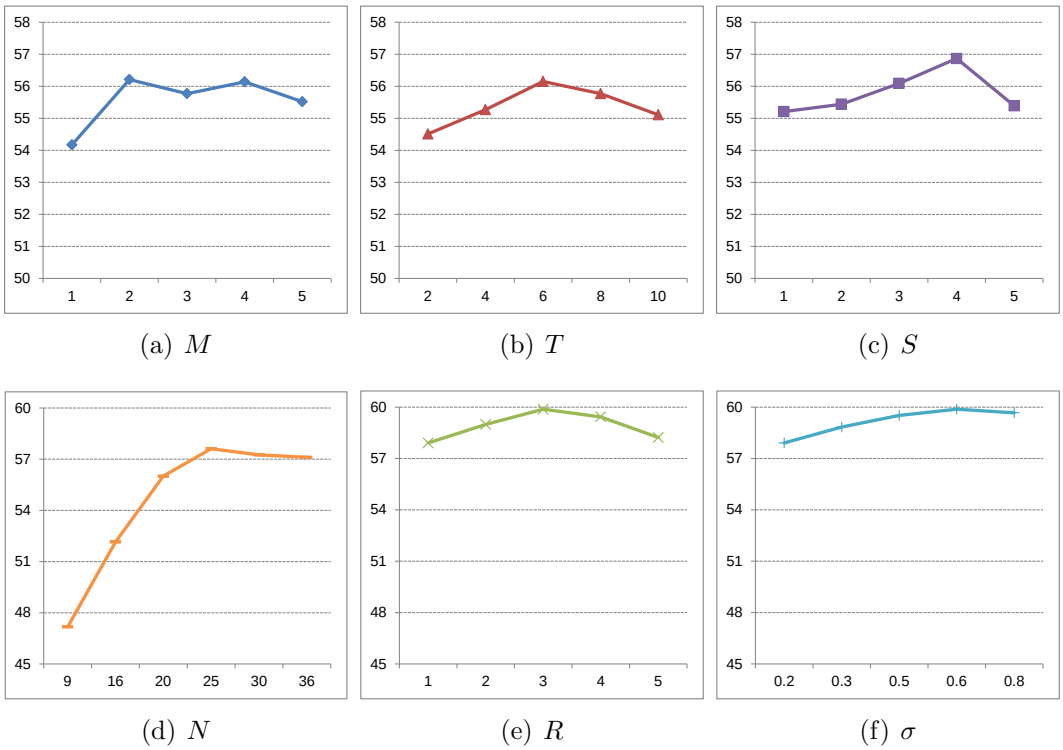


Figure 3.7: Comparison of classification performance between active learning and passive learning for *happiness* expression in G_v. The classification accuracy is defined as the ratio of the number of correctly classified images to the total number of images in the test dataset.



(a) M (b) T (c) S

(d) N (e) R (f) σ

Figure 3.8: Impact of parameter settings. The Y-axes denote the recognition accuracy.

CHAPTER 4

FACIAL EXPRESSION FEATURES

In this chapter, two novel facial feature extraction algorithms are presented to handle facial expression images that are collected using the search based framework discussed in the previous chapter. The first feature, namely spatially enhanced local binary pattern, incorporates the spatial contextual information into the famous LBP descriptor using a shape context based representation. The other one, namely Local Patch Pattern, combines local feature descriptors exracted from neighboring patches to form a second order representation. Experiment results show that both features are more descriptive and robust in the presence of noise, which are commonly observed in practical environments.

4.1 Spatially enhanced Local Binary Pattern

4.1.1 Introduction

Texture analysis plays an important role for many applications in image processing and computer vision, such as facial expression recognition, image segmentation, content based image retrieval, and medical imaging. Various methods have been proposed for texture feature extraction over the past decades [104, 105]. In particular, LBP based approaches have drawn significant attentions due to LBP's discriminative

power and computational simplicity. The original LBP operator proposed by Ojala *et al.* [106] labels each pixel of an image with a binary number formed by thresholding the 3×3 neighborhood of each pixel against the intensity of the pixel into a series of 0s and 1s. The histogram of LBPs captures the distribution of the local micro-patterns and is able to characterize texture content.

Since scale and rotation are important in representing texture content, the original LBP operator has been extended to address such two issues. In order to capture dominant features at different scales, the original LBP operator was extended to use neighborhood of various sizes. Using circular neighborhoods and bi-linearly interpolating the pixel values allow any radius and any number of pixels in the neighborhood [50]. Another variant has considered the different shapes for the neighborhood calculation. The circular definition of neighborhood in the original LBP has been replaced by ellipse, parabola, hyperbola and archimedean spiral of different sizes [107]. To obtain rotation invariance, a bitwise shift was performed on the binary pattern until the binary value was matched with one of 36 rotation invariant patterns [106]. It has also been shown that a subset of LBPs can be used to describe most patterns occurring in images; hence, the dimensionality of LBP descriptors can be greatly reduced [108].

Many other variants to the original LBP operator have also been proposed. Tan and Triggs [109] quantized the difference between a pixel and its neighbors into three levels (instead of 2 values as in the original LBP) with a three-valued code in order to reduce LBP's sensitivity to noise in near-uniform image regions. Nanni *et al.* [107] extended this idea by using a five-value encoding in order to obtain a more robust descriptor. Alternatively, the neighborhood pixels were thresholded against their median and mean values, instead of the center pixel, to reduce the effect of

noise [110]. Xie *et al.* [91] proposed Local Gabor XOR Pattern to exploit the Gabor phase information by extracting LBP from images convolved with the Gabor filters. Recently, Guo *et al.* [111] proposed the Local Configuration Pattern (LCP) which integrates both the image microscopic configuration and the occurrences of the local binary patterns.

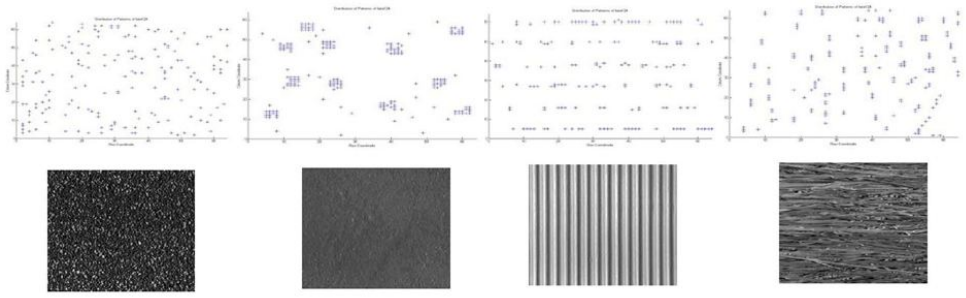


Figure 4.1: Illustration of the spatial distribution of a same texture pattern in four different texture images. It demonstrates the importance of spatial information in describing textures.

However, the spatial distribution (*i.e.* structure) of the local micro-patterns has not been exploited, which has been proved to be an important property of texture [108] (see Figure 4.1). Since shape context [112] is very good at characterizing the spatial relationship among the sample points along a contour, we propose a novel texture descriptor, namely spatially enhanced LBP, by utilizing shape context based representation to encapsulate the spatial distribution of LBPs. In order to make our texture descriptor invariant to rotation, the original shape context is adapted to the local property of each pixel. Firstly, a LBP-coded image is constructed with the LBP operator for a given image, where each pixel value corresponds to the binary value

of the LBP extracted from the pixel at the same location of the given image. A sample case is shown in Fig. 4.2 where the circulation starts at the top-left position. Secondly, shape context is extracted to capture the spatial distribution of the local binary patterns. At each sampling point in the LBP-coded image, one modified shape context histogram is constructed, where each bin of the histogram contains a sub-histogram of the LBPs that fall within the area covered by the bin. We use a dense sampling approach in our experiments. Finally, shape context histograms extracted at all the sampling points are concatenated to form a texture descriptor.

The main contributions of the proposed method are summarized as follows: 1) We propose a novel texture descriptor to encapsulate the spatial distribution of the local binary patterns in a global context. Experimental results show that the proposed method outperforms the original LBP in various well known texture databases under different imaging conditions. 2) We solve the rotation invariance problem of the shape context descriptor by rotating its coordinate system based on the dominant gradient orientation within the neighborhood.

4.1.2 Local Binary Pattern

The original LBP operator undertaken at each pixel labels the neighbor pixels of each pixel by thresholding a circular neighborhood with radius of R pixels with the center pixel t_c's grayscale intensity value (see Fig. 4.2). Formally, the LBP operator is defined as follows:

$$LBP(P,R) = \sum_{i=0}^{P-1} u(t_i - t_c)2^i, \tag{4.1}$$

where P denotes the number of pixels in the neighborhood, R is the radius in pixels of the circular neighborhood, t_i and t_c are the intensity values of the neighbor i and the

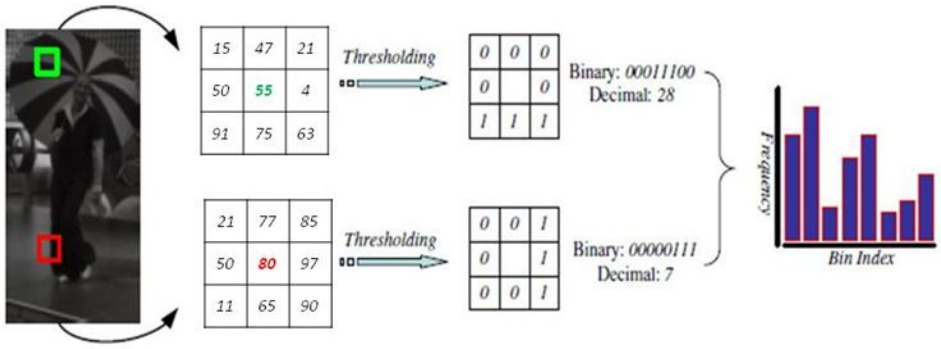


Figure 4.2: A sample illustration of applying the original LBP operator to a pixel, where the circulation starts at the top-left position.

center pixel, respectively, and $u(x)$ is a step function, *i.e.* $u(x) = 1$ when $x \geq 0$ and $u(x) = 0$ otherwise. A pattern is called uniform when containing at most two bitwise transitions from 0 to 1, or vice versa when the binary string is considered circular. For example, 00001100 and 11110111 are uniform patterns. Rotation invariance is achieved by recognizing that $LBP(P, R)$ originates from particular rotation-invariant patterns.

4.1.3 Modified Shape Context Descriptor

Shape context (SC) captures the spatial relationship among the sample points along a shape contour. Specifically, the shape context for a sampling point p_i is a histogram computed by partitioning an image with a log-polar coordinate system, where the sampling point is the origin of the coordinate system and the value of each bin in the histogram equals to the number of sample points fall within the corresponding partition as shown in Fig. 4.4.

However, shape context is not invariant to rotation. That is, when a shape is rotated, the shape context of a sampling point will also change, since the starting angle of the log-polar coordinate does not change adaptively. In order to achieve rotation invariance, the starting angle of the log-polar coordinate for a sampling point will be aligned to the dominant gradient orientation in the neighborhood of p_i. In addition, for a given sampling point p_i, instead of counting the number of sample points within a bin for a binary shape image, we construct a sub-histogram of the local binary patterns within the partition corresponding to the bin for a texture image.

4.1.4 Spatially Enhanced LBP

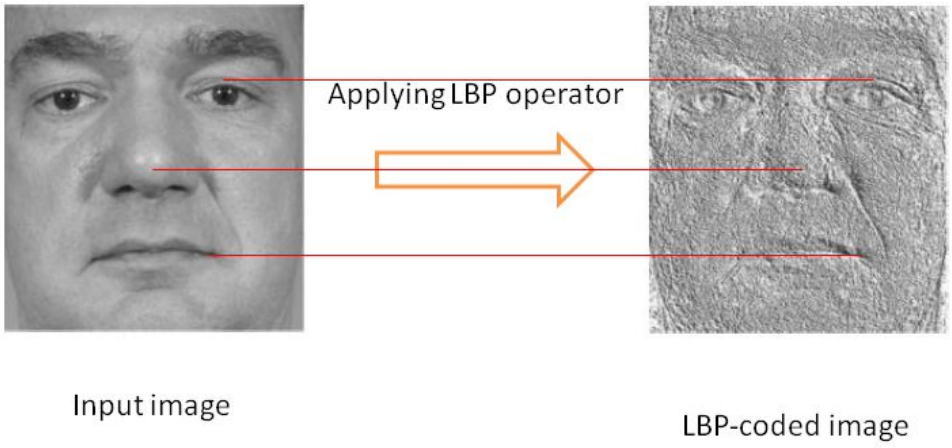


Figure 4.3: Illustration of a LBP coded face image.

There are three major steps to compute the spatially enhanced LBP descriptor for a given image: obtaining LBP-coded image, computing modified shape context descriptor for each sampling point, and forming the final texture descriptor. Let

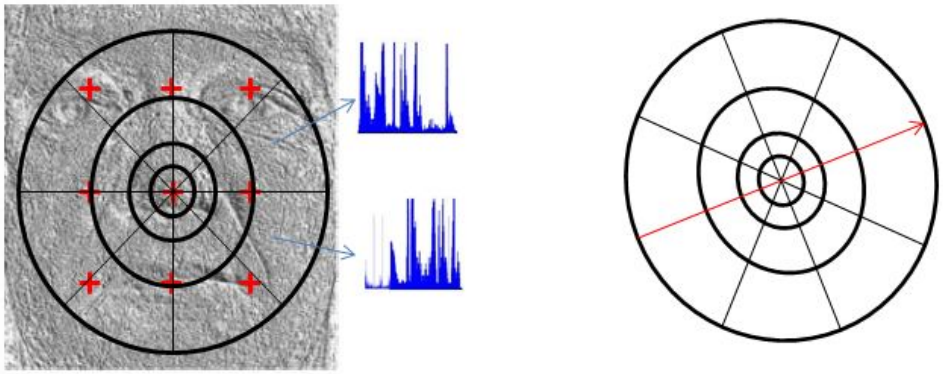


Figure 4.4: Illustration of computing the shape context for the sampling point (*i.e.* the origin of the log-polar coordinate system).

H_i denote the modified shape context descriptor for the i-th sampling point. The final texture descriptor is the concatenation of all the H_i. That is, $H = \{H_k\}, k = 0, 1, ..., K-1$, where K is the number of sampling points.

Note that the placement of sampling points can be modified and optimized according to specific applications. In our experiment, a dense sampling of K points is used and shown to give a good overall classification accuracy. A weighting scheme can be also applied to each histogram H_k based on its importance for a better classification performance.

4.1.5 Experiments

To evaluate the performance of the proposed approach, three widely used texture image datasets were used as in [111]: 1) The Outex_TC_00012 dataset [50] contains 9120 images representing 24 different texture classes captured under different illumination conditions and rotations. We used 20 images of each texture class for training and the rest 8640 images for testing. 2) The KTH-TIPS2 dataset [113] contains 4752

images from 11 different texture classes. Each texture class has 4 physical samples and each sample has 108 images that are obtained at 9 different scales and 12 different illumination and rotation settings. Similar to [111], the 108 images from one sample were randomly selected from each class for training and other images were used for testing. This was repeated 500 times with different training and testing sets. 3) The Columbia-Utrecht (CUReT) dataset [114] consists of 61 different texture classes, where each class has 205 images with different viewpoints and illuminations. Half of the images of each texture class were used for training and the other half for testing.

In our experiment, we selected the original LBP operator with different parameters as the baseline. They were also used to construct LBP-coded images. In order to compute the modified shape context descriptor, 16 equally spaced sampling points were placed on a given image. For each sampling point, a modified shape context histogram was constructed with N_θ equally spaced angle bins and N_d logarithmic spaced distance bins. Grid search was used to optimize the two parameters N_θ and N_d, the optimal values were found to be $N_\theta = 6$ and $N_d = 3$. The support vector machine (SVM) was used as the classifier and the kernel for the SVM is the Gaussian Radial Basis Function (RBF).

The experimental results with the three databases are reported in Table. 4.1. As observed, the proposed method consistently outperform the conventional LBP operators. In particular, the performance is increased more than 4% (the largest improvement among the three datasets) in the most challenging database KTH-TIPS2, which demonstrates that the proposed method is more robust against varying illumination and pose conditions.

To study the influence of the rotation invariant property of the modified shape context descriptor, LBP-coded images were obtained using the $LBP^{riu2}_{16,2}$ operator as it was shown to be robust across all the three testing databases. The results reported in Table. 4.2 show that the rotation invariant shape context descriptor outperforms its conventional counterpart by a considerable margin in all test databases, which indicates it is essential to adapt the conventional shape context.

Table 4.1: Classification accuracy of our method compared with LBP.

	KTH-TIPS2	Outex	CUReT
$LBP^{u2}_{8,1}$	50.72%	58.94%	85.42%
$LBP^{u2}_{16,2}$	49.87%	58.15%	81.12%
$LBP^{u2}_{24,3}$	49.58%	50.33%	81.05%
$LBP^{riu2}_{8,1}$	48.15%	75.25%	85.48%
$LBP^{riu2}_{16,2}$	50.01%	78.89%	85.76%
$LBP^{riu2}_{24,3}$	47.80%	78.63%	85.00%
$LBP^{u2}_{8,1} + SC$	**54.15%**	65.45%	85.57%
$LBP^{u2}_{16,2} + SC$	49.16%	65.00%	82.17%
$LBP^{u2}_{24,3} + SC$	46.95%	60.14%	78.55%
$LBP^{riu2}_{8,1} + SC$	52.30%	78.37%	87.01%
$LBP^{riu2}_{16,2} + SC$	54.13%	79.20%	**87.15%**
$LBP^{riu2}_{24,3} + SC$	52.94%	**79.77%**	86.90%

Table 4.2: Influence of rotation invariance.

	KTH-TIPS2	Outex	CUReT
Rotation Invariant SC	**54.13%**	**79.20%**	**87.15%**
Conventional SC	53.95%	75.94%	84.80%

Table 4.3: Influence of rotation invariance.

	JAFFE	CK	GWI
Conventional LBP	84.13%	89.50%	46.05%
LBP + SC	85.27%	89.73%	48.60%

We also study the proposed method for facial expression recognition. Three datasets are used in our experiments:JAFFE [4], COhn-Kanade DFAT (CK) [68], and the facial expression dataset constructed using Google web image search with our image collection framework discussed in Chapter 3. The JAFFE dataset contains 213 images of seven basic facial expressions which were posed by 10 Japanese females. The CK dataset consists of approximately 500 images from 100 subjects ranged in age from 18 to 30 years, of which 65The GWI facial expression dataset includes 50 images for each of the seven basic facial expressions. We compare the proposed method with conventional LBP, and the bet results are reported in Figure 4.3.

4.1.6 Summary

In this section, we presented a texture feature extraction method by taking the spatial distribution of the local binary patterns into account in a global context with a shape context based approach. We also achieved the rotation invariance of the shape context descriptor of a sampling point by rotating its coordinate system according to the dominant gradient orientation within the neighborhood of the sampling point. The proposed method has been evaluated against the conventional LBP descriptor with three widely used benchmark datasets: Outex, KTH-TIPS2 and CUReT. Experimental results demonstrate the superior performance of our proposed descriptor

in texture classification, in particular with challenging texture datasets. It would be promising to extend the proposed idea to other LBP variants.

4.2 Local Patch Pattern

4.2.1 Introduction

Automatic facial expression recognition has attracted significant attention over the past decades due to its importance in a wide range of applications such as human-computer interaction (HCI), image or video understanding, and affective computing. The state-of-art facial expression recognition methods are able to achieve impressive accuracy in tightly controlled laboratory settings, where face images are normally acquired in near frontal pose under strict lighting requirements. However, the performance degrades abruptly in highly unconstrained conditions akin to those found in the real world, thus limiting their practical use [8].

One major factor affecting the performance of existing facial expression recognition methods in practical environments is the difficulty of handling the diverse head pose variations. In general, head pose variations can be divided into two categories: those resulting from in-plane rotations and those produced by out-plane rotations. The former occurs when the head tilts to the left or right without turning, as shown in the first row of Figure 4.5. The whole frontal faces are visible and toward the camera, but they are not necessarily upright. The latter refers to cases where the faces are turned away from the camera as depicted in the second row of Figure 4.5. Some parts of the faces are occluded, thereby the rotated faces tend to be less informative and more diverse in appearances. The intra-class variations introduced by head poses together with some other factors are often more pronounced than the inter-class differences

caused by different types of facial expressions, which can be attributed to the reasons behind the decrease in recognition performance.

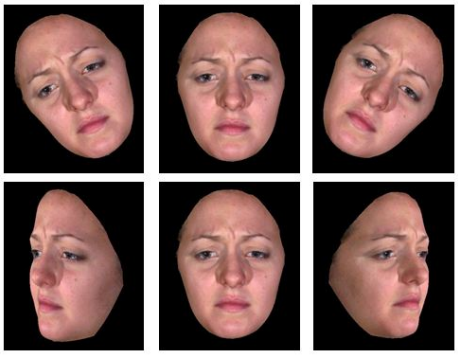


Figure 4.5: Illustrations of the two types of head pose variations. The first row shows the head pose variations caused by in-plane rotations and the second row shows the out-plane variations.

It has been widely acknowledged in the literature that incorporating face alignment procedures in the facial expression recognition pipeline can effectively remove some undesired intra-class variations (such as head poses described above) and greatly improve the recognition performance [115]. However, face alignment has to be typically performed either manually or by training algorithms with samples that have been hand-labeled with facial components. Due to the amount of supervision required by these methods, the alignment procedures are not always feasible in practical settings. In addition, most face alignment methods are known to fail when large pose variations are presented [116], even manual alignment is prone to human errors. The misalignment may change the underlying semantic meanings of face images and negatively affect the subsequent expression recognition accuracy.

In this work, instead of requiring precise alignments of faces, we investigate approaches to model facial expression as an orderless collection of visual words, namely bag-of-visual-words (BOVW) model. The orderless property is a double-edged sword. The good part is that discarding the spatial information of the visual words allows a degree of face misalignment and pose variations, making it ideal for practical environments. The downside is that it limits the descriptive power for representing facial expressions, thus it requires to work with more discriminative features for robust codebook construction. To utilize the positive effects and to limit the negative impacts, we propose a a novel local texture descriptor for BOVW based facial expression recognition, namely Local Patch Pattern (LPP). The LPP descriptor aggregates the statistics about the distributions of texture patterns around the neighborhood of a keypoint. The idea is inspired by the Local Binary Pattern (LBP). However, instead of using the pixel values directly, we propose to combine the local texture descriptors extracted from neighboring patches for more robust feature representation. LPP can be used in conjunction with any histogram based local texture features. In this work, the SIFT descriptor is used to capture the local distribution of textures within a patch.

In summary, the main contribution of our work can be highlighted as the follows:

1. A novel local descriptor, *i.e.* LPP, is proposed to aggregate the texture patterns around a keypoint by combining local texture descriptors extracted from neighboring patches in a way inspired by the Local Binary Pattern. The patch based approach allows LPP to be more descriptive and robust in the presence of noise and illumination changes which are commonly observed in practical environments.

2. We propose a bag-of-words based facial expression recognition framework. Our method can tolerate face misalignments that are caused by face registration errors. Although it requires only rough localization of faces which can be easily obtained using popular face detectors such as Viola-Jones face detector [2], the recognition performance is comparable to state-of-art methods relying on precise face alignments.

3. We conduct comprehensive experiments to investigate the BOW based face representations for expression recognition using the proposed feature comparing to conventional SIFT descriptor. Experimental results demonstrate that the loss of spatial information (due to the BOW approach) does not severely degrade the recognition performance when adequate feature (*e.g.* LPP) is used; and it allows a certain degree of face misalignments and pose variations which can benefit practical applications. Further experiments also show that our method can deal with multiview expression recognition to some extent.

The rest of this section is organized as follows. In Section 4.2.2, we review the related works and introduce the motivations of this work. In Section 4.2.3, the BOW based framework for facial expression recognition is described, which is followed by presenting the proposed LPP descriptor. In Section 4.2.4, experimental results on two publicly available databases are reported and discussed to demonstrate the effectiveness of the proposed method.

4.2.2 Related Work

Near-frontal facial expression recognition

Near-frontal facial expression recognition has been studied intensively over the past decades, and a variety of approaches have been presented. In general, these approaches can be divided into two groups based on the features used: geometric based approaches [6, 30, 38, 39] and appearance based approaches [42, 81, 82, 83, 100, 117]. For details, please refer to Section 3.2.2.

Multiview facial expression recognition

Most of the existing methods focus on the near-frontal facial expression recognition since facial expression datasets primarily capture frontal view face images only. Recent datasets such as BU-3DFE [103] allows investigation of multiview facial expression recognition, and a few researchers [116, 118, 119, 120] have begun to explore this fascinating area. Based on the BU-3DFE datasets, they synthesized multiview facial images by rotating the 3D facial expression models in the database to the desired poses and projecting them onto a 2D image plane.

By using the synthesized multiview facial images, Hu *et al.* [118] investigated the problem of facial expression recognition from non-frontal views with five pan angles, namely 0°, 30°, 45°, 60° and 90°, respectively. They combined the geometric features, defined by the location of 83 manually labeled facial feature points, and various classifiers such as nearest neighbor and the support vector machine to recognize six universal facial expressions. Zheng *et al.* [116] studied the same problem with the same five pan angles. Instead of using the geometric features, they employed the

texture features, defined as the scale-invariant feature transform (SIFT) feature vectors. They divided a facial image into subregions, and then extracted SIFTdescriptors from each subregion in the image. They proposed a novel method for feature selection based on the minimization of an upper bound of the Bayes error and reduced the dimensionality of the SIFT feature vectors. The reduced-dimensional feature vectors were then classified with the k-nearest-neighbor (KNN) classifier. Rudovic *et al*. [120] proposed an approach to multiview facial expression recognition based on a set of 39 manually labeled facial points. The facial points were projected into a low dimensional manifold by multi-class Linear Discriminant Analysis (LDA), and a Gaussian Mixture Model was used to estimate the head pose. They proposed a Coupled Scaled Gaussian Process Regression (CSGPR) model to learn the mapping between a discrete set of non-frontal poses and the frontal pose. Facial expression recognition was achieved by applying a multi-class support vector machine classifier to the pose-normalized facial points. Moore *et al*. [119] investigated Local Binary Patterns (LBP) and its variants for multiview facial expression recognition, and use it for pose normalization. Instead of using a set of manually labeled facial feature points, they adopt a dense uniform sampling approach and use a multi-class support vector machine to learn pose and pose dependent facial expression classifiers.

4.2.3 LPP based facial expression representation

Bag-of-words model represents faces as an orderless collection of visual words, where each visual word is formed by a group of similar local descriptors. The orderless property allows the representation to be invariant to face misalignments and pose variations to a certain degree, making it ideal for practical environments. However,

ignoring the spatial relationship will limit the descriptive power for representing facial expressions. In order to exploit the benefits of BOW model while maintaining sufficient recognition accuracy, we use the LPP descriptor in conjunction with SIFT for a more descriptive face representations. In the following, we first describe the workflow of the bag-of-words based facial expression recognition framework. A review of the SIFT descriptor is followed in Section 4.2.3 and we present the proposed LPP descriptor in Section 4.2.3.

Workflow of the Framework

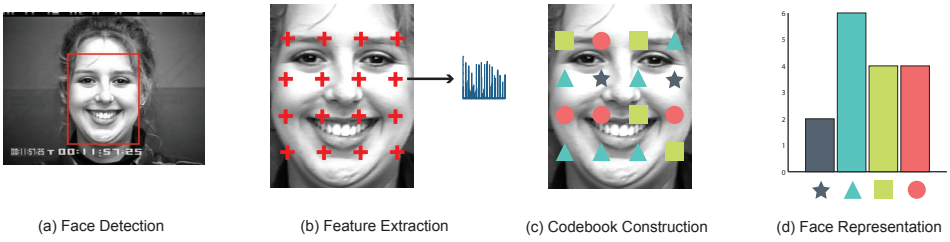


Figure 4.6: Workflow of the BOW based framework for facial expression recognition. The red crosses in (b) refer to the densely sampled keypoint locations, and the different shapes in (c) refer to different visual words obtained using KNN.

The workflow of the proposed BOW based framework for facial expression recognition is presented in Figure 4.6. First, face detection is performed using the Viola-Jones [2] face detector for each input image, and the detected face regions are cropped. The proposed method works on the raw output of the face detector and does not require any other preprocessing steps such as face alignment. We then compute the LPP descriptors densely sampled on the cropped face image with a fixed step size. It should be noted that when the step size is small, there will be significant overlap between

neighboring descriptors. Subsequently in an offline step, the contextualized SIFT descriptors randomly selected from a set of training images are quantized to build a visual vocabulary using approximate K-means clustering. The clustering approach is based on calculating data-to-cluster distances using the Approximate Nearest Neighbor algorithm, and each cluster center represent a visual word. The effects of the sampling step size w and the vocabulary size v will be discussed in Section 4.2.4. Finally, each feature descriptor can be mapped to the nearest visual word, and a given face image can be represented as a histogram counting the occurrences of each visual word in the image.

LBP

The LBP descriptor captures the first order circular derivative pattern of an image, which is a micro texture pattern generated by concatenating the binary gradient directions [49]. It labels the pixel of the image by thresholding a circular neighborhood with radius of R pixels with the center pixel t_c's value in grayscale, and considering the results as a binary number. Formally, the LBP operator is defined as follows:

$$LBP(P,R) = \sum_{i=0}^{p-1} u(t_i - t_c)2^i,$$

where P denotes the number of pixels in the neighborhood, R is the radius in pixels of the circular neighborhood, t_i and t_c are the intensity in grayscale of the neighbor i and the center pixel respectively. $u(x)$ is a step function, *i.e.* $u(x) = 1$ when $x \geq 0$ and $u(x) = 0$ otherwise. Using circular neighborhoods and bi-linearly interpolating the pixel values allow any radius R and number of pixels P in the neighborhood [50].

SIFT

The SIFT descriptor encodes local gradient information in the neighborhoods of some keypoint locations. For each keypoint, there are three steps to calculate its SIFT descriptor. Firstly, gradient magnitudes $m(x, y)$ and orientations $\theta(x, y)$ are computed in a 16×16 pixels sampling region centered on the interest point location using pixel difference:

$$m(x,y) = \sqrt{\left(L\left(x+1,y\right) - L\left(x-1,y\right)\right)^2 + \left(L\left(x,y+1\right) - L\left(x,y-1\right)\right)^2}, \quad (4.2)$$

$$\theta(x,y) = \tan^{-1}\left(\frac{L\left(x,y+1\right) - L\left(x,y-1\right)}{L\left(x+1,y\right) - L\left(x-1,y\right)}\right), \quad (4.3)$$

where L is the Gaussian smoothed image, x and y are pixel coordinates. In order to reduce the influence introduced by small changes in the position of the window, the gradient magnitudes are weighted with a Gaussian weighting function with σ equals to one half the width of descriptor window. The weights are reduced smoothly from center to the edge. Smaller weights are assigned to the gradients that are far from the center of the descriptor as they are most likely to be affected by registration errors.

Next, the weighted gradient magnitudes are accumulated into an orientation histogram with 8 orientation bins at a step size of 45° over 4×4 pixels regions. Soft assignment of values to adjacent histogram bins is performed by trilinear interpolation to reduce the effects of location and dominant orientation mis-estimation. Specifically, each gradient magnitude into a bin is multiplied by a weight of $1 - d$ for each dimension, where d is the distance between the corresponding gradient orientation and the orientation the bin represents. For each keypoint, a 4×4 array of orientation histograms are computed. The raw descriptor is obtained by concatenating the array

of orientation histograms, resulting a $4 \times 4 \times 8 = 128$ dimensional feature vector. The peaks of the gradient orientation in the sampling region correspond to the dominant orientation of the keypoint. Orientation invariance is achieved by rotating the coordinates of the descriptor and the gradient orientation relative to its dominant orientation.

The last step is to normalize the raw feature vector to reduce the effects of illumination change. The vector is first normalized to unit length to enhance variance in contrast change, as the change in contrast is equivalent to multiply the pixel values by a constant which will cause the gradient magnitudes to be multiplied by the same constant. The feature vector is invariant to brightness changes as they are computed using pixel difference, so a constant added or removed from pixel values will not affect the gradient values. The non-linear illumination changes such as camera saturation can cause a large change in relative magnitudes for certain gradients [92]. The influence of large gradients are reduced by thresholding all dimensions of the unit vector to a value of no more than 0.2 and, the resulting vector is once again normalized to unit length to make the feature vector more robust to non-linear illumination changes.

LPP

LBP considers the spatial context in pixel level which is not sufficient to deal with the high intraclass variations and interclass similarities of facial expressions. To address this issue, we extract the LPP descriptor which is formed by aggregating SIFT descriptors extracted from neighboring patches around a keypoint. Besides encoding the local gradient distribution within a single patch like SIFT, the LPP descriptor also considers the relationship among the gradient distributions extracted from its neighboring regions.

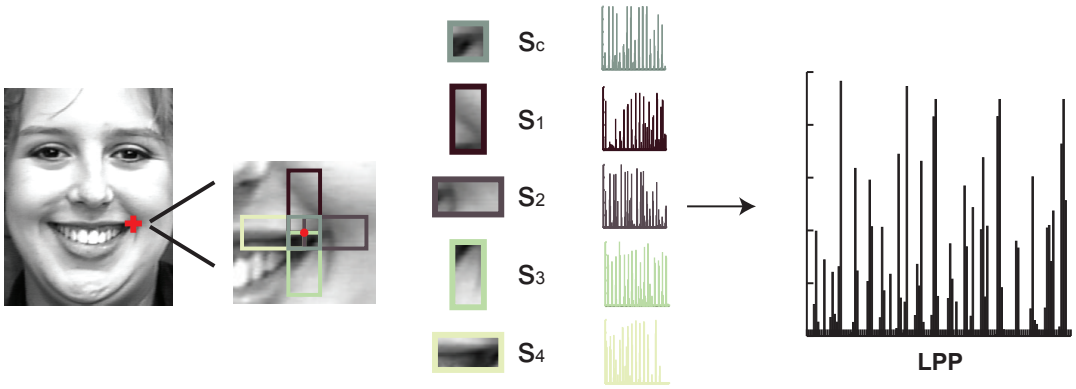


Figure 4.7: Illustration of the LPP descriptor. The red dot in the center indicates the keypoint.

The descriptor is obtained in two steps. At the first step, SIFT descriptors are extracted from five patches in a cross shaped neighborhood of each key point, as shown in Figure 4.7. The red point in the middle refers to the sampling point p, which is also the center point of the center region. The center patch is a square with a size of $n \times n$ pixels. Assume the sampling point has a coordinate of $(0, 0)$, the center point for the left and right patches are at $(-\frac{m}{2}, 0)$ and $(\frac{m}{2}, 0)$ respectively. Both of them have a size of $m \times n$ pixels. The top and bottom patches have their centers at $(0, \frac{m}{2})$ and $(0, -\frac{m}{2})$ with a size of $n \times m$ pixels. The influence of the patch size parameters m and n will be investigated in Section 4.2.4. Five SIFT descriptors are extracted from the five patches individually, which will be used as the pooling candidates for deriving the LPP descriptor.

At the second step, the proposed LPP descriptor is formed by combing the SIFT descriptors obtained at the previous step. The objective is to construct a more descriptive feature representation that preserves the neighborhood information. We define

four operations to combine the descriptors, namely average neighborhood (AN), max neighborhood (MN), average contrast (AC) and max contrast (MC). The average neighborhood operation labels the sampling point by averaging the center descriptor with its four neighbors, *i.e.* the SIFT descriptors extracted from the left, right, top and bottom patches as shown in Figure 4.7. Formally, it is defined as follows:

$$h^{AN} = \frac{1}{5}\left(\sum_{i=1}^{4} s_i + s_c\right), \tag{4.4}$$

where s_i stands for the ith neighboring SIFT descriptor obtained from the first step and s_c refers to the center descriptor. The max neighborhood operation is performed by assigning each bin of the histogram h^{MN} the largest value out of the associated bins from the five pooling descriptors, formally:

$$h_j^{MN} = \max(s_{c,j}, s_{1,j}, \cdots, s_{4,j}), \text{for } j = 1, \cdots, 128, \tag{4.5}$$

where $s_{i,j}$ refers to the jth bin of the SIFT descriptor (histogram) extracted from the ith neighboring patches and $s_{c,j}$ refers to the jth bin of the center SIFT descriptor. In the contrast based pooling operations, the four neighboring SIFT descriptors are normalized by substracting the center descriptor. Formally, the average contrast operation is defined as follows:

$$h^{AC} = \frac{1}{4}\sum_{i=1}^{4}\left(s_i - s_c\right), \tag{4.6}$$

where h_i^p refers to the neighboring SIFT descriptors at sampling point p and h_c^p is the center descriptor. And the maximum contrast operation is defined as follows:

$$h_j^{MC} = \max(s_{1,j} - s_{c,j}, \cdots, s_{4,j} - s_{c,j}), \text{for} j = 1, \cdots, 128. \tag{4.7}$$

4.2.4 Experiments

Datasets

Two publicly available datasets are used in our experiments: Cohn-Kanade DFAT (CK+) [68] and BU-3DFE [103] dataset. The Cohn-Kanade DFAT dataset consists of approximately 500 images from 100 subjects ranged in age from 18 to 30 years, of which 65% are female. The distribution of the ethnic groups is: 81% Euro-American, 13% Afro-American and 6% other groups. The BU-3DFE dataset is a 3D facial expression dataset [103]. It contains 100 subjects, of which 56% are female and 44% are male, ranging from 18 years to 70 years old with a variety of ethnic backgrounds. For each subject, the dataset captures six universal expressions (*happiness*, *disgust*, *fear*, *angry*, *surprise* and *sadness*) with four levels of intensity plus the *neutral* expression. In our experiment, we generate 7 facial images for each subject by rotating and projecting the 3D expression models with the strongest intensity. Each facial image corresponds to one facial view with 7 yaw angles ($-45°$, $-30°$, $-15°$, $0°$, $+15°$, $+30°$, $+45°$). In total we have 4900 facial images. We believe that the combination of yaw angels is able to resemble the pose variations typically found in realistic environments. An overview of the evaluation datasets is shown in Table 4.4.

Experimental Settings

The experiments were conducted on 7 categories of universal facial expressions: *happiness*, *sadness*, *anger*, *fear*, *disgust*, *surprise* and *neutral*. The original face images are cropped using the Viola-Jone face detector [2] before feature extraction. The LPP descriptor works on the raw output of the face detector, and no other face alignment procedures are performed. The conventional SIFT descriptor is used as a benchmark.

Table 4.4: Statistics of the evaluation datasets used in the experiments. The columns of Male and female show the percentage of male subjects and female subjects respectively.

	Images	Subjects	Female	Male
$CK+$	500	100	65%	35%
$BU-3DFE$	4900	100	56%	44%

In order to allow for a fair comparison, the parameters in both methods are optimized empirically during a series of preliminary experiments. The effects of the parameter settings will be discussed in the following sections.

We also compare the proposed method with LBP descriptor, which has been widely used in facial expression recognition. The LBP descriptor captures the first order circular derivative pattern of an image, which is a micro texture pattern generated by concatenating the binary gradient directions [49]. It labels the pixel of the image by thresholding a circular neighborhood region with the center pixel's value in grayscale, and considering the results as a binary number. Following our previous work [121, 122], we use the 59-bin $LBP^{u2}_{8,2}$ operator. As the LBP based face representation assumes faces are well aligned, we perform automatic eye localization on faces detected by the Viola-Jones face detector. For this task, we adopt the Average of Synthetic Exact Filters (ASEF) [96] which is a class of correlation filters. Then we align and normalize the faces into a common coordinate system based on the detected eye locations and the distance between the two eyes.

Support Vector Machine (SVM) is used as the classifier, and one SVM classifier is trained for each facial expression. The radial basis function (RBF) kernel given in Equation 4.8 and the histogram intersection kernel given in Equation 4.9 are ued in our experiments.

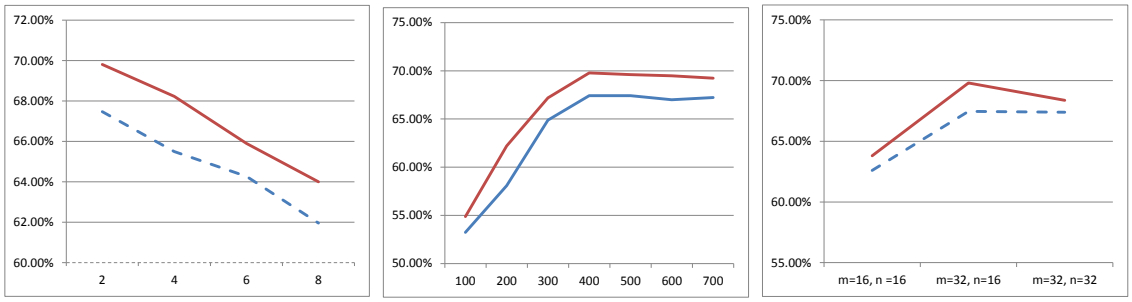
$$k(x, y) = exp\left(-\gamma \|x - y\|^2\right) \tag{4.8}$$

$$k(a, b) = \sum_{i=1}^{n} \min\left(a_i, b_i\right) \tag{4.9}$$

SVM was originally developed for binary classification. In order to extend SVM for multi-class classification, we use the One-Versus-All approach, which trains a binary classifier to classify one class of interest (positive) versus all other classes (negative). These independent SVM classifiers are used to provide seven predictions of the presence or absence of the facial expression in unseen face images and the class with the greatest class-membership probability estimation value is output as the recognized facial expression. In our experiment, the dataset is randomly divided into 6 partitions of roughly equal number of subjects belonging to each facial expression class. We use 4 partitions for training and 1 partition for estimating the parameters of the SVM classifier. After the parameters are fixed, the SVM classifier is applied to the last partition which is unseen during the training process of the classifier. The process is repeated 5 times, and the average recognition performance on the test sets are reported as the final result.

Parameter Settings

There are four parameters that need to be selected to optimize the performance of the proposed bag-of-words facial expression recognition framework. The parameter w and v introduced in Section 4.2.3 controls the dense sampling step size and vocabulary



(a) w (b) v (c) m, n

Figure 4.8: Impact of parameter settings. The y-axis denotes the recognition accuracy. The red solid line corresponds to the results obtained using Histogram Intersection kernel and the blue dotted line corresponds to the results obtained using the RBF kernel.

(codebook) size, respectively. The parameter m and n introduced in Section 4.2.3 determines the pooling neighborhood size. We study the effects of each parameter by varying their values one at a time while fixing all the others. For example, when w is being studied, we only alter the value of d and keep the value of k and r unchanged. We report the results on the CK+ database, as it is one of the most widely used benchmark in the literature.

The results reported in Figure 4.8(a) demonstrate the impacts of w. As can be seen, the recognition performance increases considerably as the sampling step size decreases from 8 pixels to 2 pixels. Since computational efficiency is not the focus of this work, a dense sample step size of 2 is used in the subsequent experiments.

The vocabulary size v determines the descriptive power of the proposed method. If v is set to small, dissimilar feature descriptors can be mapped into the same visual word leading to the decrease in recognition performance. With a large value, the method becomes more descriptive but less robust to noises as similar feature are more

Table 4.5: Frontal facial expression recognition results on CK+

	RBF	Histogram Intersection
LPP	67.47%	69.81%
SIFT	54.10%	60.66%
LBP	60.50%	60.52%
LPP (aligned)	74.60%	76.24%
LBP (aligned)	89.52%	89.52%

likely to be mapped to different visual words. In our experiments, it is found that the recognition performance is increased linearly with v, peaked when $v = 400$. From that point, the performance starts to decline for larger vocabulary sizes. The result suggests that a vocabulary size of 400 is the optimal trade-off between descriptive power and tolerance to noises.

Varying the size of m and n changes the size of the patches from where the LPP descriptor will be extracted. With a fixed sampling step size, a larger value of m or n will increase redundant information between the neighboring descriptors while a small value can cause loss of important details. Figure 4.8(c) shows their impacts to the recognition accuracy.

Frontal Facial Expression Recognition

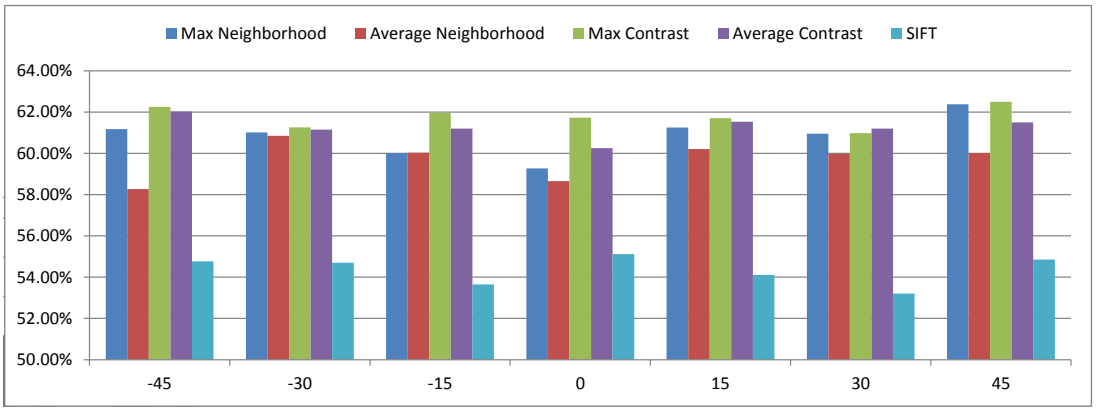
In this section, we evaluate the proposed LPP descriptor for frontal facial expression recognition. The experimental results obtained from the CK dataset are shown in the first three rows in Table 4.5. It is clear that the BOW based facial expression framework using LPP descriptor clearly outperforms the conventional SIFT and LBP

when no face alignment is performed. The result indicates that our proposed extension to the conventional SIFT descriptor is effective for facial expression recognition, especially in the presence of severe face misalignment. Another interesting observation comes from the comparisons between the two kernels used in SVM. The histogram intersection kernel consistently outperforms the RBF kernel, which indicates that it is very effective for histogram based features.

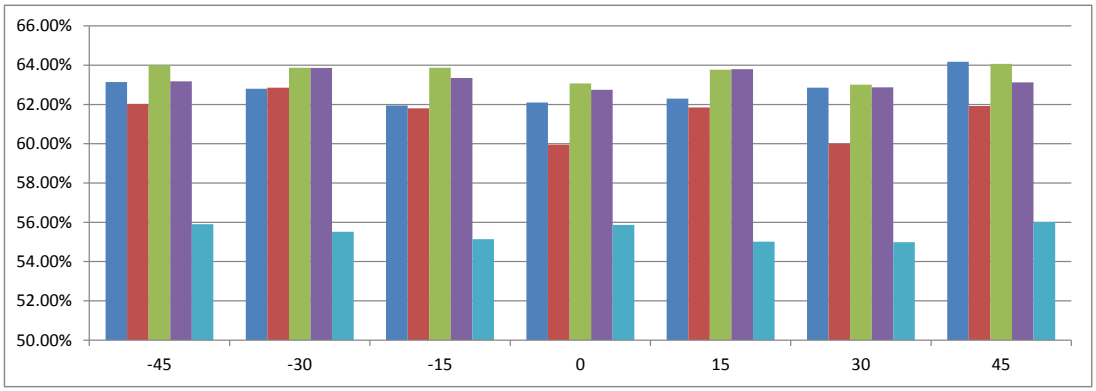
It is also worth to note that the performance of LBP is slightly worse than the other two methods. One reason is that LBP utilizes the pixel value which is sensitive to change. Additionally, LBP based facial expression representation relies on accurate face alignment. The significant misalignment existed in our testing data will change the semantic meaning of the underlying pixels thus negatively affect the recognition accuracy. To make a more fair comparison, we perform the face alignment procedures as described in Section 4.2.4. The experimental results on aligned faces are reported in the last two rows of Table 4.5. As expected, the proposed method is outperformed by the LBP based method, since it completely ignores the spatial information. However, our method performs consistently in both experiments, suggesting that it is more robust to face misalignments. This attribute makes it more suitable for practical use.

Multiview Facial Expression Recognition

The experiments are performed on the BU-3DFE dataset. The multiview facial expression images are obtained by rotating and projecting the 3D expression models, resulting in 7 different poses corresponding to ($-45°$, $-30°$, $-15°$, $0°$, $+15°$, $+30°$, $+45°$) yaw angels. Pose estimation is normally performed before recognition, so a view dependent facial expression classifier can be trained for each view individually. We test LPP descriptors obtained using different configurations defined in Section 4.2.3



(a) RBF



(b) Histogram Intersection

Figure 4.9: Multiview Facial expression recognition results on BU-3DFE. Figure 4.9(a) corresponds to the results obtained using the RBF kernel and Figure 4.9(b) corresponds to the results obtained using the Histogram Intersection kernel. (The figure is best viewed in color.)

against the conventional SIFT descriptor, and the experimental results are shown in Figure 4.9. As can be seen, the proposed method consistently outperforms the conventional SIFT. The result demonstrates the ability of our method to handle multivew facial expression images. Moreover, it is interested to note that best recognition results are obtained when faces are rotated at 45° in yaw, which is consistent to previous findings [118].

4.2.5 Summary

In this work, we present a novel texture descriptor for BOVW based facial expression recognition that are tolerant to face misalignments and pose variations. Due to the loss of spatial information, bag-of-words based representation using conventional SIFT alone does not provide adequate descriptive power to deal with the high intra-class variations and interclass similarities of facial expressions. To address the issue, we proposed a novel local texture descriptor by aggregating SIFT descriptor extracted within a neighborhood, namely Local Patch Pattern (LPP). It is designed to be more descriptive but less prone to noise. Extensive experimental results on two publicly available datasets demonstrate that loss of spatial information does not significantly decrease the performance of facial expression recognition; and conversely it allows a certain degree of freedom to face misalignments and pose variations.

CHAPTER 5

SPECTRAL EMBEDDING BASED FACIAL EXPRESSION RECOGNITION

5.1 Introduction

Automatic facial expression recognition is an active research topic with a wide range of potential applications including human-computer interactions, augmented reality and affective computing [8]. Although recent years have witnessed significant progress in the field [8], accurate recognition of facial expression remains a challenging problem, particularly for realistic facial expressions. One of the key reasons is that the high variations exist in facial expression images of the same type, which are caused by human face appearance, age, gender and ethnic groups. They are commonly observed when different people execute the same expression. Meanwhile, the problem is further hampered by high similarities among different facial expression types, which often be found when a same person executes different expression without explicit exaggeration. If the intensity of an expressions is low, the differences among facial expressions can easily be shadowed by facial appearance, thus increasing the difficulties for recognition.

Due to the high intraclass variations and interclass similarities, effective feature extraction is vital to facial expression recognition. In general, existing feature expression features can be categorized into two groups: appearance features [6, 30, 38, 39] and geometric features [42, 53, 81, 82, 83]. The appearance features model the appearance changes of faces, such as wrinkles and furrows, by directly utilizing pixel values. It can be extracted on either an entire face or local regions of a face image. Alternatively, geometry based features utilize the shape and locations of facial components (*e.g.* eyes and mouth) to represent the face geometry.

It is commonly acknowledged that different features extracted from a same pattern can reflect different characteristics of the pattern [123]. Hence, it is anticipated that the performance of facial expression recognition can benefit from multiview representations, where a view is defined as a type of feature that describes a subset of facial expression characteristics. However, there is often no obvious way to select and combine different types of features. If redundant or noisy features are chosen at the expense of discriminant features, the recognition performance can be adversely affected. To make the matter worse, facial expression features are typical of very high dimension. A simple concatenation of different features may greatly increase the computation cost and lead to inferior recognition results.

In this chapter, we present a feature selection and fusion framework for faical expression recognition based on Multiview Spectral Embedding (MSE) [123]. Inspired by the recent success of multiview features in related domains [124], our proposed framework treats feature selection and fusion as a multiview dimension reduction problem and aims to find a unified low dimensional subspace that captures information from all sources (*e.g.* different features and labels) by preserving local geometric

properties of the original features. Specifically, by assuming that facial expression features extracted from one type of expressions forms a manifold embedded in a high dimensional feature space, we construct a neighborhood graph that encodes the structure of the manifold locally. In order to maximize the discriminative power, we propose to build the neighborhood graph in a supervised manner by utilizing the label information of training data. After we combine the Laplacian matrix associated with the graph of each view with the multiview spectral embedding algorithm, a unified low dimensional feature space is obtained by performing spectral analysis of the combined matrix. Finally, a linearization method is utilized to map unseen data to the learned unified subspace for facial expression recognition.

The main contributions of our work are summarized as follows:

1. Spectral embedding based feature fusion framework is proposed to combine the appearance based and geometry based features for facial expression recognition.

2. A supervised multi-view spectral embedding algorithm is developed to achieve more discriminative embedding. By utilizing the label information of the training data, the neighborhood graph of a feature space can be constructed in a supervised manner to better capture the manifold structure of the feature space.

3. In order to solve the out-of-sample problem, we utilize a linearization method to map unseen data to the unified low dimensional subspace discovered by the MSE algorithm, where facial expression recognition can be performed.

4. We perform a comprehensive study of the widely used facial expression features, including Active Appearance Model (AAM) [85], Local Binary Pattern

(LBP) [50], Multiscale Weber Local Descriptor (Multiscale-WLD) [122], Scale-Invariant Feature Transform (SIFT) descriptor [92] and Gabor filters [58]. Extensive experimental results show that our Multiview Spectral Embedding (MSE) based multi-feature fusion method leads to clearly improved recognition performance for challenging realistic facial expressions.

The rest of this chapter is organized as follows. In Section 5.2, we review the popular facial expression features and feature level fusion techniques. In Section 5.3, the theory and method of MSE used for facial expression features fusion are presented. In Section 5.4, the facial expression features studied in our experiments are described. In Section 5.5, experimental results on three datasets are reported and discussed to demonstrate the effectiveness of the proposed method. Finally, conclusions are drawn in Section 5.6

5.2 Related Work

Feature fusion refers to the process of integrating multiple features extracted separately from different modalities into a joint and unique representation. The most intuitive approach is to simply concatenate the feature vectors from different modalities to form a new vector. However, the concatenation is not physically meaningful and leading to degrade the discriminative power of the individual feature. In addition, the structural information of each feature space is lost in such a straightforward concatenation [125].

To better exploit the complementary properties of different features, Multiple Kernel Learning has been used for feature fusion by a group of related methods [126, 127, 128, 129]. It combines different features by building base kernels for each

feature, then a weight is assigned for each base kernel indicating the contribution of the associated feature. Gehler *et al.* [126] performed MKL in a boosting manner to learn the weights of different features. Yang *et al.* [128] introduced an intermediate representation "group" between low level images and high level semantic categories, and apply MKL to find the optimal weights of each group for feature combination. In [129], a group lasso regularizer is imposed to obtain a compact feature set.

Recently, graph based spectral embedding methods have emerged as a powerful tool for feature fusion [123, 124, 130]. By assuming that similar features form a manifold embedded in the high dimensional feature space, they aim to find a united low dimensional subspace that best preserves their local neighborhood structures. They work by firstly constructing a sparse graph for each feature. Then one can construct matrices of which the spectral decomposition reveals the low dimensional structure of the manifold. Finally, to utilize the complementary properties of different features, the matrices are combined and the low dimensional subspace is represented by the eigenvectors of the resulted combined matrix. Zhang *et al.* [130] proposed a multiple feature combining algorithm based on spectral graph based manifold learning and patch alignment framework. Yu *et al.* [124] explored the complementary nature of different features with pairwise constraints.

5.3 Multiview Spectral Embedding

In this section, the principle of MSE will be reviewed followed by deriving a modified version appropriate for facial expression feature fusion. Before proceeding further, We will firstly define the notations used throughout the chapter. We use the notation similar to that of [123]. Lower case letters represent feature vectors extracted

from a single view, *e.g.*, x. Subscript n of x_n refers to the nth element of x. Capital letters represent feature matrix of a dataset, where each column refers to a feature vector, *e.g.* x. Superscript (i) of $X^{(i)}$ and $x^{(i)}$ represent data from the ith feature space.

5.3.1 Conventional Multiview Spectral Embedding

The principle of Multiview Spectral Embedding is to seek a unified low dimensional subspace that best preserves the local neighborhood structures from different views. Formally, given a multiview dataset $X = \left\{X^{(i)} = \left[x_1^{(i)}, \ldots, x_N^{(i)}\right] \in R^{m_i \times N}\right\}_1^m$ with N images and m views, MSE aims to find a low dimensional representation $Y = [y_i, \ldots, y_N] \in R^{d \times N}$, where $d < \sum_{i=1}^{m} m_i$ and m_i corresponds to the dimension of the ith view. According to the Patch Alignment Framework (PAF) [131], MSE can be divided into two stages: local patches construction and global alignment.

A local patch refers to a neighborhood formed by a feature vector and its closest related ones (*e.g.*, nearest neighbors). As the local neighborhood structures may differ from different views, local patches construction is performed on each view separately. Given an arbitrary feature vector $x_j^{(i)}$ in the ith view $X^{(i)} \in R^{m_i \times n}$, the local patch of x_j is formed by itself and its k closest related ones, *i.e.*, $X_j^{(i)} = \left[x_{j_1}^{(i)}, \ldots, {}^{(i)}_{j_k}\right]$. One local patch is computed for each feature vector on the view, resulting a total of N local patches for the ith view. For each patch $X_j^{(i)}$, there exist a corresponding low dimensional representation $Y_j^{(i)} \in R^{d \times (k+1)}$. In order to preserve the locality in the low dimensional space, MSE aims to minimize the dissimilarities (*e.g.* distance) between the given feature vector $x_j^{(i)}$ and its k neighbors. Thereby, an objective function can

be defined as:

$$\arg\min_{Y_j^{(i)}} \sum_{l=1}^{k} \|y_j^{(i)} - y_{j_l}^{(i)}\|^2 \left(w_j^{(i)}\right)_l, \tag{5.1}$$

where $(w_j^{(i)})_l$ is a weight determined by similarities between $x_j^{(i)}$ and $x_l^{(i)}$. Equation (5.1) can be rewritten as:

$$\arg\min_{Y_j^{(i)}} \operatorname{tr}\left(Y_j^{(i)} L_j^{(i)} \left(Y_j^{(i)}\right)^\intercal\right), \tag{5.2}$$

where $L_j^{(i)}$ is the Laplacian matrix that encodes the local structures of the patch and is defined as:

$$L_j^{(i)} = \begin{bmatrix} \sum_{l=1}^{k} \left(w_j^{(i)}\right)_l & -\left(w_j^{(i)}\right)^\intercal \\ -w_j^{(i)} & diag\left(w_j^{(i)}\right) \end{bmatrix}. \tag{5.3}$$

After optimal low dimensional representations are obtained for each patch from every view, global alignment is performed by summing up all the local patches as follows:

$$\begin{aligned} &\arg\min_{Y,\alpha} \sum_{j=1}^{N} \sum_{i=1}^{m} \alpha_i \operatorname{tr}\left(Y_j^{(i)} L_j^{(i)} \left(Y_j^{(i)}\right)^\intercal\right) \\ &= \arg\min_{Y,\alpha} \sum_{i=1}^{m} \alpha_i^r \operatorname{tr}\left(Y L^{(i)} Y^\intercal\right), \\ &\text{s.t. } YY^\intercal = I, \alpha_i \geq 0, \sum_{i=1}^{m} \alpha_i = 1. \end{aligned} \tag{5.4}$$

In Equation (5.4), $L^{(i)}$ is the alignment matrix of the ith view and it is defined as

$$L^{(i)} = D^{(i)} - W^{(i)} \tag{5.5}$$

where $\left[W^{(i)}\right]_{pq} = \left(w_p^{(i)}\right)_q$ and $D^{(i)}$ is a diagonal matrix with its diagonal element computed as the row sum of $W^{(i)}$. A set of m nonnegative weights $\alpha = [\alpha_1, \ldots, \alpha_m]$ are associated with the importance of each view. The larger a weight is, the more important the view plays in learning the low dimensional subspace. By directly

applying the weight α_i to each view, the optimal solution is obtained when $\alpha_i = 1$ corresponds to the ith view with minimum $\operatorname{tr}\left(YL^{(i)}Y^\intercal\right)$ and $\alpha_i = 0$ otherwise. However, the results are not desired because it only takes into account the information from a single view while the discriminating information from the other views are completely discarded. To avoid the trival solution and exploit the complementary properties of differnt features, MSE substitutes a_i to a_i^r with $r > 1$ so each has a particular contribution to the final low diemsnonal subspace.

The solution to Equation (5.4) is a nonlinearly constrained nonconvex optimization problem, and a local optimal solution can be obtained using alternating optimization by iteratively updating Y and α in an alternating fashion.

For a fixed Y, we can compute α with Lagrange multiplier. The Lagrange function is

$$L\left(\alpha, \lambda\right) = \sum_{i=1}^{m} \alpha_i^r \operatorname{tr}\left(YL_n^{(i)}Y_\intercal\right) - \lambda\left(\sum_{i=1}^{m} a_i - 1\right). \tag{5.6}$$

By setting the derivative of $L(\alpha, \lambda)$ with respect to α_i and Λ to zero, α_i can be obtained by

$$\alpha_i = \frac{\left(1/\operatorname{tr}\left(YL_n^{(i)}Y^\intercal\right)\right)^{1/(r-1)}}{\sum_{i=1}^{m}\left(1/\operatorname{tr}\left(YL_n^{(i)}Y^\intercal\right)\right)^{1/(r-1)}} \tag{5.7}$$

Afterwards, with a fixed α, Equation (5.4) is simplified to

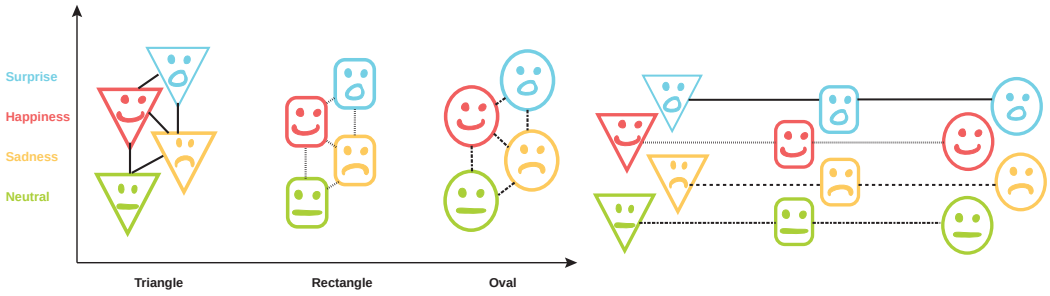
$$\min_{Y} \operatorname{tr}(YLY^T) \text{ s.t. } YY^T = I, \tag{5.8}$$

where $L = \sum_{i=1}^{m} a_i^r L_n^{(i)}$. Equation (5.8) has a global optimal solution Y, given as the eigenvectors associated with the smallest d eigenvalues of L, in which d is the predefined size of the target low dimensional subspace Y.

5.3.2 Supervised Multiview Spectral Embedding

The conventional MSE (cMSE) algorithm proposed by Xia *et al*. [123] is designed to be as general as possible with the intention of covering a wide range of applications. In this section, we derived a modified MSE algorithm specifically tailored for facial expression recognition. The contributions are twofold. First, we proposed to construct the neighborhood graph in a supervised setting to exploit the class label information of facial expression images used for training. Second, we present a way to handle unseen data points, other than rebuilding the mapping from high dimensional to low dimensional space which could be infeasible. The modified MSE algorithm is described below.

Supervised Neighborhood Graph Construction



(a) *Conventionalneighborhoodgraphconstruction.*(b) *Supervisedneighborhoodgraphconstruction.*

Figure 5.1: Illustration of conventional and supervised neighborhood graph construction methods. Assume the only available distance function is based on the face shape, the data points representing different facial expression form a neighborhood in 5.1(a) which is not intended. In contrast, the supervised graph illustrated in 5.1(b) is preferred as points depicting same type of expressions are connected by utilizing the label information (*i.e*. color).

MSE finds the low dimensional representation of a high dimensional dataset through spectral decomposition of a weighted combination of Laplacian matrices. The Laplacian matrices are associated with sparse graphs constructed from different views separately, in which the vertices represent data samples and the edges represent neighborhood relations. The underlying idea is to most faithfully preserve the local structures of the graph at each vertex, *i.e.* mapping nearby inputs to nearby outputs. Therefore, the resulting low dimensional representation is directly dependent on the neighborhood graphs used for encoding the relationships of the data samples.

The nearest neighbor graph is used in the conventional MSE algorithm, where a data sample is connected with its k nearest neighbors. However, the approach is not ideal for facial expression recognition as the neighborhood graphs built unsupervised may not capture the intended information. For example, assume there existed an imaginary 2D feature space as illustrated in Figure 5.1(a), where horizontal axis refers to the face shape and vertical axis refers to colors depicting different types of facial expressions. Consider each facial expression image as a point in this space, the distance between the data points should ideally be represented by color. Yet the only available distance function is based on face shape, which is often the case as an adequate distance function is normally hard to define. It can result in data points depicting different facial expressions to be unintentionally connected in the neighborhood graph, and accordingly change the low dimensional representation of data.

To address this problem, we exploit the class label (*i.e.* color in our imaginary feature space) of the training facial expression images by imposing a constraint that a data sample can only be connected with its k nearest neighbors from the same

facial expression class (*e.g.* happiness) when constructing the neighborhood graph as shown in 5.1(b). The objective is to minimize the distance of the data samples in the same class while separating them from semantically dissimilar ones in the obtained low dimensional space.

Linearization

The conventional MSE algorithm finds a nonlinear mapping from high dimensional space X to low dimensional space Y that are only defined on seen data. The low dimensional representation need to be rebuilt each time when a novel data sample is presented, which is not feasible for two reasons. For one, the computational complexity is squared with the number of data samples N, the problem may become impractical to solve as N increases. Even when N is small, the rebuilding can still be very time consuming and undesirable for facial expression recognition. For the other, the class labels for novel data samples are unknown, thus making it difficult to use the aforementioned supervised scheme exploiting the prior information of class identities.

In order to deal with this issue, we apply a linearization procedure to obtain a projection matrix U for embedding new image samples into the low-dimensional space for recognition. Specifically, given the multiview dataset $X \in R^{\sum_{i=1}^{m} m_i \times N}$, we seek a linear projection matrix U that can map X to the low dimension embedding $Y \in R^{d \times N}$ where $d < \sum_{i=1}^{m} m_i$, *i.e.* $Y = U^{\intercal} X$. By substituting for Y in Equation (5.4), it can be rewritten as:

$$\arg\min_{U,\alpha} \sum_{i=1}^{m} \alpha_i^r \operatorname{tr}\left(U^\intercal X L^{(i)} X^\intercal U\right), \tag{5.9}$$
$$\text{s.t. } U^\intercal X X^\intercal U = I, \alpha_i \geq 0, \sum_{i=1}^{m} \alpha_i = 1.$$

The solution to Equation (5.9) is obtained through alternating optimization as discussed in Section 5.3.1. The global optimal solution U is given as the eigenvectors associated with the smallest d eigenvalues of $XLX^\intercal$. The projection matrix U can be seen as an estimation of the transformation from the high dimension to low dimension space, which is obtained from a training set using our supervised scheme. Novel image samples can then be projected by U to the low dimensional space Y, where facial expression recognition is performed.

5.4 Facial Expression Features

The goal of feature extraction is to convert pixel data into a high-level representation of shape, motion, color, and texture of facial images. We represent facial expression images using a combination of appearance and geometric features. In order to find the optimal combination, we have selected five types of appearance features and one type of geometric feature, including Multiscale Weber Local Descriptor (Multiscale-WLD), Local Binary Pattern (LBP), Scale-Invariant Feature Transform (SIFT) descriptor, Gabor filters and Active Appearance Model (AAM). The extracted features are used for subsequent feature fusion.

The aforementioned five features have all achieved considerable success for facial expression recognition. While some of them may share certain commonalities, all of them have their own characteristics. For example, LBP, WLD and SIFT captures

micro texture patters and use a histogram to represent the pattern distribution within the block. Although the histogram based approach makes them more robust to local changes (noise), the spatial location informations are discarded to a certain degree. The Gabor filters, on the other hand, captures global shape information centered at a pixel. However, it is more sensitive to small variations in expression and noise (such as a blur at the pixel's location). AAM is on another league of its own, as it tracks the facial landmark points which can definitely be helpful for improving recognition performance. Thus, it is clear that the selected five features are able to represent facial expressions from distinct perspectives. This motivates us to propose a feature fusion framework to effectively and efficiently exploit the complementary characteristics of the features.

5.4.1 Multiscale Weber Local Descriptor (Multiscale-WLD)

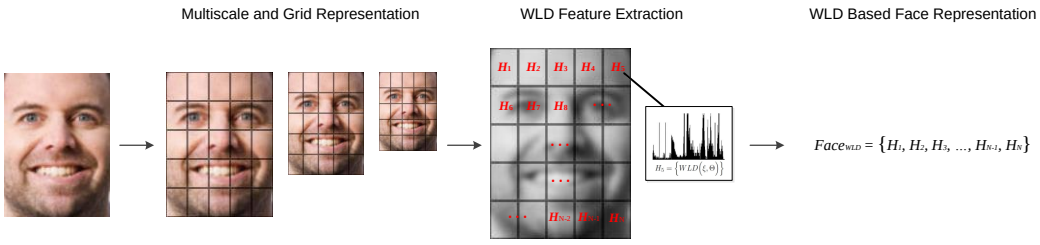


Figure 5.2: Multiscale-WLD Facial Expression Feature.

Weber Local Descriptor (WLD) characterizes texture information of an image by considering the ratio of changes in pixel intensity, which is inspired by the high sensitivity of human visual system to small changes in intensity of a stimulus. WLD

includes two components, differential excitation and orientation. Differential excitation measures the ratio of change in pixel intensity between a center pixel against its neighbors. The orientation component is computed as the gradient orientation of a pixel. After obtaining the two components for each pixel, a $2D$ WLD histogram is constructed to represent the image.

WLD is extended to allow multiscale analysis of facial expression images[122]. It is achieved by down-sampling the original face image to form an image pyramid followed by applying a WLD operator with fixed neighborhood size as shown in Figure 5.2. WLD histogram computed over a global face image does not capture the spatial locations of the micro texture patterns since the patterns tend to be averaged over the whole image area which will reduce the discriminative power. In order to overcome this issue, a face image is equally divided into a set of rectangular regions and a histogram is computed independently for each sub-region. The final Multiscale-WLD feature vector is formulated by concatenating the histograms extracted at different regions and scales. The distance between two Multiscale-WLD feature vectors reflects the extent to which two facial images contain similar micro texture structures within corresponding spatial regions. We divide a facial expression image to 5×5 blocks and computes the WLD histogram from each block individually.

5.4.2 Local Binary Pattern (LBP)

The LBP descriptor captures the first order circular derivative pattern of an image, which is a micro texture pattern generated by concatenating the binary gradient directions [49]. It labels the pixel of the image by thresholding a circular neighborhood with radius of R pixels with the center pixel t_c's value in grayscale, and considering

the results as a binary number. Formally, the LBP operator is defined as follows:

$$LBP(P,R) = \sum_{i=0}^{p-1} u(t_i - t_c)2^i,$$

where P denotes the number of pixels in the neighborhood, R is the radius in pixels of the circular neighborhood, t_i and t_c are the intensity in grayscale of the neighbor i and the center pixel respectively. $u(x)$ is a step function, *i.e.* $u(x) = 1$ when $x \geq 0$ and $u(x) = 0$ otherwise. Using circular neighborhoods and bi-linearly interpolating the pixel values allow any radius R and number of pixels P in the neighborhood [50]. The patters are called uniform if they contain at most two bitwise transitions from 0 to 1 or vice versa when the binary string is considered circular. For example, 00001100 and 11110111 are uniform patterns. Rotation invariance is achieved by recognizing that $LBP(P,R)$ originates from some particular rotation-invariant patterns. Following [53], we use the 59-bin $LBP^{u2}_{8,2}$ operator. Similar to Multiscale-WLD, each facial image is segmented into a grid of 6×7 regions. We compute a $LBP^{u2}_{8,2}$ operator for each of the 42 regions, and the concatenation of the histograms are used to represent the facial expression image.

5.4.3 SIFT

SIFT [92] characterize local gradient information over square windows centered on some interest point locations. Since we have applied AAM to locate facial landmarks defining face shapes, the 68 points obtained by AAM are used as keypoints. For each keypoint, there are three steps to calculate its SIFT descriptor. Firstly, gradient magnitudes and orientations are computed, sampled from a square region around the keypoint. Secondly, in order to eliminate the influence introduced by small changed in the position of the window, the gradient magnitudes are weighted with a Gaussian

weighting function. Thirdly, the weighted gradient magnitudes are accumulated into an orientation histogram, whose peaks are considered as the dominant orientations of the keypoint. Orientation invariance is achieved by rotating the coordinates of the descriptor and the gradient orientation relative to its dominant orientation. The final descriptor is obtained by concatenating the orientation histograms over all locations, and the vector is normalized to unit length to reduce the effects of illumination change. The facial expression images are represented using a bag of visual words model, where the visual vocabulary is generated from the SIFT descriptors.

5.4.4 Gabor Magnitude Representation

Gabor filters characterize image textures by decomposing them into different orientations and scales. It can be defined as follows:

$$\psi_{\mu,\nu}(z) = \frac{\|k_{\mu,\nu}\|^2}{\sigma^2} e^{\left(\|k_{\mu,\nu}\|^2 \frac{\|z\|^2}{2\sigma^2}\right)} \left[e^{ik_{\mu,\nu}z} - e^{-\frac{\sigma^2}{2}}\right],$$

where μ and ν define the orientation and scale of the Gabor filter respectively, $z = (x, y)$ denotes the pixel and the wave vector $k_{\mu,\nu}$ is defined as $k_{\mu,\nu} = k_\nu e^{i\varphi_\mu}$ where $\varphi_\mu = \frac{\pi\mu}{8}$ is the orientation parameter. $k_\nu = \frac{k_{max}}{f^\nu}$, where f is the spacing factor between filters in the frequency domain. The convolution of an image I with a Gabor filter $\psi_{\mu,\nu}$ is defined as $F_{\mu,\nu} = I(z) * \psi_{\mu,\nu}(z)$ where $*$ denotes the convolution operator. Gabor magnitude representation is defined as

$$M_{\mu,\nu}(z) = \sqrt{Im(F_{\mu,\nu}(z))^2 + Re(F_{\mu,\nu}(z))^2}$$

where Im denotes the imaginary part and Re denotes the real part of F. Same as [58], facial expression images are converted into Gabor magnitude representation using a bank of Gabor filters at 8 orientations and 5 spatial frequencies.

5.4.5 Active Appearance Model (AAM)

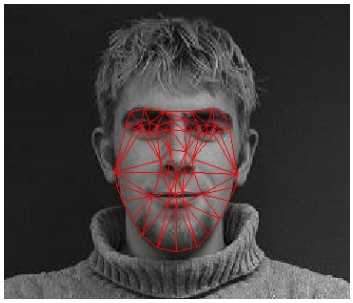


Figure 5.3: Illustration of AAM shape registration.

The AAM [85] is a generative method for modeling the shape and appearance of an object. Given a training dataset T with m labeled objects, Procrustes analysis is used to align the objects into a common coordinate system. The shapes of objects are represented by the coordinates of the vertex of a 2D triangular mesh (see Figure 5.3), and their variations are captured by the principal components of the covariance matrix of the aligned shapes. Similarly, the variations of appearance is obtained by applying PCA to the object images, warped to a canonical frame defined using the mean shape of the aligned shapes. Afterward, the shape s of a new object can be represented as $s = \overline{s} + P_s b_s$, where $\overline{s}$ is the mean shape, P_s are the principal components defining the shape variations and b_s are the shape parameters. And its appearance t becomes $t = \overline{t} + P_t b_t$, where $\overline{t}$ is the mean appearance vector, P_t encodes the appearance variations obtained from the training dataset and b_t are the appearance parameters. The parameters b_s and b_t are estimated by a gradient decent method. By varying the parameters, the AAM is able to represent large variations

in shape and appearance. Following [68], we use the coordinates of the 68 vertex defining the face shape to represent facial expressions.

5.5 Experimental Results and Discussions

5.5.1 Datasets

Three datasets are used in our experiments: JAFFE [4], Cohn-Kanade DFAT (CK) [68] and a facial expression dataset constructed using Google web images (GWI) [121, 122]. The JAFFE dataset contains 213 images of the seven basic facial expressions which were posed by 10 Japanese females. It is the most trivial dataset of the three, and serves as the baseline in the experiments. The Cohn-Kanade DFAT dataset consists of approximately 500 images from 100 subjects ranged in age from 18 to 30 years, of which 65% are female. The distribution of the ethnic groups is: 81% Euro-American, 13% Afro-American and 6% other groups. The GWI facial expression dataset constructed using Google web image search includes 50 images for each of the seven basic facial expressions. It is considered to be most challenging one as it is collected in much more diverse imaging conditions and contains a much larger variety of subjects. An overview of the evaluation datasets is shown in Table 5.1. All face images are resized to a fixed size of 110 x 150 pixels. Histogram equalization is performed to remove the illumination effect in the images.

5.5.2 Experimental Settings

The experiments were conducted on 7 categories of universal facial expressions: *happiness*, *sadness*, *anger*, *fear*, *disgust*, *surprise* and *neutral*. The multiview representation of facial expression images are constructed from the following five different features: 3600 dimensional Multiscale-WLD (M-WLD) [122], 2478 dimensional LBP

Table 5.1: Statistics of the evaluation datasets used in the experiments. The columns of Male and female show the percentage of male subjects and female subjects respectively.

	Images	Subjects	Female	Male
G	350	328	59.2%	40.8%
$JAFFE$	213	10	100%	0%
CK	500	100	65%	35%

[53], 400 dimensional SIFT features (SIFT) [68], 42650 dimensional Gabor filters [53], and 136 dimensional AAM shape features [68]. In our experiments, three MSE parameters (d, k, r) are experimentally set to 200, 11, and 8 respectively. Here d is the dimension of the embedding, k is the number of neighbors used for building the neighborhood graph, and r determines the contributions of each view to the final embedding. In Section 5.5.5, we study the impact of different parameter settings.

Face images are preprocessed to remove background noise before feature extraction. Given a face image, automatic face detection is firstly performed using the Viola-Jones face detector [2]. After faces are detected, we perform automatic eye localization on the detected face regions. In order to align different face images into a common coordinate system based on eye locations. For this task, we adopt the Average of Synthetic Exact Filters (ASEF) [96] which is a class of correlation filters. Then we align and normalize the faces based on the detected eye locations and the distance between the two eyes. Finally, facial images of 110×115 pixels are cropped from the original frames and are used in the experiments.

Before proceeding further, we provide a walk through of the facial expression recognition algorithm used for evaluating the facial expression datasets. Given a set of training images, we first extract the multiview features to represent the facial expressions. Afterward, we perform the feature fusion using the proposed framework to obtain a projection matrix, so the multiview features can be mapped to a low dimensional space. And the facial expression recognition is conducted in the obtained low dimensional subspace. Specifically, an SVM classifier is trained for each facial expression. Since SVM was originally developed for binary classification, in order to extend SVM for multi-class classification, we use the One-Versus-All approach, which trains a binary classifier to classify one class of interest (positive) versus all other classes (negative). These independent SVM classifiers are used to provide seven predictions of the presence or absence of the facial expression in unseen face images and the class with the greatest class-membership probability estimation value is output as the recognized facial expression. In our experiments, the dataset is randomly divided into 6 partitions of roughly equal number of subjects belonging to each facial expression class. We use 4 partitions for training and 1 partition for estimating the parameters of the SVM classifier. After the parameters are fixed, the SVM classifier is applied to the last partition which is unseen during the training process of the classifier. The process is repeated 5 times, and the average recognition performance on the test sets are reported as the final result.

5.5.3 Performance comparison with single view features

In this section, we compare the proposed multiview feature with the raw high dimensional single view features. The experiment results on three datasets are show

Table 5.2: Comparison of the multiview feature with single view features.

	sMSE(ours)	LBP	M-WLD	SIFT	Gabor	AAM
GWI	**62.5%**	46.1%	59.9%	40.5%	54.5%	35.7%
$JAFFE$	**85.7%**	84.1%	**85.7%**	75.4%	84.2%	77.0%
CK	**96.0%**	89.5%	95.7%	77.3%	89.2%	75.5%

in Table 5.2. As can be seen, the proposed multiview feature consistently outperform the single view features on two out of the three datasets. The result indicates that our MSE based framework is able to exploit the complementary properties of different views to obtain an effective low dimensional representation for multiview data. On the JAFFE dataset, it is a bit unexpected that the performance of the multiview feature is the same as the best single view feature (namely M-WLD). The slightly unsatisfactory performance can be attributed to the small size of the JAFFE dataset, which can increase the density of the neighborhood graph and in turn make MSE suffer from folding [132]. Moreover, the small dataset size will cause MSE more sensitive to the noise and outliers in the data. If outliers are connected to their k nearest neighbors when they are very distant, it will degrade the performance of MSE and even may result in overfitting. Another observation from Table 5.2 is that the geometry based method (AAM) is generally outperformed by appearance based features. The main reason could be that AAM relies on accurate and reliable facial feature detection which is very difficult and error prone. This becomes evident as the worst performed texture feature (SIFT) is still roughly 5% better on the GWI dataset, where the facial feature detection is the hardest to accomplish.

5.5.4 Experiment on multi-modal features fusion

In this section, we demonstrate the effectiveness of the proposed facial feature fusion framework (denoted as sMSE) by comparing it with other feature combination techniques. Similar to [123], the following methods are used: 1) best single view embedding (BSE): the best performed single view features in each dataset, Laplacian Eigenmap (LE) is used for dimension reduction; 2) average single view embedding (ASE): the average performance of the single view based spectral embedding using LE; 3) conventional MSE (cMSE): the conventional MSE. To better understand the capability of our frame, we also include these additional methods: 4) concatenated features (CF): concatenate all feature vectors from different views to form a new vector, normalization is performed before the concatenation; 5) LE-CF: applying LE to the concatenated features to obtain the low dimensional subspace; 6) LLE-CF: LLE is used to the concatenated features to build the low dimensional subspace.

The experimental results are reported in Table 5.3. As expected, the proposed method achieves the best results on CK and GWI. The performance is slightly inferior to the best single view embedding on $JAFFE$ due to the problem of small dataset size as discussed in Section 5.5.3. However, our method constantly outperforms the conventional MSE on all three datasets. The result suggests that our modifications are effective for facial expression recognition. It is also worth noted that the concatenation based methods performs the worst on all three datasets, which confirms that a simple concatenation of feature vectors cannot effectively explore the complementary nature of different views.

Table 5.3: Comparison of different feature fusion techniques.

	sMSE(ours)	BSE	ASE	cMSE	CF	LE-CF	LLE-CF
GWI	**62.5%**	59.9%	51.4%	60.7%	45.2%	52.0%	52.5%
JAFFE	85.7%	**85.8%**	80.8%	85.6%	70.8%	80.5%	81.4%
CK	**96.0%**	95.5%	88%	95.7%	82.0%	88.0%	90.3%

5.5.5 Impact of Parameter Settings

In this section, we study the impacts of the parameter settings of d, k and r. The experiments are performed by varying their values one at a time while fixing all the others. For example, when d is being studied, we only alter the value of d and keep the value of k and r unchanged. The CK dataset is used for training and testing in a 5-fold cross validation manner.

The parameter d controls the dimensions of the low dimensional subspace. In order to investigate its effects on the recognition performance, we vary the value of d from 50 to 500 by a step of 50. The experimental result is reported in Figure 5.4(a). It shows that the recognition performance is gained linearly with the value of d, and the progress continues as d approaches to 200. When d becomes larger than 200, there is no significant improvements in performance as d increases, but the computational cost is greatly increased. The result suggests that the dimensionality of 200 is adequate for facial expression representation, where the discriminative power and computational cost are well balanced.

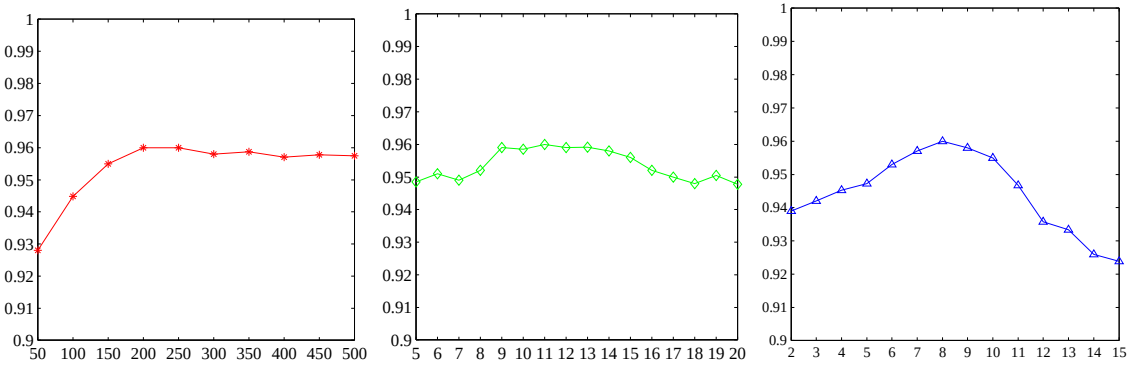
The parameter k determines the number of neighbors used for constructing the neighborhood graph. If k is set too small, the neighborhood graph will be prone to

the noise data within in the neighborhood. On the other hand, if it is set too large, the local linearity assumption (*i.e.* each each data point and its k nearest neighbors from the same facial expression class lie on a linear manifold locally embedded in the high dimensional space) will be violated, resulting in an ineffective representation of the manifold. In our experiment, the value of k is varied in the range between 5 and 20 with an increment of 1. Figure 5.4(b) demonstrates the relationship between recognition performance and the value of k. As can be seen, there is initially a small increase in recognition performance which is peaked at $k = 11$. The performance keeps stable as k changes from 11 to 14, and starts to decline when $k > 14$. The result suggests that the optimal value of k is 11.

The parameter r is introduced in Equation (5.9) to avoid a trivial solution of the objective function. Paired with the weights of different views, it decides the contribution of each view to the final low dimensional embedding. Theoretically, a smaller r exaggerates the contribution from the most discriminative view, and a larger r makes the contributions from all views similar. Therefore, the selection of r should be based on the complementary characteristics all all views. Rich complementary multiview features favor a large r, otherwise r should be small. Figure 5.4(c) shows that the complementary properties of the five features used in our experiments can be best exploited when r is set to 8.

5.6 Conclusion

In this chapter, we presented a novel facial expression recognition method by fusing multiple features with spectral embedding techniques. Based on the existing multi-view embedding algorithm, we propose a supervised implementation by taking



(a) d (b) k (c) r

Figure 5.4: Impact of parameter settings. The y-axis denotes the recognition accuracy.

label information of training data into account. In addition, a linearization method is utilized for handling the out-of-sample problem. In our studies, five widely used facial expression features, namely AAM, LBP, Multiscale-WLD, SIFT and Gabor magnitude features, have been investigated. Extensive experimental results on two benchmark datasets and one challenging web image dataset have demonstrated that the resulted multi-view feature leads to clearly improved recognition performance on the challenging realistic facial expressions, which shows the effectiveness of the proposed feature fusion framework.

CHAPTER 6

TOWARDS VIDEO BASED FACIAL EXPRESSION RECOGNITION

6.1 Introduction

Nowadays automatic facial expression recognition has attracted significant attention because of its usefulness in a wide range of applications such as human-computer interaction (HCI), multimedia indexing and retrieval, image or video understanding and clinical research, since facial expression plays a critical role in our social communication[8]. Previous works on 2D facial expression recognition can be classified into two categories, single image (i.e. frame) based and image sequence (i.e. video) based [8]. Image based methods typically assume that facial expression recognition can be performed on one representative facial expression image. Alternatively, image sequence based methods treat each facial expression as a temporally dynamic process and aim to explore its dynamic characteristics besides the appearance in each facial expression frame.

Due to the diverse imaging devices, various facial expression videos could have different frame rates as well spatial resolutions. As a result, one algorithm workable for one type of facial expression videos may not work very well for other types since

some features (e.g. distance and angle between tracked facial points) highly depend on the smoothness of image sequences. As indicated in [133], frame rate does influence recognition performance of micro-expressions and a temporal interpolation method was proposed to solve such frame rate issue. However, there are few studies on the impact of the frame rates of image sequences on facial expression recognition performance. In addition, the psychology study reveals that human beings are able to readily recognize facial expressions from very short sequences [134]. Similar conclusion was also drawn in human action recognition that only a subset of frames was enough to achieve a performance similar to the one obtained from all the frames of an action sequence [135], though how to select the right frames was not addressed.

Therefore, in this chapter we systematically investigate two issues, 1) how the number of frames of a facial expression sequence influences facial expression recognition accuracy, and 2) how to choose a set of appropriate frames (i.e. keyframe). Uniform down-sampling of a given facial expression sequence is feasible, however, not always effective due to the ignorance of expression dynamics. Therefore, we assume that facial expression is characterized with a number keypoints and formulate keyframe selection as identifying a set of frames which best covers the whole facial expression sequence. Firstly, each frame is represented with a number of keypoints. Secondly, keypoints of all frames are matched and traced to form a global keypoint pool to characterize a facial expression sequence. Finally, a set of frames is selected by maximizing their coverage against the global keypoint pool and minimizing the redundancy among them. Our work will focus on the recognition of six prototypic emotional expressions, *i.e.* *happiness*, *sadness*, *surprise*, *anger*, *fear* and *disgust*.

6.2 Keypoint Based Keyframe Selection

In this section, we present the keypoints based keyframe selection framework [136]. The framework consists of three steps: 1) keypoints extraction, 2) keypoint pool reconstruction through keypoint matching and chaining, and 3) keyframe selection.

Firstly, for a given video sequence, keypoints are identified from each frame and descriptors are extracted for each keypoint. Lowe's SIFT descriptor [92] is utilized for keypoint extraction and representation. Hence, each keypoint is represented with a $4 \times 4 \times 8 = 128$ dimension feature vector, a 4×4 array of orientation histograms with 8 orientation bins in each.

Secondly, a keypoints pool is formed to represent the video sequence through keypoint matching. In order to reduce computational cost, we adopt a matching strategy that considers only those candidate keypoints within a certain radius R of the target keypoint for potential matching. We present an Inter-window Keypoint Chaining scheme for keypoint mathcing, where keypoints are matched within a temporal window of size W and chained across multiple windows. Assume there are three frames, f_i, f_j and f_m, as shown in Figure 6.1. When a keypoint k_1 in frame f_i is matched with another keypoint k_2 in frame f_j, and the same keypoint k_2 is matched with a third keypoint k_3 in frame f_m, satisfying $|i - j| <= W$ and $|m - j| <= W$, we link these matches into a chain, which would finally contribute to the same unique keypoint in the global pool K without matching keypoints between f_i and f_m (i.e. pairing f_i and f_m). In our experiments, R is empirically set to 100, W to 5.

We also propose Intra-Window Keypoint Chaining. As shown in Figure 6.2, k_1 is matched with k_3 but not k_2, and k_2 is matched with k_3. In this case, k_1, k_2 and k_3 will also be linked by a single chain, which could ease the problem of missed

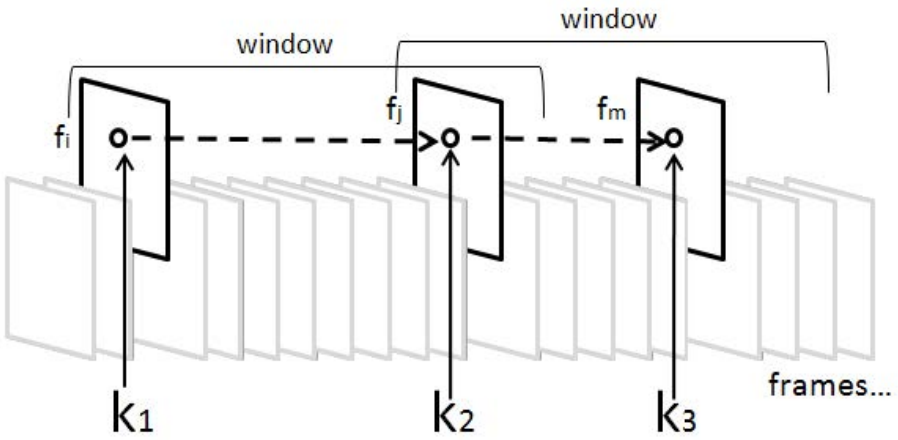


Figure 6.1: Inter-window chaining of keypoints.

matching (*e.g.* k_1 should be a true match with k_2). After the keypoint chaining on frames, each keypoint either belongs to a chain of matched keypoints or becomes an singleton without any connection. All singleton keypoints are removed. Each chain is represented by its starting keypoint and the number of keypoints on that chain, denoted by (k_x, N_x), where k_x is the starting keypoint of x-th chain and N_x is the number of keypoints in the chain. The global pool of keypoint K is then formed by aggregating all (k_x, N_x).

To remove false-positive matches, the RANSAC algorithm [137] is iteratively invoked to detect sets of geometrically consistent keypoint matches. This process is repeated until no further large set of matches (*e.g.* five matches in a group) can be found.

Finally, keyframes are selected based on the criteria that the keypoints of those frames should cover the keypoint pool as much as possible. A greedy algorithm is used for this task. Specifically, the frame with the highest number of keypoints against the

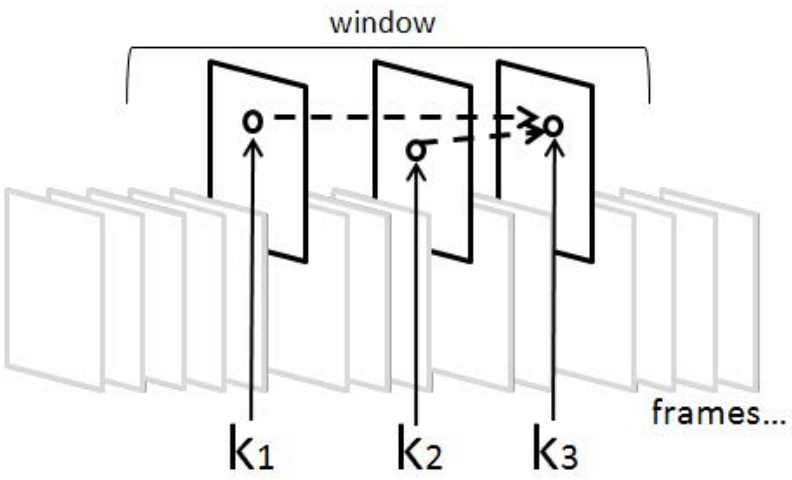


Figure 6.2: Intra-window chaining of keypoints.

keypoint pool is chosen as the first keyframe. Then at each iteration, a frame is chosen based on two metrics, namely Coverage and Redundancy. Assume the keypoint pool is divided into two sets, $K_{covered}$ and $K_{uncovered}$. At the beginning of the process, $K_{uncovered}$ contains all keypoints in K and $K_{covered}$ is empty. For frame f_i, denote its keypoint set as FP_i, *Coverage* and *Redundancy* are computed as:

$$C(f_i) = |FP_i \cap K_{uncovered}|. \tag{6.1}$$

$$R(f_i) = |FP_i \cap K_{covered}|. \tag{6.2}$$

A frame is ranked as (α is set to 1 empirically in the experiments):

$$Influence(f_i) = C(f_i) - \alpha R(f_i) \tag{6.3}$$

At the end of each iteration, the frame with the highest influence value and positive coverage will be selected as a keyframe, and all new keypoints will be moved from

$K_{uncovered}$ to $K_{covered}$. The iteration repeats until all or a predefined percentage of coverage $STOP$ of the pool K are covered.

6.3 Facial Expression Recognition Method

As discussed in Section 6.1, the main focus of this chapter is to investigate the impact of the number of frames in a facial expression sequence on facial expression recognition accuracy, instead of developing novel facial expression recognition algorithms. Hence, we adopt an robust facial expression recognition method proposed by Zhao *et al*. [100] for our experiments and briefly describe it in the following sections.

6.3.1 Face Registration

Given a facial expression image sequence, face detection is firstly conducted on all frames in the sequence using Viola-Jones face detector [2]. Secondly, we perform automatic eye localization on the detected face region by utilizing a class of correlation filters, the Average of Synthetic Exact Filters (ASEF) [96]. The purpose is to align different face image data into a common coordinate system based on eye locations. Finally, we align and normalize the faces based on the detected eye locations and the distance between the two eyes. Facial images of 110×115 pixels are cropped from the original frames.

6.3.2 Facial Feature Extraction

In our experiments, we use LBP [49] and LBP-TOP [100] for image based and video (image sequence) based recognition, respectively. The two features are selected becaused they are one of the most representative and top-performing methods in the field. The LBP [49] descriptor labels the pixel of an image by thresholding a

neighborhood of pixels with the center value in grayscale and considering the results as a binary number (see Figure 6.3). The histogram of LBP labels computed over a region is used as a texture descriptor. For image based recognition, we use the 59-bin $LBP^{u2}_{8,2}$ operator [100]. Each facial image is segmented into a grid of 6×7 overlapping regions, which is the best partition discovered in [100]. We compute a $LBP^{u2}_{8,2}$ operator for each of the 42 regions, resulting a 2478-dimensional feature vector for each face image.

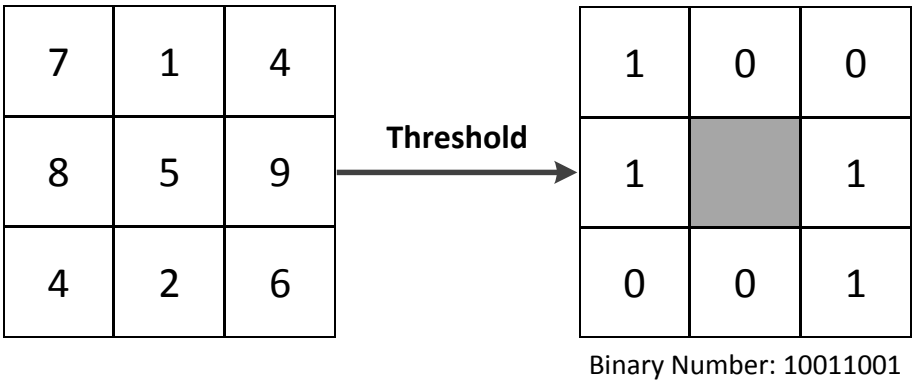


Figure 6.3: Basic LBP operator.

LBP-TOP extends LBP into three planes by computing LBP from Three Orthogonal Planes, XY, XT and YT as shown in Figure 6.4, where X-Y represents spatial plane and T represents temporal axis. The three LBP histograms computed on the three planes are computed and concatenated into a single histogram. Following [100], we use the 177 bin $LBP-TOP^{u2}_{8,8,8,3,3,3}$ descriptor in the experiments. Similar to LBP, each face image from a sequence is also divided into 6×7 overlapping

regions. A $LBP - TOP^{u2}_{8,8,8,3,3,3}$ is computed for each of the 42 regions, yielding a 7434-dimensional feature vector.

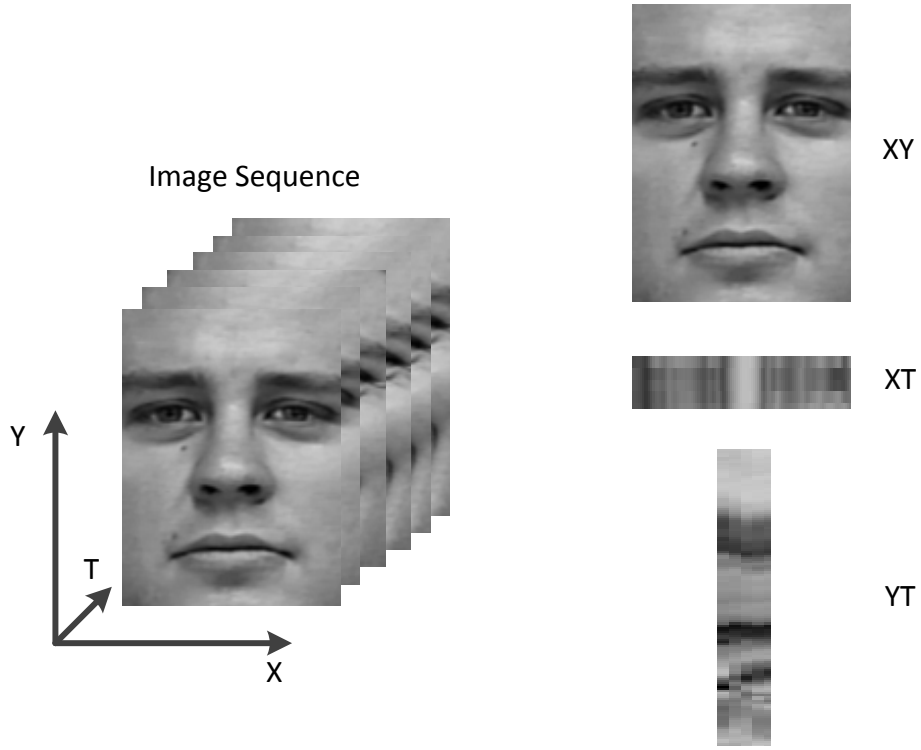


Figure 6.4: Three Orthogonal Planes, XY, XT and YT.

6.3.3 Classifier

We treat facial expression recognition as a classification problem to be solved by SVM classifiers. The radial basis function (RBF) kernel is used since it provides the best performance in our experiments. SVM was originally developed for binary classification. In order to extend SVM for multi-class classification, we use the One-Versus-All approach, which trains a binary classifier to classify one class of interest

(positive) versus all other classes (negative). Six binary SVM classifiers are learned for each of the six facial expressions. The independent classifiers are used to provide six predictions of the presence or absence of the facial expression in unseen face images and the class with the largest class-membership probability is output as the recognized facial expression.

6.4 Experiments

6.4.1 Experimental Settings

Figure 6.5: Sample facial expression images from the CK dataset. In clockwise order, *happiness*, *sadness*, *fear*, *disgust*, *surprise* and *anger*.

We use the Cohn-Kanade facial expression database [63] for our evaluation. The Cohn-Kanade DFAT database, which has been the de-factor benchmark dataset for facial expression recognition, consists of approximately 500 image sequences from 100

subjects ranged in age from 18 to 30 years, of which 65% are female. The distribution of the ethnic groups is: 81% Euro-American, 13% Afro-American and 6% other groups. The subjects were instructed to perform a series of 23 facial displays that include single and combinations of action units, six of which are based on descriptions of prototypical emotions, *anger*, *disgust*, *fear*, *happiness*, *sadness* and *surprise*. A sample image of each expression is shown in Figure 6.5. Each expression sequence was captured at frame rate 30 frames per second.

In our study, a sequence is chosen if it is one of the six basic emotions and has more than 10 frames. In total, 310 sequences from the dataset are selected. In order to minimize the influence caused by background noise, each face is preprocessed (e.g. registration and cropping) as outlined in Section 6.3.1. Histogram equalization is further applied to reduce the illumination difference among those images. In our experiments, the coverage rate is set to 100% meaning that the selected keyframes cover all the keypoints of the keypoint pool. We study the recognition accuracy using the selected keyframes, which is defined as the ratio of the number of correctly classified facial expression sequence to the total number of sequences in the test database.

6.4.2 Keyframe Selection

As shown in Figure 6.6 and 6.7, each expression evolves from the neutral status (the first frame) to the peak status (the peak frame). However, difference between adjacent frames are hardly noticeable. As highlighted by red rectangles, the selected keyframes are representative of different stages for a given expression sequence. It is also observed from the selection order of those keyframes that the first two frames

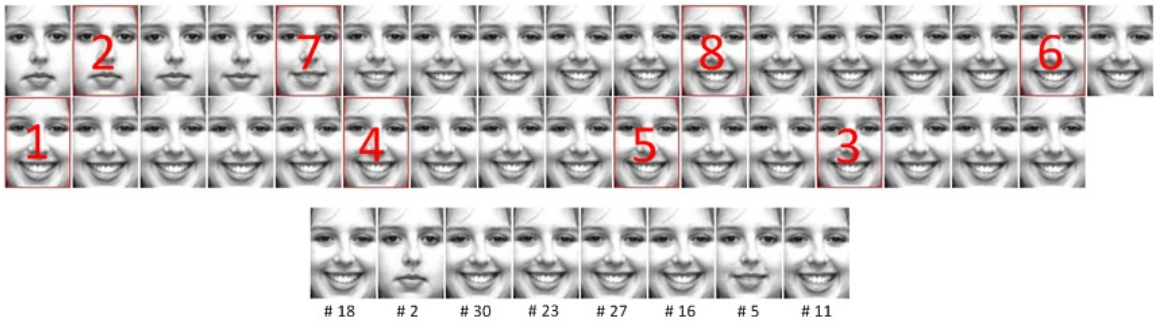


Figure 6.6: Keyframe selection results for s052_004. The original sequence (the upper part) is organized in temporal order from left to righ and top to bottom. The selected keyframes are highlighted with red rectangles. The overlay numbers indicate the selection order. Below each original expression sequence, the keyframes are organized in the selection order. The numbers at the bottom are the frame numbers of the original sequence.

corresponds to early neutral status and late peak status. When more frames are allowed, the intermediate expression status will be better depicted.

6.4.3 How Many Frames?

Table 6.1: Comparison of recognition accuracy with keyframes, uniformly sampled frames, and the whole sequence frames.

	Keyframes	Uniform	Whole
Avg. # of Frames	7.51	10	20.22
Recognition Accuracy	85.13%	85.21%	88.21%

In this section, we compare expression recognition performance between a subset of expression sequences and all expression frames and demonstrate the advantage of

keyframe selection over uniform sampling on expression recognition. Uniform sampling is to uniformly partition the whole sequence into $n-1$ segments, where n is the number of frames to be selected. Then the first frame of each segment and the last frame of the sequence will be chosen as the uniformly sampled frames.

After keyframe selection, the number of selected frames of each expression sequence varies from 4 to 8. In average, the length is reduced from original 20.22 frames per sequence to 7.51 frames per sequence. We also choose 10 frames through uniform sampling from each sequence for the comparison. As shown in Table 6.1, recognition performance with 10 uniformly sampled frames is comparable to that obtained with all the frames. Particularly, recognition performance will maintain at the same level, while the average number of frames used for recognition is further reduced to 7.51 through our keyframe selection method. Therefore, a set of appropriately chosen frames from an expression sequence is sufficient to achieve promising recognition accuracy.

We further perform recognition experiments with different number of target frames, n, from 3 (the minimum requirement for feature extraction with LBP-TOP) to 10. Note that the average number of keyframes may be less than the target figure. For example, when the target number is 6, it is true that 6 frames will be uniformly chosen from a given expression sequence. However, through keyframe selection method, one sequence could be covered with 4 frames to achieve 100% coverage rate.

As shown in Figure 6.8, recognition accuracy increases while more frames are chosen, which indicates using more frames is helpful in facial expression recognition. However, improvement slows down when the number of frames chosen is around 8.

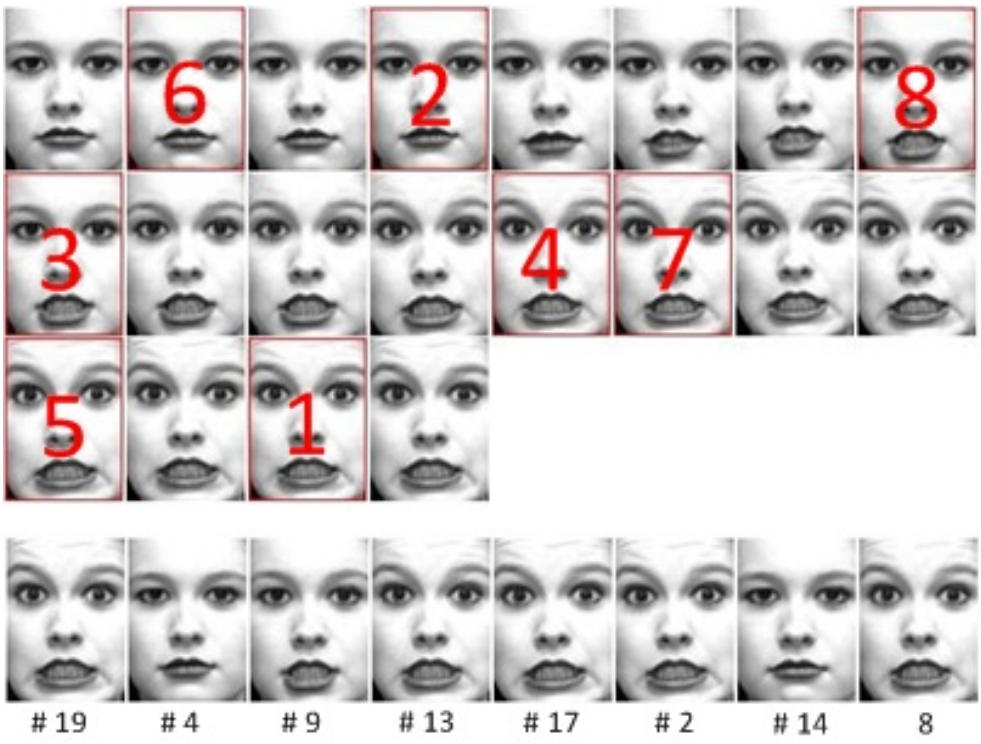


Figure 6.7: Keyframe selection results for s074_001

Therefore, there is a critical point we should balance performance and computational resource.

It is also noticed that recognition performance is better with keyframes than with uniform sampling for the same given number of frames, which indicates that our keyframe selection method is able to choose more representative frames than uniform sampling. It is because uniform sampling selects the frames uniformly without considering specific characteristics of different expressions. Note that in Figure 6.8 the curve of Keyframe terminates earlier than that of Uniform Sampling, since the number of frames obtained with keyframe selection algorithm under full coverage is

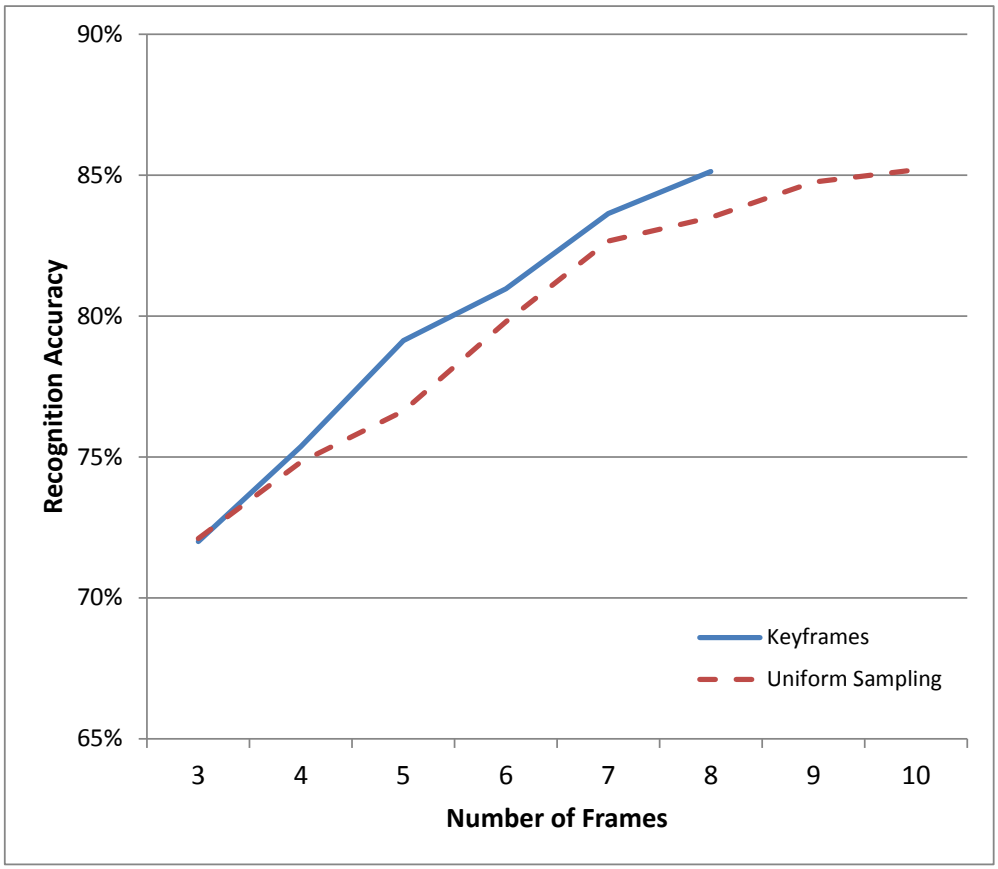


Figure 6.8: Comparison of recognition results with keyframes and with uniform sampling under different numbers of target frames. Since the number of frames obtained through keyframe selection algorithm under full coverage is fewer than 9, the results of Keyframe-based approach is not available at 9 and 10.

fewer than 9. This indicates that redundancy exists in the whole sequence and it is feasible to achieve comparable recognition accuracy with fewer frames.

6.4.4 Single Frame: Keyframe vs Peak frame

As shown in Figure 6.5, it is possible for human beings to recognize a facial expression from one sample image. The question is how to choose the right frame. In the CK dataset, an expression is captured from neutral status (i.e. neutral frame)

Table 6.2: Image based facial expression recognition results of using a single keyframe and the peak frame (i.e. the last frame).

	Keyframe	Peak-frame
Recognition Accuracy	91.36%	92.01%

to peak status (i.e. peak frame) in a controlled manner. The peak frame is generally used for image based facial expression recognition. Our keyframe selection method is able to select the frame with the highest number of keypoints against the keypoint pool. It is noticed in our experiments that the first two selected frames generally include a frame close to the neutral frame and one frame close to the peak frame. Therefore, we choose one frame from the first two selected frames, which is temporally later than the other. The accuracy of the peak-frame selection is 96.1%.

As shown in Table 6.2, the recognition performance with the keyframe automatically selected by our algorithm is very close to that with manually controlled peak frame. This provides further evidence that our keyframe selection method is very good at identifying representative frame from a sequence.

It is also interesting to observe that the image based facial expression recognition accuracy is higher than the image sequence based one. This could be explained with the difference between two features utilized for each task respectively, LBP and LBP-TOP. The feature dimension of LBP-TOP is twice more than that of LBP, which could impose difficulty to SVM classifier. As seen in [100], LBP-TOP based facial expression recognition accuracy could be improved to more than 90% with some techniques.

6.5 Conclusion

In this chapter, we systematically study the impact of the length of facial expression sequences (in terms of the number of frames) on the performance of facial expression recognition with the widely used CK dataset. It has been shown that a subset of the whole expression sequence (e.g. half of the frames) is sufficient to achieve comparable recognition accuracy to that of the whole sequence. In addition, the length can be further reduced without clearly compromising recognition performance, when our well designed keyframe selection method replaces the conventional uniform down-sampling scheme. Therefore, it is necessary to balance the performance gain and excessive data, which will guide our future work on extracting features robust to diverse temporal attributes.

CHAPTER 7

CONCLUSION

This thesis investigates automatic facial expression recognition, with particular interests in its applications in realistic environments. Three important issues have been addressed: collecting a large amount of realistic training data from the Web, devising several novel facial features and a multi-modal recognition algorithm, and exploring facial expression recognition in image sequences. This chapter summarizes the main contributions and suggests directions for future works.

7.1 Main Contributions

7.1.1 Search based framework for collecting realistic training data from the Web

A large amount of realistic training data is required to develop robust facial expression algorithms. However, obtaining such data is a tedious and time consuming task that requires tremendous efforts which are proportional to the dataset size. In Chapter 3, we propose a search based framework to harvest realistic facial expression images from the web. By adopting an active learning based method to remove noisy

images from search results returned by commercial search engines, the proposed approach minimizes the human efforts during the dataset construction and maximizes the scalability for future research.

7.1.2 GWI - A realistic facial expression dataset

A realistic facial expression dataset (namely GWI) is collected based on the text based image search results from Google. The dataset contains a very diverse set of human subjects and imaging environments. Comprehensive experiments have been performed to demonstrate its distinguish characteristics compared with other widely used datasets in the literature that were collected in strictly controlled environments, which indicates the necessity to further advance facial expression recognition with more realistic datasets.

7.1.3 Novel facial expression features

Due to the high intraclass variations and interclass similarities, effective feature extraction is vital to facial expression recognition. The extracted features should represent different types of facial expressions in a way which is not significantly affected by age, gender, or appearance of the subjects. It is also desirable to have features which are robust to face localization errors and occlusions. In Chapter 3, Multiscale-WLD (MWLD) is proposed to represent facial images by its local textures and the spatial layout of the textures. The spatial layout is captured by partitioning a face image into grids and each grid is represented with WLD descriptor to capture its local textures. Additionally, Chapter 4 presents two novel facial expression features. Spatially enhanced Local Binary Pattern (SLBP) considers the spatial distribution

of the Local Binary Patterns (LBP) using a shape context based method. And Local Patch Pattern (LPP) combines the local distributions of textures extracted from neighboring patches for more robust feature representation.

7.1.4 Spectral embedding based multi-modal facial expression recognition algorithm

It is commonly acknowledged that the performance of facial expression recognition can benefit from a combination of multiple features. However there is often no obvious way to select and combine different types of features. In Chapter 5, a spectral embedding based feature combination algorithm is proposed for multi-modal analysis of facial expressions. By assuming that facial expression features extracted from one type of expressions forms a manifold embedded in a high dimensional feature space, a neighborhood graph is constructed to encode the structure of the manifold locally. After the Laplacian matrices associated with the neighborhood graph from each view are combined, a unified low dimensional feature space is obtained by performing spectral analysis of the combined matrix. The experimental results obtained using a set of geometry and texture features clearly demonstrate the effectiveness of the proposed feature fusion framework on realistic facial expression data.

7.1.5 Keypoint based frame selection for facial expression recognition

Systematic investigations are performed on how the number of frames in a facial expression sequence can affect the accuracy of facial expression recognition. And we utilized a keyframe selection method through keypoint based frame representation. Experimental results on the popular CK facial expression dataset indicate that

recognition accuracy achieved with half of the sequence frames is comparable to that utilizing all the sequence frames. Our key frame selection method can further reduce the number of frames without clearly compromising recognition accuracy which indicates that it is important to derive framerate invariant visual features to characterize facial expressions.

7.2 Future Work

7.2.1 Dynamic extensions of the proposed facial expression features

Facial expressions are dynamic and occur over time. This dynamic nature of facial expressions is important for the recognition process, as well as for making finer distinction among facial expression categories [8]. The facial expression features presented in this thesis focus on 2D facial expression images. We plan to extend them to spatial-temporal to take full advantage of the motion information.

7.2.2 Trajectory based representation of facial expression sequence

In our keyframe based representation of facial expression sequence presented in Chapter 6, a face is viewed as a whole without considering its individual components, *e.g.* eyes, mouth, *etc*. However, a facial expression is comprised of complex interactions between muscle movements of different facial components. Due to the low intense of the muscle movements, the difference caused by facial expressions can often be masked by other facial appearances (such as identity). We plan to investigate trajectory based representation based on different facial landmarks which may

exaggerate the difference and potentially improve the facial expression recognition performance.

BIBLIOGRAPHY

[1] Guillaume Duchenne. *Mcanisme de la physionomie humaine*. J.-B. Bailliere, 1862. <xiv, 7, 8>

[2] Paul Viola and Michael J. Jones. Robust Real-Time Face Detection. *International Journal of Computer Vision*, 57(2):137–154, May 2004. <xiv, 10, 11, 12, 18, 34, 71, 74, 80, 106, 118>

[3] D. Vukadinovic and M. Pantic. Fully automatic facial feature point detection using Gabor feature based boosted classifiers. In *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, pages 1692–1698, 2005. <xiv, 13>

[4] M.J. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba. Coding facial expression with Gabor wavelets. In *IEEE International Conference on Automatic Face & Gesture Recognition*, pages 200–205, 1998. <xiv, 13, 17, 23, 44, 67, 105>

[5] Ya Chang, Changbo Hu, Rogerio Feris, and Matthew Turk. Manifold based analysis of facial expression. *Image and Vision Computing*, 24(6):605–614, June 2006. <xiv, 14>

[6] Y. Tian, T. Kanade, and J. Cohn. Recognizing action units for facial expression analysis. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 23 (2):97–115, 2001. <xiv, 14, 15, 16, 21, 29, 72, 89>

[7] A. Hadid, M. Pietikainen, and T. Ahonen. A discriA discriminative feature space for detecting and recognizing faces. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2004. <xiv, 18>

[8] Zhihong Zeng, Maja Pantic, G.I. Rosman, and T.S. Huang. A Survey of Affection Recognition Methods: Audio, Visual, and Spontaneous Expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1):39–58, January 2009. <1, 7, 24, 68, 88, 113, 132>

[9] Paul Ekman and Wallace V. Friesen. Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, 17(2):124–129, February 1971. <7, 19>

[10] M. Suwa, N. Sugie, and K. Fujimora. A preliminary note on pattern recognition of human emotional expression. In *International Joint Conference on Pattern Recognition*, pages 408–410, 1978. <7>

[11] Ashok Samal and Prasana A. Lyengar. Automatic recognition and analysis of human faces and facial expressions: a survey. *Pattern Recognition*, 25(1):65–77, January 1992. <7>

[12] Maja Pantic and Leon J.M. Rothkrantz. Automatic Analysis of Facial Expressions: The State of the Art. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 22(12):1424–1445, December 2000. <7>

[13] Beat Fasel and Juergen Luettin. Automatic facial expression analysis: a Survey. *Pattern Recognition*, 36(1):259–275, January 2003. <7>

[14] Ying-Li Tian, Takeo Kanade, and Jeffrey F. Cohn. Facial Expression Analysis. In *Handbook of Face Recognition*, pages 247–275. Springer New York, 2005. <7>

[15] M.H. Yang, D. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002. <10>

[16] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998. <10>

[17] K. K. Sung and T. Poggio. Example-based learning for view-based human face detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 20 (1):39–51, 1998. <10>

[18] H. Schneiderman and T. Kanade. Object detecion using the statistics of parts. *International Journal of Computer Vision*, 56(3):151–177, 2004. <10, 12>

[19] Yoav Freund and Robert E. Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boostin. *Journal of Computer and System Sciences*, 55:119–139, 1997. <10>

[20] M. Jones and P. Viola. Fast Multi-view Face Detection. Technical report, Mitsubishi Electric Research Laboratories, 2003. <11, 12>

[21] S.Z. Li and Zhenqiu Zhang. FloatBoost learning and statistical face detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 26(9):1112–1123, September 2004. <11, 12>

[22] Chang Huang, Haizhou Ai, Yuan Li, and Shihong Lao. High-Performance Rotation Invariant Multiview Face Detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 29(4):671–686, April 2007. <11, 12>

[23] P. Dollar, Z. Tu, H. Tao, and S. Belongie. Feature mining for image classification. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007. <12>

[24] M.T. Pham and T.J. Cham. Fast training and selection and Haar features using statistics in boosting-based face detection. In *Proceedings of IEEE International Conference on Computer Vision*, 2007. <12>

[25] J. Wu, S.C. Brubaker, M. Mullin, and J. Rehg. Fast asymmetric learning for cascade face detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 30(3):369–382, 2008. <12>

[26] M.T. Pham and T.J. Cham. Online learning asymmetric boosted classifiers for object detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2007. <12>

[27] B. Heisele, T. Serre, and T. Poggio. A component-based framework for face detection and identification. *International Journal of Computer Vision*, 74(2): 167–181, 2007. <12>

[28] D. Keren, M. Osadchy, and C. Gotsman. Antifaces: A novel fast method for image detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 23(7):747–761, 2001. <12>

[29] M. Osadchy, Y. LeCun, and M. Miller. Synergistic face detection and pose estimation with energy-based models. *Journal of Machine Learning Research*, 8:1197–1215, May 2007. <12>

[30] Z. Zhang, M.J. Lyons, M. Schuster, and S. Akamatus. Comparison between geometry-based and Gabor-wavelets-based facial expression recognition using multi-layer perceptron. In *IEEE International Conference on Automatic Face & Gesture Recognition*, 1998. <14, 17, 29, 72, 89>

[31] M. Pantic and L. Rothkrantz. Expert system for automatic analysis of facial expression. *Image and Vision Computing*, 18(11):881–905, 2000. <14>

[32] R.E. Kaliouby and P. Robinson. Real-time inference of complex mental states from facial expressions and head gestures. In *IEEE CVPR Workshop on Real-time Vision for HumanCComputer Interaction*, 2004. <14, 21>

[33] M. Pantic and L.J.M. Rothkrantz. Facial action recognition for facial expression analysis from static face images. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 34(3):1449–1461, June 2004. <14>

[34] M. Pantic and I. Patras. Dynamics of facial expression: recognition of facial actions and their temporal segments from face profile image sequences. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 36(2): 433–449, April 2006. <14, 19>

[35] T.F. Cootes, C.J. Taylor, D.H. Cooper, and J. Graham. Active Shape Models - Their Training and Application. *Computer Vision and Image Understanding*, 61(1):38–59, January 1995. <14, 31>

[36] Chuang-Lin Huang and Yu-Ming Huang. Facial Expression Recognition Using Model-Based Feature Extraction and Action Parameters Classification. *Journal of Visual Communication and Image Representation*, 8(3):278–290, September 1997. <15>

[37] Kwok-Wai Wan, Kin-Man Lam, and Kit-Chong Ng. An accurate active shape model for facial feature extraction. *Pattern Recognition Letters*, 26(15):2409–2423, November 2005. <15>

[38] Michel Valstar, I. Patras, and M. Pantic. Facial Action Unit Detection using Probabilistic Actively Learned Support Vector Machines on Tracked Facial Point. In *IEEE Conference on Computer Vision and Pattern Recognition*, page 76, 2005. <15, 29, 30, 72, 89>

[39] Michel Valstar and Maja Pantic. Fully Automatic Facial Action Unit Detection and Temporal Analysis. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop*, pages 146–149, 2006. <15, 29, 30, 72, 89>

[40] M. Turk and A.P. Pentland. Face recognition using eigenfaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1991. <17, 30>

[41] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997. <17, 30>

[42] M.S. Bartlett, J.R. Movellan, and T.J. Sejnowski. Face Recognition by Independent Component Analysis. *IEEE Transactions on Neural Networks*, 13(6): 1450–1464, 2002. <17, 29, 30, 72, 89>

[43] M.J. Lyons, J. Budynek, and S. Akamatsu. Automatic Classification of Single Facial Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(12):1357–1362, 1999. <17, 20, 23>

[44] G. Donato, M. Bartlett, J. Hager, P. Ekman, and T. Sejnowski. Classifying facial actions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(10):974–989, 1999. <17>

[45] G. Ford. Tutorial on gabor filters. Technical report, Machine Perception Lab, Institute of Neural Computation, 2002. <17>

[46] J.R. Movellan. *Tutorial on Gabor Filters*, 2008. <17>

[47] Ying-Li Tian, T. Kanade, and J. Cohn. Eye-state action unit detection by Gabor wavelets. In *Proceedings of International Conference on Multi-modal Interfaces*, pages 143–150, 2000. <17>

[48] Y. Tian, T. Kanade, and J. Cohn. Evaluation of gabor-wavelet-based facial action unit recognition in image sequences of increasing complexity. In *IEEE International Conference on Automatic Face and Gesture Recognition*, page 229, 2002. <17>

[49] T. Ojala, M. Pietikainen, and D. Harwood. A comparative study of texture measures with classification based on featured distribution. *Pattern Recognition*, 29(1):51–59, 1996. <17, 31, 75, 81, 101, 118>

[50] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002. <18, 31, 59, 64, 75, 91, 102>

[51] T. Ahonen, A. Hadid, and M. Pietikainen. Face recognition with local binary patterns. In *European Conference on Computer Vision*, 2004. <18, 24>

[52] X. Feng, A. Hadid, and M. Pietikainen. A coarse-to-fine classification scheme for facial expression recognition. In *International Conference on Image Analysis and Recognition*, 2004. <18>

[53] Caifeng Shan, Shaogang Gong, and Peter W. McOwan. Facial expression recognition based on Local Binary Patterns: A comprehensive study. *Image and Vision Computing*, 27(6):803–816, May 2009. <18, 21, 23, 26, 29, 31, 34, 37, 40, 89, 102, 106>

[54] Y. Wang, H. Ai, B. Wu, and C. Huang. Real time facial expression recognition with AdaBoost. In *IEEE International Conference on Pattern Recognition*, pages 926–929, 2004. <18, 32>

[55] S.U. Jung and D.H. Kim and K.H. An and M.J. Chung. Efficient rectangle feature extraction for real-time facial expression recognition based on AdaBoost.

In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1941–1946, 2005. <18>

[56] Y. Tian. Evaluation of face resolution for expression analysis. In *CVPR Workshop on Face Processing in Video*, 2004. <19>

[57] I. Cohen, N. Sebe, A. Garg, L. Chen, and T.S. Huang. Facial expression recognition from video sequences: temporal and static modeling. *Computer Vision and Image Understanding*, 91(1-2):160–187, 2003. <19>

[58] Marian Stewart Barlett, Gwen Littlewort, Mark Frank, Claudia Lainscsek, Ian Fasel, and Javier Movellan. Recognizing Facial Expression: Machine Learning and Application to Spontaneous Behavior. In *Computer Vision and Pattern Recognition*, 2005. <19, 21, 91, 103>

[59] M. Yeasin, B. Bullot, and R. Sharma. From facial expression to level of interests: a spatio-temporal approach. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2004. <19>

[60] P. Ekman, W.V. Friesen, M. O'Sullivan, A. Chan, I. Diacoyanni-Tarlatzis, K. Heider, R. Krauser, W.A. Lecompte, and T. Pitcairn an dP.E. Ricci-Bitti. Universals and cultural differences in facial expressions of emotion. *Journal of Personality and Social Psychology*, 53(4):712–719, 1972. <19>

[61] P. Ekman. Strong Evidence for Universals in Facial Expressions: A Reply to Russell's Mistaken Critique. *Psychological Bulletin*, 115(2):268–287, 1994. <19>

[62] P. Ekman. Facial Action Coding System. In *A Human Face*. Salt Lake City, USA, 2002. <19, 20>

[63] T. Kanade, J. Cohn, and Y. Tian. Comprehensive Database for Facial Expression Analysis. In *IEEE International Conference on Automatic Face & Gesture Recognition*, pages 46–53, 2000. <21, 23, 44, 121>

[64] M. Pantic, M.F. Valstar, R. Rademaker, and L. Maat. Web-based database for facial expression analysis. In *IEEE International Conference on Multimedia & Expo*, pages 317–321, 2005. <21, 23>

[65] I. Cohen, A. Garg, and T.S. Huang. Emotion Recognition from Facial Expressions using Multilevel HMM. In *Neural Information Processing Systems*, 2000. <21>

[66] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang. Local Gabor Binary Pattern Histogram Sequence (LGBPHS): A Novel Non-Statical Model for Face

Representation and Recognition. In *IEEE International Conference on Computer Vision*, 2005. <21, 32>

[67] C.S. Lee and A. Elgammal. Facial expression analysis using nonlinear decomposable generative models. In *IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, 2005. <22>

[68] P. Lucey, J.F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2010. <23, 67, 80, 105, 106>

[69] Jacob Whitehill, Gwen Littlewort, Ian Fasel, Marian Bartlett, and Javier Movellan. Towards Practical Smile Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(11):2106–2111, 2009. <23, 24, 29>

[70] G. Qi, X. Hua, Y. Rui, J. Tang, and H. Zhang. Two-Dimensional Active Learning for Image Classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008. <25>

[71] Jie Chen, Chu He, Guoying Zhao, Matti Pietikainen, Xilin Chen, and Wen Gao. WLD: A Robust Local Image Descriptor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1705–1720, September 2010. <26, 32, 37, 38, 54>

[72] Bingbing Ni, Dong Xu, and Shuicheng Yan. Histogram Contextualization. *IEEE Transactions on Image Processing*, 21(2):778–788, February 2012. <26, 41>

[73] Rob Fergus, Fei-Fei Li, Pietro Perona, and Andrew Zisserman. Learning Object Categories From Internet Image Searches. *Proceedings of The IEEE*, 98(8): 1453–1466, August 2010. <27, 28>

[74] Li-Jia Li and Li Fei-Fei. OPTIMOL: Automatic Online Picture Collection via Incremental Model Learning. *International Journal of Computer Vision*, 88(2): 147–168, 2010. <27>

[75] B. Collins, J. Deng, L. Kai, and L. Fei-Fei. Towards scalable dataset construction: An active learning approach. In *European Conference on Computer Vision*, 2008. <28>

[76] Tamara L. Berg and David A. Forsyth. Animals on the Web. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2006. <28>

[77] F. Schroff, A. Criminsi, and A. Zisserman. Harvesting Image Databases from the Web. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33 (4):754–766, April 2011. <28>

[78] Bryan C. Russell, Antonio Torralba, Kevin P. Murphy, and William T. Freeman. LabelMe: A Database and Web-Based Tool for Image Annotation. *International Journal of Computer Vision*, 77(1-3):157–173, 2008. <28>

[79] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. <28>

[80] Abhinav Dhall, Roland Goecke, Simon Lucey, and Tom Gedeon. Static Facial Expression Analysis In Tough Conditions: Data, Evaluation Protocol And Benchmark. In *Accepted for publication at IEEE ICCV 2011 workshop BEFIT*, 2011. <29>

[81] Zhen Cui, Shiguang Shan, Xilin Chen, and Lei Zhang. Sparsely Encoded Local Descriptor for Face Recognition. In *International Conference on Automatic Face & Gesture Recognition*, 2011. <29, 31, 72, 89>

[82] Z. Cao, Q. Yin, J. Sun, and X. Tang. Face recognition with learning based descriptor. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2010. <29, 31, 72, 89>

[83] G. Hua and A. Akbarzadeh. A robust elastic and partial matching metric for face recognition. In *IEEE International Conference on Computer Vision*, 2009. <29, 31, 72, 89>

[84] Jian Yang, David Zhang, Alejandro F. Frangi, and Jing yu Yang. Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(1):131–137, January 2004. <30>

[85] T.F. Cootes, G.J. Edwards, and C.J. Taylor. Active Appearance Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):681–685, June 2001. <30, 90, 104>

[86] Bouchra Abboud, Franck Davoine, and Mo Dang. Facial expression recognition and synthesis based on an appearance model. *Signal Processing: Image Communication*, 19(8):723–740, September 2004. <31>

[87] Jaewon Sung, Takeo Kanade, and Daijin Kim. A Unified Gradient-Based Approach for Combining ASM into AAM. *International Journal of Computer Vision*, 75(2):297–309, 2007. <31>

[88] Bhmed Bilal Ashraf, Simon Lucey, Jeffrey F. Cohn, Tsuhan Chen, Zara Ambadar, and Kenneth M. Prkachin an dPatricia E. Solomon. The painful face - Pain expression recognition using active appearance models. *Image and Vision Computing*, 27(12):1788–1796, November 2009. <31>

[89] B. Zhang, S. Shan, X. Chen, and W. Gao. Histogram of Gabor phase patterns (HGPP): a novel object representation approach for face recognition. *IEEE Transactions on Image Processing*, 16(1):57–68, 2007. <31>

[90] T. Ahonen, A. Hadid, and M. Pietikainen. Face Description with Local Binary Patterns: Application to Face Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):2037–2041, 2006. <31, 44>

[91] S. Xie, S. Shan, X. Chen, and J. Chen. Fusing Local Patterns of Gabor Magnitude and Phase for Face Recognition. *IEEE Transaction on Image Processing*, 19(5):1349–1361, May 2010. <32, 60>

[92] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. <32, 77, 91, 102, 115>

[93] A. Albiol, D. Monzo, A. Martin, and J. Sastre. Face recognition using HOG-EBGM. *Pattern Recognition Letters*, 29:1537–1543, 2008. <32>

[94] Paul Ekman. Universal Facial Expressions of Emotion. *California Mental Health Research Digest*, 8(4), Autumn 1970. <32, 42>

[95] C. Strapparava and A. Valitutti. Wordnet-affect: an affective extension of wordnet. In *International Conference on Language Resources and Evaluation*, 2004. <33>

[96] D.S. Bolme, B.A. Draper, and J.R. Beveridge. Average of Synthetic Exact Filters. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. <34, 81, 106, 118>

[97] Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley Blackwell, 1998. <35>

[98] Ajay J. Joshi, Fatih Porikli, and Nikolaos Papanikolopoulos. Multi-Class Active Learning for Image Classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009. <36>

[99] Hsuan-Tien Lin, Chih-Jen Lin, and Ruby C. Weng. A note on Platt's probabilistic outputs for support vector machines. *Machine Learning*, 68(3):267–276, 2007. <36>

[100] Guoying Zhao and M. Pietikainen. Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):915–928, June 2007. <37, 40, 72, 118, 119, 127>

[101] A.K. Jain. *Fundamental of Digital Signal Processing*. Prentice Hall, 1989. <37>

[102] Shichuan Du and Aleix Martinez. The resolution of facial expressions of emotion. *Journal of Vision*, 11(13):1–13, November 2011. <47>

[103] Lijun Yin, Xiaozhou Wei, Yi Sun, Jun Wang, and Matthew J. Rosato. A 3D facial expression database for facial behavior research. In *International Conference on Automatic Face & Gesture Recognition*, 2006. <51, 72, 80>

[104] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, October 2005. <58>

[105] Z. Li, G. Liu, Y. Yang, and J. You. Scale and rotation-invariant local binary pattern using scale-adaptive texton and subuniform-based circular shift. *IEEE Transactions on Image Processing*, 21(4):2130–2140, October 2012. <58>

[106] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002. <59>

[107] L. Nanni, A. Lumini, and S. Brahnam. Local binary patterns variants as texture descriptors for medical image analysis. *Artificial Intelligence in Medicine*, 49 (2):117–125, June 2010. <59>

[108] S. Liao, M. Law, and A. Chuang. Dominant local binary patterns for texture classification. *IEEE Transactions on Image Processing*, 18(5):1107–1118, May 2009. <59, 60>

[109] X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under diffcult lighting conditions. *IEEE Transactions on Image Processing*, 19 (6):1635–1650, June 2010. <59>

[110] A. Hafiane, G. Seetharaman, K. Palaniappan, and B. Zavidovique. Rotationally invariant hashing of median binary patterns for texture classification. *Image Analysis and Recognition, Lecture Notes in Computer Science*, 5112:619–629, 2008. <60>

[111] Y. Guo, G. Zhao, and M. Pietikainen. Texture classification using a linear configuration model based descriptor. In *British Machine Vision Conference*, 2011. <60, 64, 65>

[112] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, April 2002. <60>

[113] B. Caputo, E. Hayman, and P. Mallikarjuna. Class-specific material categorisation. In *European Conference on Computer Vision*, 2005. <64>

[114] K.J. Dana, B. van Ginneken, S.K. Nayar, and J.J. Koenderink. Reflectance and texture of real world surfaces. *ACM Transactions on Graphics*, 18(1):1–34, January 1999. <65>

[115] Gary B. Huang, Marwan Mattar, Honglak Lee, and Erik Learned-Miller. Learning to Align from Scratch. In *Advances in Neural Information Processing Systems (NIPS)*, 2012. <69>

[116] W. Zheng, H. Tang, Z. Lin, and T.S. Huang. A novel approach to expression recognition from non-frontal face images. In *IEEE International Conference on Computer Vision*, 2009. <69, 72>

[117] M. F. Valstar and M. Pantic. Combined support vector machines and hidden Markov models for modeling facial action temporal dynamics. In *IEEE international conference on Human Computer Interaction*, 2007. <72>

[118] Yuxiao Hu, Zhihong Zeng, Lijun Yin, Xiaozhou Wei, Xi Zhou, and Thomas Huang. Multi-View Facial Expression Recognition. In *IEEE International Conference on Automatic Face & Gesture Recognition*, pages 1–6, 2006. <72, 86>

[119] S. Moore and R. Bowden. Local binary patterns for multi-view facial expression recognition. *Computer Vision and Image Understanding*, 115(4):541–558, April 2011. <72, 73>

[120] Ognjen Rudovic, Maja Panic, and Ioannis Patras. Coupled Gaussian Processes for Pose-Invariant Facial Expression Recognition. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 35(6):1357–1369, June 2013. <72, 73>

[121] Kaimin Yu, Zhiyong Wang, Li Zhuo, and Dagan Feng. Harvesting Web Images for Realistic Facial Expression Recognition. In *International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, 2010. <81, 105>

[122] Kaimin Yu, Zhiyong Wang, Li Zhuo, Jianjun Wang, Zheru Chi, and Dagan Feng. Learning realistic facial expression from web images. *Pattern Recognition*, 46(8):2144–2155, 2013. <81, 91, 101, 105>

[123] Tian Xia, Dacheng Tao, Tao Mei, and Yongdong Zhang. Multiview Spectral Embedding. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 40 (6):1438–1446, December 2010. <89, 92, 96, 109>

[124] Jun Yu, Dacheng Tao, Yong Rui, and Jun Cheng. Pairwise constraints based multiview features fusion for scene classification. *Pattern Recognition*, 46(2): 483–496, Feburary 2013. <89, 92>

[125] Yi Yang, Jingkuan Song, Zi Huang, Zhigang Ma, Nicu Sebe, and Alexander G. Hauptmann. Multi-Feature Fusion via Hierarchical Regression for Multimedia Analysis. *IEEE Transactions on Multimedia*, (In-Press), 2012. <91>

[126] Peter Gehler and Sebastian Nowozin. On Feature Combination of Multiclass Object Classification. In *International Conference on Computer Vision*, 2009. <91, 92>

[127] L. Cao, J. Luo, F. Liang, and T. S. Huang. Heterogeneous feature machines for visual recognition. In *International Conference on Computer Vision*, 2009. <91>

[128] J. Yang, Y. Li, Y. Tian, L. Duan, and W. Gao. Group-sensitive multiple kernel learning for object categorization. In *International Conference on Computer Vision*, 2009. <91, 92>

[129] Yi-Ren Yeh, Ting-Chu Lin, Yung-Yu Chuang, and Yu-Chiang Frank Wang. A Novel Multiple Kernel Learning Framework for Heterogeneous Feature Fusion and Variable Selection. *IEEE Transaction on Multimedia*, 14(3):563–574, June 2012. <91, 92>

[130] Lefei Zhang, Liangpei Zhang, Dacheng Tao, and Xin Huang. On Combining Multiple Features for Hyperspectral Remote Sensing Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 50(3):879–893, March 2012. <92>

[131] Tianhao Zhang, Dacheng Tao, Xuelong Li, and Jie Yang. Patch alignment for dimension reduction. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1299–1313, September 2009. <93>

[132] Matthew Brand. Charting a manifold. In *Advances in Neural Information Processing Systems*, 2002. <108>

[133] T. Pfister, X. Li, G. Zhao, and M. Pietikainen. Recognising Spontaneous Facial Micro-expressions. In *IEEE International Conference on Computer Vision*, 2011. <114>

[134] D. W. Cunningham and C. Wallraven. Dynamic information for the recognition of conversational expressions. *Journal of Vision*, 9(13):1–17, 2009. <114>

[135] K. Schindler and L. van Gool. Action snippets: How many frames does human action recognition require? In *IEEE Conference on Computer Vision and Pattern Recognition*, 2008. <114>

[136] Genliang Guan, Zhiyong Wang, Shiyang Lu, J.D. Deng, and Dagan Feng. Keypoint-Based Keyframe Selection. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(4):729–734, April 2013. <115>

[137] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24:381–395, 1981. <116>