Emotion Recognition from Eye Region Signals using Local Binary Patterns

by

Gaurav Jain

A thesis submitted in conformity with the requirements for the degree of Master of Applied Science
Graduate Department of Electrical and Computer Engineering
University of Toronto

Copyright © 2011 by Gaurav Jain
Abstract

Emotion Recognition from Eye Region Signals using Local Binary Patterns

Gaurav Jain

Master of Applied Science

Graduate Department of Electrical and Computer Engineering

University of Toronto

2011

Automated facial expression analysis for Emotion Recognition (ER) is an active research area towards creating socially intelligent systems. The eye region, often considered integral for ER by psychologists and neuroscientists, has received very little attention in engineering and computer sciences. Using eye region as an input signal presents several benefits for low-cost, non-intrusive ER applications.

This work proposes two frameworks towards ER from eye region images. The first framework uses Local Binary Patterns (LBP) as the feature extractor on grayscale eye region images. The results validate the eye region as a significant contributor towards communicating the emotion in the face by achieving high person-dependent accuracy. The system is also able to generalize well across different environment conditions.

In the second proposed framework, a color-based approach to ER from the eye region is explored using Local Color Vector Binary Patterns (LCVBP). LCVBP extend the traditional LBP by incorporating color information extracting a rich and a highly discriminative feature set, thereby providing promising results.
Acknowledgements

I am greatly indebted to Prof. Konstantinos (Kostas) N. Plataniotis for his valuable support, and the informed guidance and advice he has extended to me throughout my work towards this thesis. Prof. Kostas has truly been a great advisor, and even a greater mentor.

I would like to acknowledge the financial support that I have had: MITACS and Xiris Automation Inc. for the MITACS-Accelerate internship opportunity, and the Department of Electrical and Computer Engineering for the research scholarship. I would like to thank my proposal and defense committee members for their time and consideration in providing constructive feedback.

A special thanks to my Multimedia Lab colleagues for helpful discussions, Foteini Agrafioti and Jeff Fortuna for their valuable answers to my questions, and Diane Kostka for her feedback, words of encouragement, and her support in a number of ways.

This thesis is dedicated to my parents, Kailash and Sunita, and my sister, Payal. Thank you for always supporting me in my endeavors, and providing me with constant encouragement throughout the period of my studies.
# Contents

Glossary ............................................. xv

1 Introduction ..................................... 1

1.1 Social Signal Processing ......................... 1

1.2 The Role of Facial and Eye Expressions .......... 3

1.3 Research Goals and Challenges ................... 6

1.4 Thesis Contributions ............................. 7

1.4.1 Grayscale LBP feature-based system ............ 7

1.4.2 LCVBP feature-based system ................... 9

1.5 Thesis Overview .................................. 10

2 Background and Literature Review ................. 12

2.1 Problem Space for Facial Expression Analysis .... 12

2.1.1 Emotions: Taxonomy ............................ 12

2.1.2 Facial Expression Databases .................... 13

2.2 Automated Facial Expression Recognition .......... 14

2.2.1 Geometric-Feature-Based ....................... 18

2.2.2 Appearance-Feature-Based ...................... 20

2.3 Summary .......................................... 24
# Image Representation

3.1 Grayscale Local Binary Patterns (LBP) ........................................... 27
3.2 Local Color Vector Binary Patterns (LCVBP) ................................. 31
   3.2.1 Color Norm Patterns ......................................................... 32
   3.2.2 Color Angular Patterns ....................................................... 34
3.3 Summary ....................................................................................... 36

# Grayscale LBP feature-based System

4.1 System Architecture ................................................................. 38
   4.1.1 Pre-Processing ................................................................. 39
   4.1.2 Feature Extraction ......................................................... 46
   4.1.3 Training and Classification ................................................ 48
4.2 Experimental Setup .................................................................... 51
   4.2.1 Dataset Description ........................................................ 51
   4.2.2 Performance Measures .................................................... 53
   4.2.3 Experiments .................................................................. 55
4.3 Results and Evaluation ............................................................... 57
   4.3.1 Module Performance: Automatic Eye Detection .................. 57
   4.3.2 Experiment 1: Person-dependent recognition on CK+ dataset 58
   4.3.3 Experiment 2: Person-independent recognition on CK+ dataset 60
   4.3.4 Experiment 3: Cross-database recognition on CK+ and MMI datasets .......................................................... 63
4.4 Discussion .................................................................................. 64
4.5 Summary .................................................................................... 69

# Color LBP (LCVBP) feature-based System

5.1 System Architecture ................................................................. 72
5.1.1 Pre-Processing ........................................ 73
5.1.2 Feature Extraction .................................... 75
5.1.3 Feature Fusion ........................................ 76

5.2 Experimental Setup ..................................... 79
5.2.1 Dataset Description ................................... 79
5.2.2 Objectives and Framework ............................ 81

5.3 Results and Evaluation .................................. 82

5.4 Summary .................................................. 84

6 Conclusions and Future Directions ..................... 87
6.1 Contributions ............................................. 87
6.1.1 Grayscale LBP feature-based system ................. 88
6.1.2 Color LBP (LCVBP) feature-based system .......... 90

6.2 Future Directions ......................................... 91
6.2.1 Boosting the Recognition and Accuracy .............. 91
6.2.2 Including Temporal Information from Video ........ 92
6.2.3 Beyond the basic emotions ............................ 93
6.2.4 Application: Implicit Tagging of Multimedia Content 93

A Review of Eye Detection ................................. 96
A.1 Shape Models ............................................ 97
A.2 Pupil and Iris Detection ................................. 98
A.3 Wavelet Features with Boosting ........................ 99

B Skin-tone Detection ....................................... 102
B.1 Approaches ............................................. 103
B.1.1 YCbCr Model .......................................... 104
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Facial Expression Corpora</td>
<td>14</td>
</tr>
<tr>
<td>2.2</td>
<td>Facial Expression Synthesis Rules from AUs</td>
<td>17</td>
</tr>
<tr>
<td>4.1</td>
<td>Automatic Eye Detection Module Performance</td>
<td>57</td>
</tr>
<tr>
<td>4.2</td>
<td>Results for the person-dependent 6-class recognition on the CK+ dataset</td>
<td>58</td>
</tr>
<tr>
<td></td>
<td>using $\chi^2$-NN classification</td>
<td></td>
</tr>
<tr>
<td>4.3</td>
<td>Confusion matrix of person-dependent 6-class recognition on the CK+ dataset.</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>Methodology: Manual Eye Detection + $LBP_{8,2} + \chi^2$-NN</td>
<td></td>
</tr>
<tr>
<td>4.4</td>
<td>Results for the person-dependent 6-class recognition on the CK+ dataset</td>
<td>59</td>
</tr>
<tr>
<td></td>
<td>using SVM classification</td>
<td></td>
</tr>
<tr>
<td>4.5</td>
<td>Confusion matrix of person-dependent 6-class recognition on the CK+ dataset.</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td>Methodology: Manual Eye Detection + $LBP_{8,2} + L$-SVM</td>
<td></td>
</tr>
<tr>
<td>4.6</td>
<td>Results for the person-independent 6-class recognition trained and tested</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>on the CK+ dataset using $\chi^2$-NN classification</td>
<td></td>
</tr>
<tr>
<td>4.7</td>
<td>Confusion matrix of person-independent 6-class recognition on the CK+ dataset.</td>
<td>61</td>
</tr>
<tr>
<td></td>
<td>Methodology: Manual Eye Detection + $LBP_{16,2} + \chi^2$-NN</td>
<td></td>
</tr>
<tr>
<td>4.8</td>
<td>Results for the person-independent 6-class recognition on the CK+ dataset</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td>using SVM classification</td>
<td></td>
</tr>
</tbody>
</table>
4.9 Confusion matrix of person-independent 6-class recognition on the CK+ dataset. Methodology: Manual Eye Detection + \( LBP_{5,1} \) + L-SVM. 

4.10 Results for the 4-class cross-database recognition trained and tested on CK+ and MMI datasets respectively using \( \chi^2 \)-NN classification. 

4.11 Results for the 4-class cross-database recognition trained and tested on CK+ and MMI datasets respectively using L-SVM classification. 

4.12 Comparisons between different methods for 6-class person-independent recognition using 10-fold cross-validation. 

4.13 Generalization performance for cross-database recognition for different approaches. 

4.14 Time and Memory costs for 8 and 16 neighborhood samples in LBP feature extraction on a \( 110 \times 60 \) eye region image, and comparison with the results listed in [90]. 

5.1 Comparison of the results for the 4-class recognition on the MMI dataset using LCVBP features and Grayscale LBP features. 

B.1 True positive detection rates (%) for each skin model under various illumination scenarios. 

B.2 False positive detection rates (%) for each skin model under various illumination scenarios.
# List of Figures

<table>
<thead>
<tr>
<th>Figure Number</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Social Signals and Behavioral Cues</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>Prototypical Expressions of six basic emotions</td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>Eye expressions of six basic emotions</td>
<td>4</td>
</tr>
<tr>
<td>2.1</td>
<td>The Circumplex model of Russell</td>
<td>13</td>
</tr>
<tr>
<td>2.2</td>
<td>The upper action units in FACS</td>
<td>16</td>
</tr>
<tr>
<td>2.3</td>
<td>Facial Fiducial Points as described in [9, 100]</td>
<td>18</td>
</tr>
<tr>
<td>2.4</td>
<td>The Motion-Units (MUs) introduced in [13]</td>
<td>19</td>
</tr>
<tr>
<td>3.1</td>
<td>The basic LBP operator</td>
<td>27</td>
</tr>
<tr>
<td>3.2</td>
<td>Example of LBP texture primitives</td>
<td>28</td>
</tr>
<tr>
<td>3.3</td>
<td>LBP Circular Neighborhood Examples</td>
<td>28</td>
</tr>
<tr>
<td>3.4</td>
<td>LBP histogram generation from the eye region image</td>
<td>31</td>
</tr>
<tr>
<td>3.5</td>
<td>Construction of a Color Vector</td>
<td>32</td>
</tr>
<tr>
<td>3.6</td>
<td>Visualization of color norm patterns extraction</td>
<td>33</td>
</tr>
<tr>
<td>3.7</td>
<td>Visualization of the color angle between R and G spectral bands</td>
<td>35</td>
</tr>
<tr>
<td>4.1</td>
<td>Overview of the system architecture for the grayscale LBP-based system</td>
<td>40</td>
</tr>
<tr>
<td>4.2</td>
<td>Face and eye detection flowchart</td>
<td>43</td>
</tr>
<tr>
<td>4.3</td>
<td>Results from the MPT Eye detection system</td>
<td>44</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>4.4</td>
<td>Eye Region Extraction Parameters</td>
<td></td>
</tr>
<tr>
<td>4.5</td>
<td>Example of LBP operator to the eye region image.</td>
<td></td>
</tr>
<tr>
<td>4.6</td>
<td>LBP feature extraction on the eye region image</td>
<td></td>
</tr>
<tr>
<td>4.7</td>
<td>$\chi^2$ dissimilarity weights.</td>
<td></td>
</tr>
<tr>
<td>4.8</td>
<td>Sample face expression images from the CK+ and MMI datasets</td>
<td></td>
</tr>
<tr>
<td>5.1</td>
<td>Overview of the system architecture for the LCVBP feature-based system.</td>
<td></td>
</tr>
<tr>
<td>5.2</td>
<td>LCVBP operator applied to a region of an eye image with $P = 8, R = 2$</td>
<td></td>
</tr>
<tr>
<td>5.3</td>
<td>Sample facial expression color images from the MMI dataset.</td>
<td></td>
</tr>
<tr>
<td>5.4</td>
<td>Comparison of recognition accuracy obtained for color norm and color angular pattern features separately with the fused LCVBP features.</td>
<td></td>
</tr>
<tr>
<td>A.1</td>
<td>Haar-like feature prototypes</td>
<td></td>
</tr>
<tr>
<td>A.2</td>
<td>Haar-like and mirror image wavelets used by the MPT face and eye detection system</td>
<td></td>
</tr>
<tr>
<td>B.1</td>
<td>An example of grayworld pre-processing</td>
<td></td>
</tr>
<tr>
<td>B.2</td>
<td>Examples of video frames from the test dataset</td>
<td></td>
</tr>
<tr>
<td>B.3</td>
<td>True Positive Detection Rate Comparison</td>
<td></td>
</tr>
<tr>
<td>B.4</td>
<td>An example of non-skin pixels falsely classified as skin</td>
<td></td>
</tr>
<tr>
<td>B.5</td>
<td>System overview of skin-tone based Face Detection</td>
<td></td>
</tr>
<tr>
<td>B.6</td>
<td>Appearance-based versus Skin-tone Face Detection</td>
<td></td>
</tr>
</tbody>
</table>
Glossary

List of Abbreviations

AAM  Active Appearance Model
AU   Action Unit
CAP  Color Angular Patterns
CK+  The Extended Cohn-Kanade Database
CNP  Color Norm Patterns
EM   Expectation Maximization
FACS Facial Action Coding System
FER  Facial Expression Recognition
FR   Face Recognition
HCI  Human Computer Interaction
ICA  Independent Component Analysis
LBP  Local Binary Patterns
LCVBP  Local Color Vector Binary Patterns
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDA</td>
<td>Linear Discriminant Analysis</td>
</tr>
<tr>
<td>LDP</td>
<td>Local Directional Patterns</td>
</tr>
<tr>
<td>LFA</td>
<td>Local Feature Analysis</td>
</tr>
<tr>
<td>MMI</td>
<td>MMI Facial Expression Database</td>
</tr>
<tr>
<td>MPT</td>
<td>Machine Perception Toolbox</td>
</tr>
<tr>
<td>NN</td>
<td>Nearest Neighbor</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal Component Analysis</td>
</tr>
<tr>
<td>RA</td>
<td>Recognition Accuracy</td>
</tr>
<tr>
<td>RANSAC</td>
<td>Random Sample Consensus</td>
</tr>
<tr>
<td>RBF</td>
<td>Radial Basis Function</td>
</tr>
<tr>
<td>RGB</td>
<td>Red-Green-Blue</td>
</tr>
<tr>
<td>SSP</td>
<td>Social Signal Processing</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machine</td>
</tr>
</tbody>
</table>

**List of Symbols**

\[ \delta(x) \] Thresholding function

\[ \gamma \] Small valued constant

\[ \theta^{(i,j)} \] Color angle between \( i \)th and \( j \)th spectral band

\[ \theta_{c(p)} \] Color angle value at a neighboring pixel
\( \theta_c \)  Color angle value at pixel \( z_c \)

c  Color vector

\( c_k \)  Color pixel value at \( z_c \) of the associated \( I_k \) band

\( g_c \)  Gray value of the center pixel

\( g_p \)  Gray value of the neighboring pixel

H  Global LBP histogram

\( h^m \)  Regional LBP histogram

\( H_{ca} \)  Global CAP histogram

\( h^m_{ca} \)  Regional CAP histogram

\( H_{cn} \)  Global CNP histogram

\( h^m_{cn} \)  Regional CNP histogram

I  Input Image

\( I_k \)  \( k \)th spectral band of the input image

\( L(z_c) \)  LBP labeled image

\( LBP_{u2} \)  Uniform LBP

\( LBP_{ca} \)  CAP

\( LBP_{cn} \)  CNP

\( M \)  Number of non-overlapping regions
\( P \)  
Number of equally spaced pixels in the LBP circular neighborhood

\( R \)  
Radius of the LBP circular neighborhood

\( r_{(i,j)} \)  
Ratio of pixel values between \( i \)th and \( j \)th spectral band

\( r_{c(p)} \)  
Color norm value at a neighboring pixel

\( r_c \)  
Color norm value at pixel \( z_c \)

\( U \)  
Number of uniform patterns

\( z_c \)  
Center pixel
Chapter 1

Introduction

Machines can help, but peering into human emotion is a more complicated task than traditional search and analytics. Marketers today aren’t mining simply for information on click-throughs and page views—they want to mine the secrets of the human heart and come up with hard data on soft concepts such as “mood” and “passion”.


How can we make the computer systems interpret human behavior in a socially-intelligent manner? Which applications are we overlooking because of a limiting view of their human-behavior understanding abilities?

1.1 Social Signal Processing

Social signals are the character of one’s attitude in a social interaction, brought together through a combination of non-verbal behavioural cues such as facial expressions, body postures, hand gestures, and vocal outbursts (Figure 1.1). The ability to
process these behavioral cues of a person to identify social signals and social behaviors is an underlying phenomenon of creating socially intelligent computer systems. Social Signal Processing (SSP) is an emerging computational framework that attempts to identify these social signals in an automated way, providing the basis for creating naturalistic, socially-aware computing interfaces, built to interact with humans, on models of human behavior [103]. In spite of recent advances in machine analysis and synthesis of relevant behavioural cues like gaze exchange, blinks, facial expressions, smiles, head nods, laughter, and similar, the research efforts in interpreting these social signals with regards to useful real-world applications are still tentative and pioneering efforts.

Figure 1.1: Social Signals and Behavioral Cues.

The field of human affective (emotional) behaviour analysis, a necessary input into the SSP framework, has gained tremendous attention and progress over the past decade [85, 112]. Here, machines can provide information about the emotional state of a human in a non-intrusive way and in real settings (“as it occurs”). Affective behaviour analysis has found its usefulness in Human-Computer Interaction (HCI) for several application areas [15] such as lie detection in psychological experiments, generation of virtual avatars for gaming and entertainment, personal assistants in a clinical or an office setting, tutoring systems, alertness systems where doctors can be
1.2 The Role of Facial and Eye Expressions

People use an array of nonverbal cues such as facial expressions, vocal nuances, hand and body gestures, and body posture to communicate their emotions. The human face, housing the majority of our sensory apparatus (eyes, ears, mouth, and nose) holds a core position for the processing of human communicative signals in the field of affective computing. It acts as our most natural and direct means of communicating and understanding other humans’ affective state and intentions on the basis of shown facial expression. Indeed, this is supported by the experiments conducted in [79] which show in everyday interaction, 7% of the communication happens through language, 38% via paralanguage whereas facial expressions contribute to the 55% of the communication.

![Figure 1.2: Prototypical Expressions of six basic emotions: Anger, Disgust, Fear, Joy, Sadness, and Surprise (from left to right).](image)

Automatic facial expression analysis especially for emotion recognition has become an active research area over the last decade [31, 112]. The advancements in automatic analysis of facial expressions can be broadly divided into two main streams: facial affect (emotion) detection pertaining to a set of basic or prototypical emotions (e.g.,
anger, disgust, fear, joy, sadness, and surprise [26] as seen in Figure 1.2) and facial
muscle action (action unit) detection defined under the Facial Action Coding System
(FACS) [25] framework, a widely used model for coding muscle actions in behav-
ioral sciences. Under the domain of facial expression analysis, the properties of eye
region signals exclusively have received very little attention for emotion recognition
in engineering and computer sciences, and have mainly been restricted to eye and
gaze tracking applications [22]. For example, studying brain movements regarding
perception of visual surroundings, driver fatigue detection systems, and gauging at-
tention levels towards certain data content. Research in psychology points towards
the relevance of attending to the properties of eyes and the surrounding region for
emotion recognition [37, 52]. Furthermore, FACS defines many action units (AUs)
for eyes (Figure 2.2). Under the FACS framework, all six basic emotions described
by Ekman (anger, disgust, fear, joy, sadness, and surprise) involve a specific change
in one element of the eye region (brow and furrow movement, pupil size variation,
and other eye region muscle variations).

Figure 1.3: Psychological studies suggest that eye expressions hold an important
position in emotion recognition. In raster-scan order: Anger, Disgust, Fear, Joy,
Sadness, and Surprise.

Also, a posed ‘happy’ expression or a fake smile can be discerned from a spon-
1.2 The Role of Facial and Eye Expressions

taneous smile which usually involves the expressive cues in the eye region, such as the wrinkles around the eye corners or the squinting of the eye opening by the ocular muscles (the so-called “Duchenne smile”) [21]. Moreover, studies on eye region analysis in patients with autism or Asperger syndrome reveal that, in addition to the basic emotions, complex feelings such as jealousy, envy or guilt can also be recognized using the information obtained from analyzing the eye region [4]. Thus, eyes, which are often referred to as “a window to the soul” can be considered integral to the inference of emotion.

From an engineering standpoint, using the eye region as a (biometric) signal for emotion inference (as opposed to other signals such as EEG, galvanic skin response, etc.) satisfies the requirements for a low-cost, non-intrusive, less-demanding type of solution that is suitable for building a real-world application such as implicit multimedia content tagging. In addition, when compared to the entire face, the eye region presents several benefits as described below.

- The complexity and the processing time are reduced as the eye region is roughly one-thirds the size of the face image.
- In real-world applications such as multimedia browsing and video conferencing, excluding the mouth region from the face analysis may result in less ambiguity and hence reduced errors or falsely recognized emotion states.
- The posed expressions or a fake smile can be rejected due to the absence of ocular cues.
- Additionally, a case can be made for eye region based emotion recognition being especially useful in privacy-sensitive applications where storing the entire face

5
1.3 Research Goals and Challenges

Given the role of facial expressions and the importance of the eye region in emotion recognition, the motivation for this thesis stems from the following research question:

Is it possible to automatically classify the emotional states of the user into one of the six basic emotions (anger, disgust, fear, joy, sadness, and surprise) by solely analyzing the information contained in the eye region from a social signal processing point-of-view?

If the answer to the above question is yes, then using the eye region (instead of the entire face) as the input signal to an emotion recognition system has several advantages.

1.3 Research Goals and Challenges

This thesis explores the information conveyed by the eye region in various facial expressions pertaining to basic emotional states such that this information can be fed as an input to a higher-level socially intelligent system. More specifically, the primary goal of this research is to design, implement, and evaluate a vision-based system which, using statistical learning techniques, classifies an eye region image into one of the six basic emotional states (anger, disgust, fear, joy, sadness, and surprise) under the context of building socially intelligent systems. In light of this goal, the thesis project includes some explicit and implicit requirements as listed below.

- The system should act non-intrusively. In other words, the system should be able to capture and analyze images from a distance without the subject having to be in contact. This is required from an everyday use point-of-view where the user interacts with a webcam-enabled computer or images are captured by a camera at a distance in a public setting.
1.4 Thesis Contributions

- The system should be able to perform indoors under varying illumination conditions. This can include incandescent and fluorescent lighting, shadows, frontal, directed, and profile lighting. This lighting variation can have an impact both on eye region detection and expression recognition.

- The system will be constrained to operate for frontal or nearly-frontal views allowing head-pose variations of 15 degrees on either side from the frontal pose.

- The faces can be present at multiple scales with respect to the camera position; therefore scale invariance, training, and normalization will be required in order to prepare the input images for feature extraction, and classification.

1.4 Thesis Contributions

This thesis makes two major contributions towards automatic inference of emotional states from eye region signals\(^1\):

- Emotion recognition from eye region signals using grayscale local binary patterns (LBP) features,

- Emotion recognition from eye region signals using local color vector binary patterns (LCVBP) features.

1.4.1 Grayscale LBP feature-based system

The first contribution of this thesis is the proposed automatic emotion recognition system based on the traditional LBP features to classify the set of basic emotions.

\(^1\)The term \textit{eye region signals} used herein refers to the information contained (set of ordered pixels) in a 2D eye region image.
from grayscale eye region images. The LBP features are chosen due to their proven effectiveness in the past for FER as they provide robustness to variations in illumination, and are able to provide highly discriminative features due to different levels of locality. The system is evaluated with two different classifiers: the low-cost nearest neighbor with chi-square distance metric, and the more sophisticated Support Vector Machines (SVMs) with a linear kernel on the widely used CK+ and MMI facial expression datasets. The experimental evaluations both the classification tools are presented in order to reflect the performance versus cost (time, memory) trade-off when deploying such a system in real-world applications.

Furthermore, an automatic eye detection module, MPT EyeFinder [32], is also integrated in the pre-processing stage to build a complete automatic emotion recognition system. The system is able to achieve excellent person-dependent recognition accuracy (= 99.5%) on the CK+ dataset, making it suitable for applications that have the flexibility to adapt to and incorporate new samples into its trained model, thereby improving the performance. For person-independent classification tests on CK+ dataset using 10-fold cross-validation, our proposed system was able to achieve the recognition accuracy of 70.4% (compared to ≈93% achieved by state-of-the-art FER systems operating on full face images [92]), proving the usefulness of eye region as a significant contributor in expression recognition. In addition, a second set of experiments is performed in order to evaluate recognition accuracy on unseen data taken under completely different environmental conditions. This is done by training the system on images from CK+ dataset and testing on MMI dataset. The system is able to achieve cross-database generalization performance upto 67.4%; findings that are consistent with other FER systems [91] with regards to the generalization ability to other datasets. A detailed description of the LBP operator can be found in Chapter 3, and a complete discussion about the grayscale LBP feature-based system
1.4 Thesis Contributions

is provided in Chapter 4.

1.4.2 LCVBP feature-based system

The first proposed grayscale LBP-based system discards any color information that might be available and can be used to extract better discriminatory features from the input images. Therefore, the second major contribution of this thesis is the proposed design of a novel approach to emotion/expression recognition from color eye region images based on LCVBP features. LCVBP, a recently proposed effective feature descriptor for face recognition, extends the idea of grayscale LBP features to include color information extracting two sets of features from color images: (1) color norm patterns (CNP) related to the norm of the multiple channels, and (2) color angular patterns (CAP) related to the spatial interactions between different color channels of the image. The main objective of the proposed LCVBP based pipeline is to investigate a color based approach for eye expression recognition and in turn, evaluate the feasibility of a LCVBP as a suitable feature descriptor for the same. Therefore, the eye center positions are manually labelled as ground truth to bypass any errors introduced by the pre-processing stage.

The success of LCVBP as a suitable feature descriptor for emotion recognition on eye region images is demonstrated by performing 4-class recognition using SVM as the classification tool. The CNP and CAP features are concatenated together into a single column feature vector by performing fusion at the feature level in order to maximize a complementary effect on recognition. The results demonstrate the fact that fused or combined LCVBP features provide increased recognition performance as compared to applying CNP or CAP separately. This highlights the fact that the norm and angular texture features in a color image are able to provide different information
and are mutually compensational for improving expression recognition performance from eye region images. The system is able to achieve an accuracy of 86.3% on a selected subset of the MMI facial expression database, providing confidence in a color based approach for visual expression recognition from facial or eye region images. A more thorough description of the LCVBP operator is provided in Chapter 3 and an in-depth explanation and evaluation of the proposed LCVBP feature-based system is provided in Chapter 5.

1.5 Thesis Overview

The remainder of this thesis is as follows:

- **Chapter 2: Background and Literature Review**
  
  Chapter 2 provides the background on how to classify emotions, the relation between facial expressions and emotions, and surveys the state-of-the-art in facial expression recognition.

- **Chapter 3: Image Representation**
  
  The grayscale local binary patterns (LBP) and local color vector binary patterns (LCVBP) features adopted for image representation for the two proposed systems are introduced in this chapter.

- **Chapter 4: Grayscale LBP feature-based system**
  
  Chapter 4 describes the emotion recognition system based on grayscale LBP features. The chapter begins by providing an overview of the system architecture, followed by an in-depth description of the different modules, and then concludes by providing the experiments and the evaluation results for this system.
• **Chapter 5: LCVBP feature-based system**

The second emotion recognition framework based on LCVBP features, along with the related experiments, results, and evaluation, is explained in this chapter.

• **Chapter 6: Conclusion and Future Directions**

This chapter highlights the work presented in this thesis, its major contributions, and the directions for future research. It concludes by describing a potential application for the proposed systems.
Chapter 2

Background and Literature Review

2.1 Problem Space for Facial Expression Analysis

2.1.1 Emotions: Taxonomy

Emotion theorists and psychologists have defined several models for emotion classification ranging from universally displayed basic emotions to culturally specific complex ones. Out of the various models in emotion research, there are two that have dominated facial expression research: Ekman’s basic set of emotions [26], and Russell’s circumplex model of affect [88].

Ekman and Freisen in 1971 [26] proposed six prototypical (basic) emotions - anger, disgust, fear, joy, sadness, and surprise - which are universally displayed and recognized from facial expressions. The universality of these basic emotions, having its roots in the universality thesis proposed by Charles Darwin, was further supported by the cross-cultural studies in [24]. This categorical description has gained popularity and possesses an advantage from the fact that facial expressions pertaining to basic emotions are easily recognized and described by humans. This model of emotion
subspace has become the most prevalent model for measuring emotion, and the facial expressions associated with these basic emotions have dominated the studies related to facial expression recognition over the last four decades.

An alternative description model of human emotions was proposed by Russell [88], where emotional states are represented as a circle in a two-dimensional bipolar space (pleasantness-unpleasantness, arousal-sleep) (as illustrated in Figure 2.1) rather than specific, discrete categories. For example, anger might be perceived as conveying extreme displeasure and moderately high arousal.

![Figure 2.1: The Circumplex model of Russell](image)

### 2.1.2 Facial Expression Databases

Researchers in the field of facial expression recognition usually perform their evaluation and report results using a number of popular facial expression databases as listed in Table 2.1. These databases are publicly available and often differ in terms of their stimuli and recording setup, actors, and the general pose of the actors. It is
2.2 Automated Facial Expression Recognition

important to note that most of these databases contain posed expressions, in which subjects are asked to act out certain emotions. For this thesis, the CK+ and MMI facial expression databases have been used for training and evaluation.

<table>
<thead>
<tr>
<th>Property</th>
<th>CK</th>
<th>CK+</th>
<th>MMI</th>
<th>JAFFE</th>
<th>POFA</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of subjects</td>
<td>100</td>
<td>123</td>
<td>43</td>
<td>10</td>
<td>14</td>
</tr>
<tr>
<td>No. of images</td>
<td>-</td>
<td>-</td>
<td>&gt;250</td>
<td>213</td>
<td>110</td>
</tr>
<tr>
<td>No. of videos</td>
<td>486</td>
<td>593</td>
<td>1280</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Gray/Color</td>
<td>Gray</td>
<td>Gray</td>
<td>Color</td>
<td>Gray</td>
<td>Gray</td>
</tr>
<tr>
<td>Resolution</td>
<td>640x490</td>
<td>640x480</td>
<td>720x576</td>
<td>256x256</td>
<td>-</td>
</tr>
<tr>
<td>Pose</td>
<td>Frontal</td>
<td>Frontal</td>
<td>Frontal/Dual</td>
<td>Frontal</td>
<td>Frontal</td>
</tr>
<tr>
<td>FACS-coded</td>
<td>Yes</td>
<td>Some</td>
<td>Yes</td>
<td>No</td>
<td>-</td>
</tr>
<tr>
<td>Emotion-labelled</td>
<td>No</td>
<td>327</td>
<td>Some</td>
<td>Yes</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 2.1: Widely used facial expression databases in research. CK: Cohn-Kanade [61], CK+: Extended Cohn-Kanade [75], MMI [86], JAFFE: Japanese Female Facial Expression [77], POFA: Pictures of Facial Affect [28].

2.2 Automated Facial Expression Recognition

The majority of research conducted in automatic analysis of facial expressions can be categorized into two main streams [83, 112]: The first deals with automated recognition of facial Action Units (AUs) defined under the Facial Action Coding System (FACS) [25] model; a widely used model for coding muscle actions in behavioral sciences. These AUs are mapped to the basic emotion categories using a high-level mapping such as the one provided in Table 2.2. The second direction pursues the
2.2 Automated Facial Expression Recognition

goal of discrete emotion detection by directly extracting features from the input image skipping the intermediary step of detecting facial muscles or AUs. Most of the existing efforts in this category focus on recognition of six basic emotions due to their universal properties, and the availability of the relevant training and test datasets. Recently, there have been a few tentative efforts to detect non-basic affective states from deliberately displayed facial expressions such as fatigue [56] and pain [3], and mental states such as agreeing, concentrated, interested, thinking, confused, and frustrated [30].

**FACS-based Facial Expression Analysis**

The Facial Action Coding System (FACS), developed by Ekman and Friesen in 1978 [25, 29], is a method of measuring facial activity in terms of facial muscle movements. FACS consists of over 44 distinct AUs corresponding to a distinct muscle or muscle group with 5 levels of intensity ranging from A to E. 30 AUs are related to the movement of facial muscles, 8 AUs describe different movements related to head orientation, and 4 AUs are defined to describe eye direction. Figure 2.2 illustrates the coded AUs and their descriptions for the upper facial region. The AUs can either occur singularly or in combination, and can be used to describe facial expressions [84] as shown in Table 2.2.

Although Ekman and Friesen inferred that prototypic expressions of emotion can be represented by specific combinations of AUs [27], emotion labels are not part of FACS and the original coding system is purely descriptive and does not map the facial actions into emotional states. In a study that compared the performance of human coders and machine analysis of AU-labeled emotion recognition [50], the authors argue that associating a set of AUs with a particular affective state regardless of the task and context leads to inaccuracies in classification. They further suggest
2.2 Automated Facial Expression Recognition

that investigating the interplay between AUs in a sequence for a given affective state might be more useful than one-to-one or many-to-one association between AUs and an affective state. Additionally, FACS-coding requires extensive training; approximately 100 minutes for each minute of video and almost 300 hours of training to achieve minimal competency in certified FACS coding. Therefore, a FACS based-approach is far from ideal for building a flexible, accurate and robust emotion recognition system that does not require training in physiology, psychology, and emotion theory. Nevertheless, FACS allows the facial activity analysis to be conducted in an objective and quantitative way, and therefore has been the ground for major research in facial expressions detection and measurement.

Most of the existing automated facial expression recognition (FER) systems employ various pattern recognition approaches and typically consist of three modules.
### 2.2 Automated Facial Expression Recognition

<table>
<thead>
<tr>
<th>Expression</th>
<th>AUs coded description [84]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>$4 + 7 + (((23 \text{ or } 24 \text{ with or not } 17) \text{ or } (16 + (25 \text{ or } 26))) \text{ or } (10 + 16 + (25 \text{ or } 26))) \text{ with or not } 2$</td>
</tr>
<tr>
<td>Disgust</td>
<td>$((10 \text{ with or not } 17) \text{ or } (9 \text{ with or not } 17)) + (25 \text{ or } 26)$</td>
</tr>
<tr>
<td>Fear</td>
<td>$(1 + 4) + (5 + 7) + 20 + (25 \text{ or } 26)$</td>
</tr>
<tr>
<td>Happiness</td>
<td>$6 + 12 + 16 + (25 \text{ or } 26)$</td>
</tr>
<tr>
<td>Sadness</td>
<td>$1 + 4 + (6 \text{ or } 7) + 15 + 17 + (25 \text{ or } 26)$</td>
</tr>
<tr>
<td>Surprise</td>
<td>$(1 + 2) + (5 \text{ without } 7) + 26$</td>
</tr>
</tbody>
</table>

Table 2.2: Facial Expression Synthesis Rules from AUs as proposed in [84]

The first module is a pre-processing stage to detect/locate the facial region in input images or videos. The detected face region is usually aligned based on the eye positions in the detected face region. In our case, we are interested in accurately determining the eye locations in order to extract eye region from the facial region. A thorough review of the various eye and eye region detection techniques is provided in Appendix A. The second module deals with 2D or 3D spatiotemporal facial feature extraction from still images or a video stream. The extracted features are either geometric features such as the shape of the facial components (eyes, mouth) and the locations of facial fiducial points (corners of the eyes, mouth), or appearance features representing the texture of the facial skin including wrinkles, bulges, and furrows. Finally, the third module attempts to classify these generated feature vectors into one of the pre-defined emotion classes by maximizing inter-class differences and minimizing intra-class variation. The following describes the prior art in automated facial expression recognition broken down by the type of feature extraction involved.
2.2 Automated Facial Expression Recognition

2.2.1 Geometric-Feature-Based

Valstar and Pantic [100] proposed a fully automated facial expression recognition system by first detecting, and then tracking 20 facial fiducial points in a video stream (Figure 2.3). The temporal segments of these fiducial points are fed through Support Vector Machines (SVMs) to recognize the activation of a single or a combinational presence of 15 pre-defined AUs. The main drawback of this system lies in its inability to detect all AUs. Chang et al. [9] defined an expression manifold by using a shape model consisting of 58 facial landmarks to detect and track facial expressions. In [97], Tian et al. considered several types of geometric facial features, location features, and shape features. Specifically, six location features (eye centers, eyebrow inner endpoints, and corners of the mouth) are extracted and transformed into 5 parameters. Recently, Kotsia and Pitas [64] presented a semi-automatic FER system, where a parametrized face mask known as Candide grid is manually placed on a neutral face image, and then tracked using Kanade-Lucas-Tomasi (KLT) to a fully expressive face. The deformation values of the grid parameters are classified using SVMs to detect combinations of AUs for expression recognition.

Figure 2.3: (a) 20 fiducial points defined by Valstar et al. [100]. (b) 58 facial landmarks used by Chang et al. [9]
2.2 Automated Facial Expression Recognition

There also have been some research efforts focused on using geometric features in image sequences or videos [13, 84, 98]. Here, facial movements can be qualified by measuring the geometrical displacement of facial feature points between the current frame and the initial frame (usually corresponding to the neutral expression). Tian et al. [98] developed multi-state facial component models to track and extract the geometric facial features, including lips, eyes, brows, and cheeks. Cohen et al. [13] adopted a model-based face tracker to track head motion and local deformation of facial features such as the eyelids, eyebrows, and mouth. The tracked motions of various facial features at each frame are referred to as Motion-Units (MUs) (as shown in Figure 2.4). The MUs represent both the activation of a facial region, and the direction and intensity of the motion. They evaluated their MU-based system using Naive-Bayes, multi-level Hidden Markov Models (HMMs), and Tree-Augmented Naive Bayes (TAN) classifiers for both static and dynamic classification.

![Figure 2.4: The Motion-Units (MUs) introduced in [13].](image)

The geometric feature-based facial expression recognition systems strongly require precise and reliable facial feature extraction detection and tracking; a requirement that is difficult to accommodate in many real-world situations. More crucially, geometric features usually cannot encode the variations in skin texture such as wrin-
2.2 Automated Facial Expression Recognition

Klines and furrows that are considered critical for facial expression modeling. The microexpressions or subtle spontaneous facial expressions are especially difficult to describe using sparse geometric features. Several comparison studies [5, 20] suggest that appearance feature-based methods yield better recognition rates than geometric feature-based ones. Furthermore, Tian’s experiments [97] demonstrate that geometric features are not available or cannot be reliably computed from low-resolution facial images captured in real environments.

2.2.2 Appearance-Feature-Based

Gabor Features
Several existing appearance feature-based methods have adopted Gabor-wavelet features for image representation [6, 34, 44, 70, 71]. Gabor filters are obtained by modulating a 2D sine wave with a gaussian envelope [18], and the representations based on the outputs of Gabor filters at multiple scales, orientations, and locations have proven successful for facial image analysis. Littlewort et al. [70] used Gabor filters for feature extraction and SVMs for classification to recognize facial expressions pertaining to the basic emotions reporting 93% recognition accuracy on Cohn-Kanade’s DFAT-504 dataset. They also compared several approaches for expression classification including Adaboost, Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA), reporting that the proposed combination of Gabor features selected by Adaboost and SVMs outperformed the other approaches. The same system has also been employed for fully automated facial action coding [6], and pain facial expression analysis [71]. Guo and Dyer [44] used gabor features on manually marked 34 fiducial points to compare several classification methods including Simplified Bayes, Adaboost, Feature Selection via Linear Programming (FSLP), and SVMs for FER.
2.2 Automated Facial Expression Recognition

Recently, Fazli et al. [34], using Gabor filters in combination with Kernel PCA and SVM, reported a success rate of 89.9% on Cohn-Kanade dataset. Donato et al. [20] explored several different approaches such as PCA, LDA, Independent Component Analysis (ICA), Local Feature Analysis (LFA), and Gabor wavelet representation for facial action recognition. They reported that Gabor-wavelet-based and ICA representations outperformed the others, providing evidence for the importance of using local filters and statistical independence for classifying facial actions.

Although Gabor filters have been widely adopted for feature extraction in FER systems, it is computationally expensive to convolve image with a set of Gabor filters to extract multi-scale and multi-orientation coefficients. For example, a Gabor-wavelet representation derived from a $48 \times 48$ face image has a high dimensionality of $O(10^5)$. There is high redundancy involved that makes it inefficient with regards to both time and memory costs, and in turn, unsuitable for expression recognition systems that require very fast feature extraction and low memory footprint.

**Active Appearance Models**

Active Appearance Models (AAMs) are a powerful tool in representing non-rigid image objects such as faces and organs in a low dimensional subspace. Acting as a hybrid approach between geometric and appearance features, an AAM contains a statistical model of both the shape and the grey-level appearance (texture) of the object of interest [14, 78]. Then, given a target image, it updates itself by minimizing the difference between the target image and a synthesized model example projected onto the image repeating this process iteratively (usually lasting a few iterations, even from poor starting estimates). The use of AAMs have been explored by Lucey et al. [76] for action unit recognition, and more recently by Ashraf et al. [3] for pain expression recognition. Utilizing AAMs, Lanitis et al. [65] characterize face deformation due
2.2 Automated Facial Expression Recognition

to facial expressions by a set of appearance parameters that are used to recognize facial expressions. Huang and colleagues [13, 107] used a 3D face tracker called the Piecewise Bezier Volume Deformation Tracker [95] to extract facial texture images.

Local Binary Patterns (LBP)
The Local Binary Pattern (LBP) operator, originally introduced for texture analysis [81], has proved to be a powerful approach to describe local structures of an image [82]. LBP’s tolerance to monotonic illumination variations and its computational simplicity have made it a popular technique for facial feature analysis in recent years [51], especially in facial expression recognition [51, 92]. Feng et al. [35] described a coarse-to-fine facial expression classification scheme using LBP. More precisely, at the coarse stage, a seven-class problem was first reduced to a two-class one, and later at the fine stage, a k-NN classifier performed the final decision. Their approach produced 77% average recognition accuracy on JAFFE dataset. Later on [36], with the same facial description, a linear programming technique was applied for expression classification. A seven-class problem was decomposed into 21 binary classifications by using the one-against-one scheme. With this method, they obtained 93.8% accuracy on the JAFFE database.

Shan et al. [91, 92] empirically evaluated LBP features with weighted chi-square template matching, SVMs, LDA, and Linear Programming classification techniques in regular and low-resolution images, concluding that LBP feature-based face representation outperformed Gabor features for FER (92.6% versus 89.8% on the Cohn-Kanade dataset using SVMs). He et al. [48] used LBP on four kinds of frequency images decomposed by Gabor wavelets for facial expression recognition, reporting increased performance than using LBP directly on the JAFFE dataset. Liao et al. [68] extracted LBP features in both intensity and gradient maps in order to consider
multiple cues. They computed the Tsallis entropy of the Gabor filter responses as the first feature set and performed null-space LDA for the second feature set achieving 94.59% accuracy for images of 64×64 pixels, and 84.62% for 16×16 pixels on the JAFFE database using SVMs as the classification tool.

Some variants of LBP have also been used to classify facial expressions from static images and image sequences. In [113], volume local binary patterns (VLBP) and local binary patterns on three-orthogonal planes (LBP-TOP) were employed on video sequences as a means to recognize facial expressions using dynamic texture. The evaluation was conducted over a range of image resolutions and frame rates and a recognition rate of 96.26% was achieved on the Cohn-Kanade database demonstrating that both approaches outperform other state-of-the-art methods. AdaBoost technique was used to learn the principal appearance and motion from the spatiotemporal descriptors. Recently, Jabid et al. [53] proposed an alternative version of the LBP, called the local directional patterns (LDP) to extract facial features for facial expression recognition and reported better performance (96.4% on the Cohn-Kanade dataset using SVMs) than the LBP counterpart [91]. LDP essentially computes the edge response values of a pixel in different directions to encode the image texture.

As described above, LBP and LBP variants have proved to be one of the most widely used feature representation techniques in recent years in the FER research community. These techniques have gained popularity due to their ability to effectively describe local characteristics of the facial image, their robustness to illumination changes, and their computational simplicity. Thus, we have selected grayscale and color LBP feature descriptors as the basis for our proposed emotion recognition systems from eye region images. The grayscale and color LBP operators are described in detail in Chapter 3 followed by the system descriptions in Chapter 4 and 5.
2.3 Summary

In this chapter, we presented the theory and background concepts behind facial expression recognition. We then presented a survey of the prior-art in automatic facial expression recognition based on the type of features used to represent the face images: geometric and appearance features. From a technical standpoint, these systems and the machine learning methods that they employ, are most relevant to our work in automated emotion recognition from eye region signals. We have attempted to summarize their methodology, results, and our evaluation of these systems. The major points from these chapter can be summarized as follows:

- Ekman’s model of basic emotional subspace (anger, disgust, fear, joy, sadness, and surprise) [26] has become the most prevalent model for measuring emotion, and the facial expressions associated with these basic emotions have dominated the study of facial expressions over the last four decades. Most of the existing efforts in facial expression recognition focus on recognition of these six basic emotions due to their universal properties, and the availability of the relevant training and test datasets.

- The majority of research conducted in automatic analysis of facial expressions can be categorized into two main streams [83, 112]: (1) automated recognition of facial action units as defined by FACS, and (2) discrete emotion detection by directly extracting features from the input image. Although FACS has been the ground for major research in facial expressions detection and measurement, FACS-coding requires extensive training and is time-consuming; approximately 100 minutes for each minute of video and takes almost 300 hours of training to achieve minimal competency in certified FACS coding. Therefore, a FACS
based-approach is far from ideal for building a flexible, accurate and a robust emotion recognition system that does not require training in physiology, psychology, and emotion theory.

- The extracted features in FER systems are mainly either geometric features such as the shape of the facial components (eyes, mouth) and the locations of facial fiducial points (corners of the eyes, mouth), or appearance features representing the texture of the facial skin including wrinkles, bulges, and furrows. The geometric feature-based facial expression recognition systems strongly require precise and reliable facial feature extraction detection and tracking; a requirement that is difficult to accommodate in many situations. Several comparison studies [5, 20] suggest that appearance feature-based methods yield better recognition rates than geometric feature-based ones.

- For appearance-based features, Gabor filters and LBP features have been widely adopted for feature extraction in FER systems. The Gabor filters are found to be computationally expensive and have a high memory footprint. On the other hand, LBP and LBP variants have proved to be one of the most widely used feature representation techniques in recent years in FER research community [92]. They have gained popularity due to their ability to effectively describe local characteristics of the facial image, their robustness to illumination changes, and their computational simplicity.

In the next chapter, the traditional grayscale LBP operator is described and a new color LBP operator called local color vector binary patterns (LCVBP) is introduced, which will form the basis of our proposed systems in Chapters 4 and 5.
Chapter 3

Image Representation

For expression or emotion recognition, we aim to find a subspace representation (linear or non-linear) from the input signal that allows us to best discriminate the inter-expression variation for different individuals under the feature extraction module. This module converts raw pixel data into a higher-level representation of shape, motion, color, texture, and spatial configuration of the face or its components. This extracted representation is subsequently used for classification. The original image space provides us with large dimensional space (e.g., an image of size $640 \times 480$ pixels gives us the total vector length of 307200 dimensions) making the data unfit for direct classification. It is essential that we make use of statistical methods to extract and select certain features from the input image that contain sufficient information to correctly determine the input class.

Chapter 2 provided an overview of the various state-of-the-art facial expression recognition systems based on different (geometric-based and appearance-based) feature descriptors. We have selected Local Binary Patterns (LBP) as the feature descriptor of choice for our proposed system that operates on eye region signals. The rest of this
chapter discusses the features descriptors employed for image representation in this work: The traditional LBP operator that operates on grayscale images, and a new extension to the LBP operator, Local Color Vector Binary Patterns (LCVBP), that incorporates color information.

3.1 Grayscale Local Binary Patterns (LBP)

The Local Binary Pattern (LBP) operator, originally introduced by Ojala et al. [81], is a powerful way of describing the texture of an image. The simple LBP operator labels the pixels of an image by thresholding a $3 \times 3$ neighborhood of each pixel with the center value and considering the results as an 8-bit binary number or an LBP label for that pixel (as illustrated in Figure 3.1). These computed binary numbers can be used to represent different local primitives such as corners, curved edges, spots, flat areas, etc. (See Figure 3.2). The 256-bin histogram of the LBP labels computed over a region (or an image) is then used as a texture descriptor for that region (or an image).

![Figure 3.1: The basic LBP operator.](image)

The small $3 \times 3$ neighborhood limits the capturing of dominant features with large scale structures. Therefore, the basic LBP operator was extended to use neighborhoods of various sizes [82]. The operator can be applied on circular neighborhoods.
3.1 Grayscale Local Binary Patterns (LBP)

Figure 3.2: Example of texture primitives encoded by LBP (black circles represent zeros and white circles represent ones).

around a pixel by bilinearly interpolating the pixel values allowing any radius value, \( R \), and any number of equally spaced pixel samples in the neighborhood, \( P \). This LBP operator can be formalized as \( LBP_{P,R} \) as shown in Figure 3.3 with different values of \( P, R \).

Figure 3.3: LBP circular neighborhood examples. From Left to Right: \((8,1)\), \((8,2)\), and \((16,2)\) circular neighborhoods respectively.

The value of the LBP code of a pixel \( z_c \) can then be formalized by:

\[
LBP_{P,R}(z_c) = \sum_{p=0}^{P-1} \delta(g_p - g_c)2^p
\]

(3.1)

where \( g_c \) corresponds to the gray value of the center pixel \( z_c \), \( g_p \) refers to the gray values of \( P \) equally spaced pixels on a circle of radius \( R \), and \( \delta(x) \) defines a thresholding
3.1 Grayscale Local Binary Patterns (LBP)

function as follows:

$$\delta(x) = \begin{cases} 
1 & \text{if } x \geq 0 \\
0 & \text{otherwise}
\end{cases}$$  \hspace{1cm} (3.2)

For $P$ neighboring samples, the $LBP_{P,R}$ operator produces $2^P$ labels corresponding to the $2^P$ different output binary words. It has been shown that there is a subset of $2^P$ labels exists that encodes most of the texture information than the others \[82\]. This subset of fundamental patterns has been named as *uniform patterns* or $LBP_{P,R}^{u2}$. An LBP is called uniform if it contains at most two bitwise transitions from $0\rightarrow1$ or vice versa (e.g., 00000000, 00010000, 11100001). The rest of the patterns which have more than two bitwise transitions are accumulated into a single bin. Experiments conducted in \[82\] conclude that uniform patterns account for nearly 90% for all patterns in $(8,1)$ neighborhood and about 70% in $(16,2)$ neighborhood. The number of uniform patterns, $U(P)$, for $P$-pixel neighborhood can be calculated as follows:

$$U(P) = P(P - 1) + 2$$  \hspace{1cm} (3.3)

Using (3.3), for an 8-pixel neighborhood we get 58 uniform patterns, yielding 59 total bins, and for a 16-pixel neighborhood we get 242 uniform patterns, thereby yielding 243 total bins. Once LBP labels have been calculated for each pixel, a histogram $h_l$ of the labeled image $L(z_c)$ can be defined as:

$$h_l = \sum_{z_c} B(L(z_c) = l), \quad l = 0, ..., (U(P) + 1) - 1$$  \hspace{1cm} (3.4)

where $U(P)$ is the number of different labels produced by the LBP operator given by 3.3 and
3.1 Grayscale Local Binary Patterns (LBP)

\[ B(x) = \begin{cases} 
1 & \text{if } x \text{ is true} \\
0 & \text{if } x \text{ is false} 
\end{cases} \quad (3.5) \]

Previously, it has been shown that facial images can be seen as a composition of micro-patterns which can be effectively detected by the LBP operator [51], especially applied towards face detection [45, 73], face recognition [1, 45], and facial expression recognition [91, 92]. Therefore, it is intuitive for us to use LBP features to represent eye region images in our work. A LBP histogram computed over the entire eye region image encodes only the occurrences of the micro-patterns without any indication about their locations. To incorporate location information, local LBP histograms can be extracted by dividing the eye region image into \( M \) small regions, \( R^0, R^1, \ldots, R^{M-1} \), and the LBP operator is applied to each individual region. Using 3.4, the regional LBP descriptor for a local region \( R^m \) can be written as follows:

\[ h^m = \left[ h_{(m,0)}, h_{(m,1)}, \ldots, h_{(m,U(P))} \right]^T \quad (3.6) \]

where, \( m = 0, 1, \ldots, M - 1 \), and \( T \) is the transpose operator. The individual histograms, \( h^m \), are then concatenated together into a single column vector to generate a spatially enhanced histogram (using Equation 3.7) representing the signature of that image. The extracted feature histogram not only represents the texture at a local level, but also encodes the global shape/description of the eye region images.

\[ H = [(h^0)^T (h^1)^T \ldots (h^{M-1})^T]^T \quad (3.7) \]

This LBP based image representation applied on grayscale images forms the basis of our first proposed pipeline for the emotion recognition system. Next, we present an extension to the LBP features that includes and makes use of the color information contained in the images.
3.2 Local Color Vector Binary Patterns (LCVBP)

Color is an important feature that can be used to derive visual cues and information for delienation and recognition of objects [40]. Given the effectiveness of grayscale texture features for face detection, face recognition (FR), and facial expression recognition, advances have been made to incorporate color information into the extraction of texture features mostly to improve the performance of FR systems [11, 58, 72]. These works primarily concentrate on extending grayscale texture operations (e.g., LBP and Gabor filters) to multiple spectral-band (color channel) seperately, and are therefore limited to encoding the texture patterns of only color pixel variations derived from each individual spectral-band. Recently, Lee et al. [67] proposed a new and sophisticated face descriptor, called Local Color Vector Binary Patterns (LCVBP) for FR purposes and demonstarted its effectiveness and increased performance over other state-of-the-art FR systems. LCVBP extends the idea of traditional LBP features, by extracting two different texture patterns for each pixel via LBP texture operation: (1) color norm patterns which relates to the magnitude of the color pixel values, and (2) color angular patterns of the mutliple inter-band angles (one per pair of different
3.2 Local Color Vector Binary Patterns (LCVBP)

color bands) which are based on the spatial interactions among the color bands of an image.\footnote{The LCVBP feature extraction technique builds upon the fundamental concept of LBP features; therefore, careful reading of Section 3.1 is highly recommended before moving forward.} We have selected LCVBP feature descriptor to form the basis of the second pipeline for our proposed system due to its rich functionality and promising results in an FR system.

![Figure 3.5: Example of how a color vector $c$ is constructed for a center pixel $z_c$.](image)

### 3.2.1 Color Norm Patterns

Let $I$ be a $K$-channel ($K = 3$ for RGB, YCbCr) color image, and $I_k$ be the individual $k$-th spectral band image (e.g., $I_1 =$ spectral component Y from YCbCr). In order to compute the color norm patterns, a color vector, $c$ is defined at a center pixel location $z_c = (x, y)$. The color vector is denoted as $c = [c_1, ..., c_K]^T$, where $c_k$ is a color pixel value at $z_c$ of the associated $I_k$ band and $T$ is the transpose operator (as shown in
3.2 Local Color Vector Binary Patterns (LCVBP)

Figure 3.5). In addition, let \( c^{(p)} \) denote the color vectors defined for \( P \) equally spaced pixel locations on a circular neighborhood of radius \( R \) around the center pixel \( z_c \). The norm of the color vector \( c \) can be calculated as follows:

\[
r_c = \| c \| = \sqrt{c_1^2 + c_2^2 + \ldots + c_k^2}
\]  

(3.8)

Similar to 3.1 and 3.2, the LBP operator for extracting color norm pattern at center pixel location \( z_c \), as illustrated in Figure 3.6, can then be defined as:

\[
LBP_{cn^{P,R}}(z_c) = \sum_{p=0}^{P-1} \delta(r_{c^{(p)}} - r_c)2^p
\]  

(3.9)

where \( r_c \) corresponds to the color norm value of the center pixel \( z_c \), \( r_{c^{(p)}} \) refers to the color norm values of \( P \) equally spaced pixels on a circle of radius \( R \), and \( \delta \) defines a thresholding function as follows:

\[
\delta(x) = \begin{cases} 
1 & \text{if } x \geq 0 \\
0 & \text{otherwise}
\end{cases}
\]  

(3.10)

Figure 3.6: Visualization of color norm patterns extraction.
Once LBP labels have been calculated for each pixel, a histogram \( h_{cn} \) of the labeled image \( L_{cn}(z_c) \) can be defined as:

\[
h_{cn} = \sum_{z_c} B(L_{cn}(z_c) = l), \quad l = 0, ..., ((U(P) + 1) - 1) \quad (3.11)
\]

where \( U(P) \) is the number of uniform patterns produced by the LBP operator given by 3.3 and

\[
B(x) = \begin{cases} 
1 & \text{if } x \text{ is true} \\
0 & \text{if } x \text{ is false}
\end{cases} \quad (3.12)
\]

To incorporate location information, local LBP histograms are extracted by dividing the eye region image into \( M \) small regions, \( R_0, R_1, ..., R_{M-1} \), and the LBP operator is applied to each individual region. Similar to 3.6, the regional LBP descriptor for a local region \( R^m \) can be written as follows:

\[
h^m_{cn} = [h_{cn}^{(m,0)}, h_{cn}^{(m,1)}, ..., h_{cn}^{(m,U(P))}]^T \quad (3.13)
\]

In order to keep the information about the spatial relation of \( M \) eye region local regions, all of the \( h^m_{cn} \) are concatenated into a single column vector, resulting in the color norm LBP histogram:

\[
H_{cn} = [(h^0_{cn})^T, (h^1_{cn})^T, ..., (h^{M-1}_{cn})^T]^T \quad (3.14)
\]

### 3.2.2 Color Angular Patterns

Color angular patterns refer to the discriminative patterns contained between different spectral bands (color channels) of the color image. To this end, the ratio of pixel values between a pair of different spectral bands is defined as follows:
3.2 Local Color Vector Binary Patterns (LCVBP)

\[ r^{(i,j)} = \frac{c_j}{c_i + \gamma}, \quad \text{for } i < j, \quad i = 1, \ldots, K - 1, \text{ and } j = 2, \ldots, K, \quad (3.15) \]

where \( c_i \) and \( c_j \) are color pixel value elements of the color vector \( c \) associated with the \( i \)th and \( j \)th spectral bands of the color image respectively, and \( \gamma \) is a small valued constant to prevent division by zero.

![Diagram](image)

**Figure 3.7:** Visualization of the color angle between R and G spectral bands, \( \theta^{(R,G)} \).

Using (3.15), the color angle between the \( i \)th and \( j \)th spectral bands is defined by:

\[ \theta^{(i,j)} = \arctan(r^{(i,j)}), \quad 0 \leq \theta \leq 90^\circ \quad (3.16) \]

This color angle, \( \theta^{(i,j)} \), can also be described as the angle between the \( i \)th spectral band (the axis) and the reference line which is formed by projecting the color vector \( c \) onto the plane associated with the \( i \)th and \( j \)th spectral bands, as shown in Figure 3.7.

Similar to the color norm patterns, the LBP operator for the color angular pattern for the center pixel \( z_c \) can be defined as follows:
3.3 Summary

This chapter described in depth the two selected image representation techniques to extract relevant features for eye region images: (1) Local Binary Patterns (LBP), and (2) Local Color Vector Binary Patterns (LCVBP).

Traditional LBP features are mainly used for texture classification operating on grayscale images, and have been previously proven effective as a low-cost (time) and highly discriminative feature set for face detection [45, 73], face recognition [1], and...
facial expression recognition [91, 92]. Chapter 4 describes our LBP feature-based emotion recognition system for eye region images.

LCVBP extends the idea of LBP to include color information by extracting color norm (magnitude) patterns and color angular patterns (spatial interaction between different color channels). LCVBP, providing a rich feature set from color face images, has recently been proposed as a promising technique to improve the performance for face recognition purposes [67]. Chapter 5 illustrates our pipeline for emotion recognition from color eye region images based on the LCVBP feature descriptor.
Chapter 4

Grayscale LBP feature-based
System

The previous chapter outlined in detail the two image representations used in this work in terms of low-level features and their associated descriptors: Local Binary Patterns (LBP) and Local Color Vector Binary Patterns (LCVBP). Based on these two feature descriptors, we have developed two pipelines for our proposed emotion recognition system using eye region signals. This chapter describes our first pipeline that uses LBP features operating on grayscale eye region images.

4.1 System Architecture

The first system designed under this work attempts to recognize emotions from eye region expressions on grayscale still images using the traditional LBP features (described in Section 3.1). The system architecture used here, as illustrated in figure 4.1, describes a trainable solution that requires training data beforehand, and performs supervised classification consisting of two main phases: training (offline) phase and
4.1 System Architecture

classification (online) phase.

During the training phase, a collection of input images are processed offline using pre-processing and feature extraction modules in order to learn a set of discriminative features from the dataset and build a model that contains statistical characterization (signature) for each class (e.g., anger, joy, ...). In the classification phase, the input test image undergoes the same pre-processing and feature extraction (this time, in an online manner), and is then classified by examining the signature of this test image against the information from the previously learnt model, making a decision about which of the signatures it resembles most. The different modules used in both the phases, i.e., pre-processing, feature extraction, and classification, are described in further detail below.

4.1.1 Pre-Processing

Face and eye detection are the most fundamental steps for vision-based automatic face analysis tasks such as face tracking, face recognition, facial expression recognition. Even in databases made available for research, the face images have diverse resolutions, face sizes, backgrounds, and are captured under varying illumination conditions. In order to extract the relevant information from the facial or the eye region in a systematic manner, image pre-processing is necessary. The pre-processing module is responsible for localizing any present faces in an image or a video frame, and then detecting, localizing, and aligning the eye region within a localized facial region. The following are the pre-processing steps that are taken:

1. Generate a binary skinmap from the input image,

2. Convert the input three-channel (RGB) image into a single-channel grayscale image,
4.1 System Architecture

Figure 4.1: Overview of the system architecture for the grayscale LBP-based system.
3. Localize the facial region(s) by detecting any present face(s) with the help of the generated skinmap and a face detector,

4. Detect the location of two eyes within the detected faces,

5. Rotate the entire image to align the eye centers on a straight line,

6. Resize the entire image to have a fixed inter-ocular distance, $d$ pixels,

7. Extract the eye region based on this fixed inter-ocular distance, $d$.

**Skin-tone Detection**

Skin-tone detection plays a crucial role in wide array of image processing and computer vision applications ranging from face detection, face tracking, face recognition, gesture recognition, content-based image retrieval (CBIR) systems, and in various surveillance type and HCI applications. Skin-tone, due to its invariance to rotation and scale in various images [55], generally acts as a pre-processing tool or a cue in early stages of these higher-level systems. Here, we have made use of the “luminance-based” skin-tone detection model proposed in [10]. This model calculates the error signals, $e(x)$ from differentiating the “luminance” map and the non-red encoded “luminance” map, and makes the decision at the pixel level of skin versus non-skin using the following equation:

$$
e(x) = Y(x) - \arg\max(G(x), B(x))$$

$$f_{skin}(x) = 1, if \ 0.0251 \leq e(x) \leq 0.1177 \quad (4.1)$$

$$Y(x) = 0.2989R + 0.5870G + 0.1402B$$

This model was selected after thorough research and evaluation of the state-of-the-art skin-tone detection models against a dataset that was built to contain sufficiently
varied skin-types and illumination conditions. The detailed report on our skin-tone
detection analysis is presented in Appendix B. The binary skinmap generated from
the skin-tone detection module is fed to the face and eye detection module, where it
is used to remove any false positives (falsely detected face and eye detections).

**RGB to Grayscale Conversion**

The system implements the RGB to Grayscale scheme proposed by Lu and Plataniotis
[74]. This algorithm has been shown to work more effectively in face detection sys-
tems as compared to the standard NTSC conversion. Below is the equation referring
to the conversion:

\[
Y(x, y) = \alpha R(x, y) + \beta G(x, y) + \gamma B(x, y)
\]  

where \(Y(x, y)\) is the luminance/gray-scale value, \(\alpha = 0.85\), \(\beta = 0.10\), \(\gamma = 0.05\) are the
values proposed for the parameters, and \(R(x, y)\), \(G(x, y)\), and \(B(x, y)\) are the Red,
Green, and Blue component values at the pixel location \((x, y)\).

**Face and Eye Detection**

An intuitive algorithm for determining the location of an eye region within an image
can be described via a two-stage process. The first stage is to apply a face detector
that allows us to narrow down the uncertainty about the possible location of eyes
in the image plane, allowing the system to perform robustly in a wide variety of illu-
mination and background conditions. The second stage is to apply an eye detection
algorithm on the output of the first detector, making the eye detection context-specific
to the face. Due to the restricted context, this eye detector can achieve high location
accuracy and reduce false alarms as compared to applying an eye detector to operate...
4.1 System Architecture

on the entire image. This process is illustrated in Figure 4.2.

To this end, we have interfaced the MATLAB implemententation of the Eye Detection functionality - *EyeFinder* - present inside the Machine Perception ToolBox (MPT) developed by Ian Fasel at the MPLab at UCSD [32]. An intermediary step to reject false face detections is added to the *EyeFinder* between the face detection and the eye detection stage, where all the detected face candidate windows are combined together with the binary skinmap generated using the selected skin-tone detection module [10] and candidates with total skin pixels less than 30% of the window size are discarded as false detections.

![Figure 4.2: Face and eye detection flowchart. (a) Search the entire image for a face at multiple scales. (b) Search within the detected face for eyes at multiple scales.](image)

**Machine Perception Toolbox - EyeFinder**

Fitting the description of a two-stage eye detection system described above, the EyeFinder in MPT determines the eye locations in two-stages. The pseudo-code is described below in Algorithm 1. In the first stage, a face is detected by scanning the
4.1 System Architecture

Figure 4.3: Results from the MPT Eye detection system.

entire image through square patches of 24×24 pixels, shifting one pixel at a time until all possible patches of this size are scanned. Each larger scale is chosen to be 1.2 times the previous scale. Upon detection of a face, the search region is confined to the face boundary and a similar scan is conducted to output an eye versus non-eye log-likelihood ratio for each square patch. This ratio is then combined with the prior for probability of eye (calculated using location and size with respect to the face detection window) to produce a final log posterior of eye versus non-eye. The search patches for locating the eyes within the face are also restricted using the observed statistics of true eye positions with respected to the estimated face location. Some of the results from running the MPT code are shown in Figure 4.3.

Image Rotation and Eye Region Extraction

Once the two eye center locations \((x_1, y_1), (x_2, y_2)\) have been determined, the entire image is rotated about the mid-point of the line joining the two eye-centers to remove the head-tilt and align the \(y\)-coordinates of the these two points. The entire image is
Algorithm 1 Code flow for the MPT EyeFinder System

\begin{algorithm}
\caption{Algorithm 1 Code flow for the MPT EyeFinder System}
\begin{algorithmic}
\State {mpeyefinder $\rightarrow$ initStream(width, height)}
\State {mpeyefinder $\rightarrow$ faceSearch(img, ...)}
\For {each face found}
\State {GPrior rightPrior = SetROI(currentface, righteye)}
\State {totalRightEyes = eyeSearch(..., currentface, rightPrior, righteye, ...)}
\State {GPrior leftPrior = SetROI(currentface, lefteye)}
\State {totalLeftEyes = eyeSearch(..., currentface, leftPrior, lefteye, ...)}
\{Make final hypothesis of best pair of eyes\}
\State {currentface $\rightarrow$ posterior(...)}
\EndFor
\end{algorithmic}
\end{algorithm}

the eye region is then extracted based on this distance, $d = 55$ pixels. The eye region is then resized to have a fixed interocular distance, $d = 55$ pixels. The eye region is then extracted based on this distance, $d$ as follows: $0.5 \times d$ in the horizontal direction on either side, $0.76 \times d$ on the top and $0.31 \times d$ on the bottom extracting a standardized rectangular eye region images of size $110 \times 60$ pixels as shown in Figure 4.4. This standardization is necessary in order to remove any position noises in appearance-based feature extraction [99]. The fixed interocular distance, $d = 55$ pixels has been chosen due to its proven effectiveness previously in face recognition [1], and facial expression recognition [91]. The parameters that encode the width and height of the extracted eye region image have been determined empirically to provide the maximum recognition rates across different experiments.
Figure 4.4: Parameters used to extract the eye region image. (a): Face image with eye centers. (b): Eye region extraction parameters based on the interocular distance, $d$. (c): Extracted eye region image.

4.1.2 Feature Extraction

Once the eye region image is extracted by the pre-processing module, the next step is to extract useful and sufficient information from the eye region image using the LBP operator previously described in Section 3.1.

Figure 4.5: Example of LBP operator to the eye region image.

The 110×60 eye region image is divided into 15 (5×3) blocks (as shown in Figure 4.6) with a fundamental block size of 22x20 pixels, determined empirically based on trade-
off performance between the recognition rate and feature vector length. In order to determine the most optimal values for the number of samples, \( P \), and the radius, \( R \), we have evaluated three different combinations for the LBP operator: \((P = 8, R = 1)\), \((P = 8, R = 2)\), and \((P = 16, R = 1)\).

Figure 4.6: LBP feature extraction on the eye region image. (Top): Regional feature histograms extraction obtained by applying the block-wise operator to the sub-regions of the eye region image. (Bottom): Concatenated global histogram representing the image.

For \( LBP_{8, R}^{u2} \), 58 out of the 256 patterns are uniform, thereby yielding 59 labels (number of histogram bins), and for \( LBP_{16, R}^{u2} \), 242 out of the 65536 patterns are uniform, giving us 243 labels. Thus, applying the \( LBP_{8, R}^{u2} \) operator on the eye region
image divided into 15 sub-regions, outputs a histogram of length 885 (59x15) and 3645 (243x15) for 8 and 16-pixel neighborhoods respectively. For the rest of this chapter, we will drop the superscript \( u^2 \) in \( LBP_{P,R}^{u^2} \), and refer \( LBP_{P,R}^{u^2} \) by \( LBP_{P,R} \), as we only make use of uniform patterns. In order to encode the global shape of the eye region image, the individual histograms are then concatenated together into a single column vector producing a single histogram representing the image.

### 4.1.3 Training and Classification

The classification process usually involves two phases: (1) building a model or leaning from the training data and (2) predicting the output class for an incoming test sample using the learnt model. For the purposes of this work, two different classifiers namely, Nearest-Neighbor and Support Vector Machines (SVMs) are chosen to evaluate the system as described below.

**\( \chi^2 \) Nearest Neighbor**

The nearest neighbor technique is a simple yet effective classification tool that selects the output class that has the smallest distance from the test sample (using a defined distance measure e.g., Euclidean, Chi-square (\( \chi^2 \)), Mahalanobis). Ahonen et al. [1] compared different dissimilarity measures for LBP histograms, and found the Chi-square statistic (\( \chi^2 \)) to be most effective, and therefore is selected for this work.

\[
\chi^2(I, T) = \sum_i \frac{(I_i - T_i)^2}{I_i + T_i} \tag{4.3}
\]

where \( I \) and \( T \) are input and test LBP histograms respectively; index \( i \) refer to \( ith \) bin in histogram. During the training phase, the LBP histograms of the eye region training images for each expression class are averaged to generate a histogram template or a model per expression. For a \( N \)-class emotion recognition system, \( N \)
different histogram templates will be generated during the training phase. During testing, a nearest neighbor classifier using the $\chi^2$ distance metric (defined by equation 4.4) is adopted where the LBP histogram of the input image is matched with the template with the minimum $\chi^2$ distance out of the $N$ templates. For expression recognition, it has been previously observed that certain areas of the face region provide more useful information than others [91]. Exploiting this fact, a weight measure, as illustrated in Fig. 4.7, is employed for each sub-region of the eye region image based on the amount of information it contains.

$$\chi^2_w(I, T) = \sum_{i,j} w_j \frac{(I_{i,j} - T_{i,j})^2}{I_{i,j} + T_{i,j}}$$  \hspace{1cm} (4.4)

where $I$ and $T$ are input and test LBP histograms respectively; indices $i$ and $j$ refer to $ith$ bin in histogram corresponding to the $jth$ local region; $w_j$ is the weight for region $j$.

Figure 4.7: The weights assigned for the weighted $\chi^2$ dissimilarity measure. Black indicates a weight of 0.5, gray indicates a weight 1.0, and white indicates a weight a weight of 1.5.

**Support Vector Machines (SVM)**

SVM is an optimal discriminant method having its roots in the Bayesian learning theory [43, 102]. For the cases where it is difficult to estimate the density model in high-dimensional space, it is preferable to use the discriminant approach. SVM performs an implicit embedding of data into a high dimensional feature space, and
then finds a linear separating hyper plane with the maximal margin in this higher dimensional space to separate data.

Given a set of instance-label training examples $S = (x_i, y_i), i = 1, ..., l, y_i = \{-1, 1\}$, the SVMs classify the test data $x$ by learning nonlinear functions of the form:

$$f(x) = \text{sgn} \left( \sum_{i=1}^{l} \alpha_i y_i K(x_i, x) + b \right)$$  (4.5)

where $\alpha_i$ are Lagrange multipliers of a dual optimization problem that describe the separating hyperplane; $K(x_i, x)$ is a kernel function of the form $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$ where $\phi$ is a non-linear function that maps the data into a higher dimensional space; $b$ is the threshold parameter of the optimal hyperplane. The training samples $x_i$ with $\alpha_i > 0$ are called support vectors, and SVM finds the hyperplane that maximizes the distance between the support vectors and the hyperplane. Linear, polynomial, and Radial Basis Function (RBF) kernels are the most frequently used kernels.

As a powerful discriminative machine learning technique, SVMs have proved to be an effective classification tool for facial expression recognition [6, 30]. SVM classification is essentially a binary (two-class) classification technique, therefore multi-class classification in this work is accomplished by using one-against-one approach, which involves training $N \times (N - 1)/2$ binary classifiers to discriminate each expression against every other expression, and choosing the class selected by the majority of the classifiers. For this work, the publicly available LIBSVM library [8] is adopted for the SVM implementation which provides multi-class classification support using the one-against-one approach. The linear kernel is selected for initial experimentation which is defined by $K(x_i, x_j) = x_i^T x_j$. 

50
4.2 Experimental Setup

4.2.1 Dataset Description

Following most of the existing work on facial expression analysis described in Chapter 2, the experiments to evaluate and validate our LBP feature-based proposed system are conducted on Cohn-Kanade and MMI databases; two of the most comprehensive and widely used databases by the current facial expression research community. These databases are described in detail below.

Cohn-Kanade (CK+) Dataset
The latest version of Cohn-Kanade dataset, CK+ [75] contains 593 FACS-coded image sequences (digitized into 640x490 or 640x480 pixel arrays) of 123 subjects (18 to 50 years of age, 69% female, 81% Euro-American, 12% Afro-American, and 6% other groups) incorporating various facial expressions from neutral onset to peak formation. Some sample images from the CK+ database are shown in fig. 4.8(a). Out of these, 327 sequences contain emotion labels belonging to 7 prototypic expressions (anger, contempt, disgust, fear, happiness, sadness, and surprise) that can be used as the ground truth. For experimental evaluation in this work, we have selected a subset from the this dataset; 309 image sequences belonging to 106 subjects and 6 prototypic emotions (anger, disgust, fear, happiness, sadness, and surprise). Three peak image frames from these selected expression sequences were extracted to form the gallery dataset (training and testing) providing a total of 927 images (135 anger, 177 disgust, 75 fear, 207 happiness, 84 sadness, and 249 surprise).
Figure 4.8: Sample face expression images from the (a) CK+ and (b) MMI datasets.
4.2 Experimental Setup

MMI Dataset
The MMI face database [86] contains more than 1500 samples of both static images and image sequences of faces in frontal and profile view. The database consists of 30 profile view and 750 dual-view video sequences (varying from 40-520 frames), which when digitized, measure 24-bit 720x576 pixels. The database includes 19 students and staff members (44% female) ranging from 19 to 62 years old, belonging to either a European, Asian, or South American ethnic background. Figure 4.8(b) shows some sample images from the MMI database. For experimental evaluation under this work, we manually selected 48 image sequences of 18 subjects belonging to one to four prototypical emotions (anger, disgust, happiness, and surprise). Data from the other two classes (fear and sadness) was not included due to the lack of its availability for a sufficient number of different subjects. Similar to the CK+ subset, three peak frames of expression were extracted for each sequences giving rise to a gallery set of 144 images (42 anger, 27 disgust, 30 happiness, and 45 surprise).

4.2.2 Performance Measures
There are several factors that have an effect on evaluation of an automated expression recognition system such as the choice of training and test datasets, the choice of resampling method, and the number of classes. Based on our review of the literature presented in Chapter 2, we define three performance measures that can be used as evaluation criteria for the proposed systems: Accuracy, Generalization, and Speed below. This allows us to design and evaluate our experiments in a systematic manner.

Accuracy
The first performance measure is the recognition accuracy (RA) of the system, i.e.,
the number of samples recognized accurately by the system from a designed testset. Most of the work in automated facial expression recognition test their systems on a single corpus of images or video using cross-validation and bootstrapping [62], and report their recognition accuracy as the performance metric. In person-dependent tests, examples of an emotional expression of the same subject are used in both the training and the test set, resulting in high RA. In the more challenging person-independent tests, the data of one particular subject are heldout for testing, and the system is trained with the rest of the data. The RA usually drops down in such scenarios.

**Generalization**

Although the person-independent evaluations are performed on the test samples previously unseen by a system, there is a database bias introduced in the results due to the fact that training and test sequences were subject to the same recording procedures and environmental conditions. A more accurate measure of performance is obtained if systems are tested across different datasets. Generalization considers the systems performance when trained on one corpus and tested on previously unseen examples from a different corpus. It is an important predictor of the systems performance in a natural computing environment. The better the generalization ability of the system, the more feasible it is to train the system on some (limited) data-set then deploy it in different interaction scenarios, with many users, without having to re-train or calibrate the system.

**Speed**

For systems in general, the time it takes a system to produce its output, starting from the moment all relevant inputs are presented to the system, is called the latency or lag. Speed is a crucial aspect of an FER system, especially if it is intended to be used
with interactive interfaces. A system is real time if its latency satisfies constraints imposed by the application. In our evaluation, we present both the memory and time costs taken by a particular configuration and compare it with other state-of-the-art systems.

4.2.3 Experiments

The primary objective of the experiments of this chapter is to demonstrate the functionality, viability, and robustness of the proposed grayscale LBP feature-based automatic emotion recognition on eye region images. In light of this and the performance measures defined above, three sets of experiments are designed to evaluate the system performance, described as follows.

Experiment 1: Person-dependent recognition on CK+ dataset

The first experiment performs 6-class emotion recognition (anger, disgust, fear, joy, sadness, and surprise) on selected images from the CK+ dataset, and is designed to evaluate person-dependent performance by incorporating the samples from all the subjects in the training model. For many real-world applications, it is desirable that the system has the flexibility to adapt to and incorporate new samples into its trained model, thereby improving its generalized performance. For this purpose, we randomly partitioned the images from each expression class into two groups. Images from one group were selected to train a generic model. From the remaining half, one out of the three frames for each unique expression sequence were then also included in the training set in order to reflect the presence of new samples from the remaining subjects; the other two frames contributing to the test dataset. This process was repeated ten times and the reported performance are the averaged results over these ten iterations.
4.2 Experimental Setup

**Experiment 2: Person-independent recognition on CK+ dataset**

The second experiment also performs 6-class emotion recognition (anger, disgust, fear, joy, sadness, and surprise) on selected images from the CK+ dataset, but in this case, for the purposes of evaluating person-independent performance of the system. Here, we adopted a 10-fold cross-validation testing scheme in the experimentation. More precisely, the data from each expression class is randomly partitioned into ten groups of roughly equal number of samples. Nine groups were used as the training data to build a training model, while the remaining group was used as the test data. The above process was repeated ten times for each group in turn to be omitted from a training process. The average RA is reported over all the ten cases.

**Experiment 3: Cross-database recognition on CK+ and MMI datasets**

The third experiment investigates the robustness of the system by training the system on images from the CK+ dataset, and testing on the images from the MMI dataset performing 4-class recognition (anger, disgust, joy, and surprise) on data that does not exist in the trained model. We used 768 images belonging to four emotion classes (135 anger, 177 disgust, 207 joy, and 249 surprise) from the CK+ dataset to train the system, and performed testing using the selected 144 images from the MMI dataset (42 anger, 27 disgust, 30 joy, and 45 surprise).

We also define a common set of experimental conditions that are valid for the different set of experiments:

1. The system performance is reported with both manual and automatic eye detection. This includes first evaluating the performance of the automatic MPT eye detection module in isolation against the ground truth data, and then com-
paring the effect of any errors that the automatic eye detection module might incur on the performance of the entire system.

2. Three different LBP operators namely, $LBP_{8,1}$, $LBP_{8,2}$, and $LBP_{16,2}$ are applied to extract feature histograms in order to measure the performance trade-off between speed and accuracy across different experiments.

3. Two classification tools; the simplistic nearest neighbor with weighted chi-square dissimilarity measure ($\chi^2$-NN), and more sophisticated SVMs with a linear kernel (L-SVM) are applied to the system.

### 4.3 Results and Evaluation

#### 4.3.1 Module Performance: Automatic Eye Detection

The ground truth data for the eye center locations is provided in the CK+ dataset, and was manually determined for the MMI dataset. The performance of the automatic eye detection as an independent module is evaluated by comparing the results on the CK+ and MMI dataset images to the ground truth as presented in Table 4.1.

<table>
<thead>
<tr>
<th>Dataset</th>
<th># of Images</th>
<th># of TPs</th>
<th># of FPs</th>
<th>Mean Error $(x, y)$ (pixels)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CK+</td>
<td>927</td>
<td>927</td>
<td>39</td>
<td>(4.54, 5.30) (5.49, 5.21)</td>
</tr>
<tr>
<td>MMI</td>
<td>144</td>
<td>144</td>
<td>48</td>
<td>(6.08, 4.98) (5.70, 5.46)</td>
</tr>
</tbody>
</table>

Table 4.1: Automatic Eye Detection Module Performance. TP: True Positives, FP: False Positives.

On 927 images from the CK+ dataset, the module provides a 100% hit rate (true
4.3 Results and Evaluation

positives) along with 39 false positives. On 144 images from the MMI dataset, the module also provides a 100% hit rate (true positives) along with 48 false positives. The module outputs fairly precise eye center locations with the average deviation of 4 to 6 pixels from exact eye centre locations in both x and y directions. The relatively higher false positive to true positive ratio in the MMI dataset can be attributed to the fact that some of the images in the MMI dataset contain a mirror which contributes to a side profile face along with the main face. All the “outliers” or false positives from both the dataset images are easily discarded based on the size of the facial region detected.

4.3.2 Experiment 1: Person-dependent recognition on CK+ dataset

The results for the person-dependent 6-class recognition on the CK+ dataset for the different LBP parameters using the $\chi^2$-NN are presented in Table 4.2.

<table>
<thead>
<tr>
<th>Feature Descriptor</th>
<th>Classifier</th>
<th>Recognition Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>$LBP_{8,1}$</td>
<td>$\chi^2$-NN</td>
<td>66.4 ± 4.6% 63.7 ± 5.2%</td>
</tr>
<tr>
<td>$LBP_{8,2}$</td>
<td>$\chi^2$-NN</td>
<td>71.4 ± 6.5% 68.5 ± 5.2%</td>
</tr>
<tr>
<td>$LBP_{16,2}$</td>
<td>$\chi^2$-NN</td>
<td>74.1 ± 5.3% 70.8 ± 5.1%</td>
</tr>
</tbody>
</table>

Table 4.2: Results for the person-dependent 6-class recognition on the CK+ dataset using $\chi^2$-NN classification. **MED**: Manual Eye Detection, **AED**: Automatic Eye Detection.

Using the eye center locations provided with the CK+ dataset, we achieved an RA of 66.4%, 71.4%, and 74.1% for $LBP_{8,1}$, $LBP_{8,2}$, $LBP_{16,2}$ operators respectively.
Using automatic eye detection in the pre-processing module, the performance drops down by 3-4%. As $\chi^2$-NN classification is geometrical in nature, this performance dip can be attributed to the small error in eye center locations that the automatic eye detection module introduces.

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Anger</th>
<th>Disgust</th>
<th>Fear</th>
<th>Joy</th>
<th>Sadness</th>
<th>Surprise</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>68.6%</td>
<td>8.4%</td>
<td>2.7%</td>
<td>9.6%</td>
<td>10.7%</td>
<td>0</td>
</tr>
<tr>
<td>Disgust</td>
<td>14.3%</td>
<td>72.6%</td>
<td>0</td>
<td>11.0%</td>
<td>2.1%</td>
<td>0</td>
</tr>
<tr>
<td>Fear</td>
<td>7.5%</td>
<td>3.8%</td>
<td>51.7%</td>
<td>3.3%</td>
<td>14.6%</td>
<td>19.2%</td>
</tr>
<tr>
<td>Joy</td>
<td>7.8%</td>
<td>4.6%</td>
<td>0.7%</td>
<td>79.9%</td>
<td>5.6%</td>
<td>1.5%</td>
</tr>
<tr>
<td>Sadness</td>
<td>23.9%</td>
<td>0.7%</td>
<td>24.6%</td>
<td>4.3%</td>
<td>37.1%</td>
<td>9.3%</td>
</tr>
<tr>
<td>Surprise</td>
<td>1.3%</td>
<td>1.1%</td>
<td>7.3%</td>
<td>4.2%</td>
<td>2.1%</td>
<td>84.0%</td>
</tr>
</tbody>
</table>

Table 4.3: Confusion matrix of person-dependent 6-class recognition on the CK+ dataset. Methodology: Manual Eye Detection + $LBP_{8,2} + \chi^2$-NN

<table>
<thead>
<tr>
<th>Feature Descriptor</th>
<th>Classifier</th>
<th>MED</th>
<th>AED</th>
</tr>
</thead>
<tbody>
<tr>
<td>$LBP_{8,1}$</td>
<td>L-SVM</td>
<td>98.5 ± 1.0%</td>
<td>99.5 ± 0.3%</td>
</tr>
<tr>
<td>$LBP_{8,2}$</td>
<td>L-SVM</td>
<td>98.9 ± 0.7%</td>
<td>99.5 ± 0.4%</td>
</tr>
<tr>
<td>$LBP_{16,2}$</td>
<td>L-SVM</td>
<td>98.0 ± 1.7%</td>
<td>99.1 ± 0.9%</td>
</tr>
</tbody>
</table>

Table 4.4: Results for the person-dependent 6-class recognition on the CK+ dataset using SVM classification. MED: Manual Eye Detection, AED: Automatic Eye Detection.

The confusion matrix of 6-class recognition of our proposed method is shown in Table 4.3. We observe that the emotions Joy and Surprise can be recognized with high
4.3 Results and Evaluation

accuracy (≈ 80-84%), Anger and Disgust with relatively less accuracy (≈ 68-72%) while Fear and Sadness are easily confused with others.

Using SVMs as the classification tool, we report excellent results (RA ≥ 98%) with both manual and automatic eye detection across the board as shown in Table 4.4. An interesting observation to make here is that the performance is slightly better using $P = 8$ neighboring samples as compared to $P = 16$ samples for the LBP operator in contrast to the observation in the $\chi^2$-NN case. Thus, applying $LBP_{8,2}$ operator combined with SVM classification allows us for high RA along with very fast feature extraction and low dimensionality. The emotion specific recognition results are provided in the confusion matrix in Table 4.5.

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Anger</th>
<th>Disgust</th>
<th>Fear</th>
<th>Joy</th>
<th>Sadness</th>
<th>Surprise</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>97.7%</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2.3%</td>
</tr>
<tr>
<td>Disgust</td>
<td>0</td>
<td>100%</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Fear</td>
<td>0</td>
<td>0</td>
<td>95.8%</td>
<td>0</td>
<td>0</td>
<td>4.2%</td>
</tr>
<tr>
<td>Joy</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100%</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Sadness</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100%</td>
<td>0</td>
</tr>
<tr>
<td>Surprise</td>
<td>0</td>
<td>1.2%</td>
<td>0</td>
<td>1.2%</td>
<td>0</td>
<td>97.6%</td>
</tr>
</tbody>
</table>

Table 4.5: Confusion matrix of person-dependent 6-class recognition on the CK+ dataset. Methodology: Manual Eye Detection + $LBP_{8,2}$ + L-SVM.

4.3.3 Experiment 2: Person-independent recognition on CK+ dataset

Next, we conducted person-independent 6-class recognition on the images from the CK+ dataset using 10-fold cross validation as described in Section 4.2.3. First, we
apply the $\chi^2$-NN as the classifier and report the results in Table 4.6.

<table>
<thead>
<tr>
<th>Feature Descriptor</th>
<th>Classifier</th>
<th>Recognition Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>$LBP_{8,1}$</td>
<td>$\chi^2$-NN</td>
<td>62.5 ± 0.5% 60.1 ± 1.7%</td>
</tr>
<tr>
<td>$LBP_{8,2}$</td>
<td>$\chi^2$-NN</td>
<td>66.9 ± 1.9% 62.2 ± 0.6%</td>
</tr>
<tr>
<td>$LBP_{16,2}$</td>
<td>$\chi^2$-NN</td>
<td>68.6 ± 1.0% 65.6 ± 1.4%</td>
</tr>
</tbody>
</table>

Table 4.6: Results for the person-independent 6-class recognition trained and tested on the CK+ dataset using $\chi^2$-NN classification. **MED**: Manual Eye Detection, **AED**: Automatic Eye Detection.

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Anger</th>
<th>Disgust</th>
<th>Fear</th>
<th>Joy</th>
<th>Sadness</th>
<th>Surprise</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>59.2%</td>
<td>12.6%</td>
<td>2.2%</td>
<td>17.0%</td>
<td>8.9%</td>
<td>0</td>
</tr>
<tr>
<td>Disgust</td>
<td>11.3%</td>
<td>76.3%</td>
<td>0</td>
<td>10.7%</td>
<td>1.7%</td>
<td>0</td>
</tr>
<tr>
<td>Fear</td>
<td>9.3%</td>
<td>4.0%</td>
<td>30.7%</td>
<td>10.7%</td>
<td>28.0%</td>
<td>17.3%</td>
</tr>
<tr>
<td>Joy</td>
<td>6.3%</td>
<td>8.2%</td>
<td>1.4%</td>
<td>78.3%</td>
<td>2.9%</td>
<td>2.9%</td>
</tr>
<tr>
<td>Sadness</td>
<td>25.0%</td>
<td>1.2%</td>
<td>28.6%</td>
<td>7.1%</td>
<td>27.4%</td>
<td>10.7%</td>
</tr>
<tr>
<td>Surprise</td>
<td>0%</td>
<td>0.8%</td>
<td>4.8%</td>
<td>4.0%</td>
<td>3.6%</td>
<td>86.7%</td>
</tr>
</tbody>
</table>

Table 4.7: Confusion matrix of person-independent 6-class recognition on the CK+ dataset. Methodology: Manual Eye Detection + $LBP_{16,2} + \chi^2$-NN

The performance of the system, as compared to the person-dependent experiments (Table 4.2), only slightly decreases by 4-6% making the case for $\chi^2$-NN as a classifier that is able to generalize well to unseen data. On the other hand, with SVM as the classifier (Table 4.8), we observe that the recognition accuracy drops down to around 63-70% in the case of the person-independent classification, whereas we observed very
high rates (RA ≈ 98%) for person-dependent classification (Table 4.4).

<table>
<thead>
<tr>
<th>Feature Descriptor</th>
<th>Classifier</th>
<th>MED</th>
<th>AED</th>
</tr>
</thead>
<tbody>
<tr>
<td>$LBP_{8,1}$</td>
<td>L-SVM</td>
<td>70.4%</td>
<td>65.8%</td>
</tr>
<tr>
<td>$LBP_{8,2}$</td>
<td>L-SVM</td>
<td>67.3%</td>
<td>63.5%</td>
</tr>
<tr>
<td>$LBP_{16,2}$</td>
<td>L-SVM</td>
<td>69.4%</td>
<td>66.2%</td>
</tr>
</tbody>
</table>

Table 4.8: Results for the person-independent 6-class recognition on the CK+ dataset using SVM classification. **MED**: Manual Eye Detection, **AED**: Automatic Eye Detection.

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Anger</th>
<th>Disgust</th>
<th>Fear</th>
<th>Joy</th>
<th>Sadness</th>
<th>Surprise</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>67.4%</td>
<td>11.9%</td>
<td>3.0%</td>
<td>8.1%</td>
<td>9.6%</td>
<td>0%</td>
</tr>
<tr>
<td>Disgust</td>
<td>7.9%</td>
<td>75.7%</td>
<td>0%</td>
<td>14.7%</td>
<td>0%</td>
<td>1.7%</td>
</tr>
<tr>
<td>Fear</td>
<td>4.0%</td>
<td>1.3%</td>
<td>36.0%</td>
<td>5.3%</td>
<td>33.3%</td>
<td>20.0%</td>
</tr>
<tr>
<td>Joy</td>
<td>11.6%</td>
<td>9.7%</td>
<td>0%</td>
<td>75.4%</td>
<td>0%</td>
<td>3.3%</td>
</tr>
<tr>
<td>Sadness</td>
<td>29.7%</td>
<td>0%</td>
<td>22.6%</td>
<td>3.6%</td>
<td>33.3%</td>
<td>10.7%</td>
</tr>
<tr>
<td>Surprise</td>
<td>1.2%</td>
<td>1.2%</td>
<td>7.2%</td>
<td>2.8%</td>
<td>2.0%</td>
<td>85.5%</td>
</tr>
</tbody>
</table>

Table 4.9: Confusion matrix of person-independent 6-class recognition on the CK+ dataset. Methodology: Manual Eye Detection + $LBP_{8,1}$ + L-SVM.

In the SVM case, $LBP_{8,1}$ operator provides the best performance with RA = 70.4% for manual eye detection and with RA = 65.8% for automatic eye detection. In contrast to the person-dependent recognition case, the automatic eye detection module actually causes a slight decrease in the performance (≈4-5%) for all the LBP operators. The confusion matrices of the person-independent recognition for both
the classifiers are presented in Tables 4.7 and 4.9. It is interesting to observe that in both the cases, the emotion Surprise can be recognized with highest accuracy (85-86%), Disgust and Joy performing second best (≈ 75-78%), Anger with the third best accuracy while Fear and Sadness are again easily confused with others.

4.3.4 Experiment 3: Cross-database recognition on CK+ and MMI datasets

From the confusion matrices of the previous experiments on CK+ dataset, we observe that Joy, Surprise, Anger, and Disgust are the emotion classes that provide us with higher recognition accuracy than Fear and Sadness. Therefore, in order to visualize the true generalized performance of the system on data that does not exist in the trained model, we performed the third set of experiments on data from these four classes by training the system on images from the CK+ dataset, and testing it on MMI data. It is important to note that the training and classification performed here are on images taken under different environment conditions (distance from the camera, illumination, stimulus), which is rarely the case in experimental evaluations.

From the results presented in Tables 4.10 and 4.11, we observe the following. First, the recognition accuracy of the system drops down in this scenario (around 8-10% for $\chi^2$-NN and 7-10% for SVM) when compared to earlier results from person-independent classification in Experiment 2. This drop in performance, even though expected, is much lower than reported by other state-of-the-art systems as discussed in the next section. Secondly, the automatic eye detection module does not have a negative effect in comparison with manual eye labelling. In fact, there is a slight increase seen in case of the $\chi^2$-NN classifier. Third, we previously observed via tests on the CK+ dataset that the $\chi^2$-NN classifier generalizes better than the L-SVM by virtue of a lesser drop.
4.4 Discussion

in performance when moving to a person-independent evaluation. In contrast, here we observe that the L-SVM classifier also is capable of providing consistent generalization performance when evaluated for cross-database generalization, and provides better RA than the $\chi^2$-NN classifier.

<table>
<thead>
<tr>
<th>Feature Descriptor</th>
<th>Classifier</th>
<th>MED</th>
<th>AED</th>
</tr>
</thead>
<tbody>
<tr>
<td>$LBP_{8,1}$</td>
<td>$\chi^2$-NN</td>
<td>54.9%</td>
<td>57.6%</td>
</tr>
<tr>
<td>$LBP_{8,2}$</td>
<td>$\chi^2$-NN</td>
<td>56.3%</td>
<td>63.2%</td>
</tr>
<tr>
<td>$LBP_{16,2}$</td>
<td>$\chi^2$-NN</td>
<td>59.7%</td>
<td>67.4%</td>
</tr>
</tbody>
</table>

Table 4.10: Results for the 4-class cross-database recognition trained and tested on CK+ and MMI datasets respectively using $\chi^2$-NN classification.

<table>
<thead>
<tr>
<th>Feature Descriptor</th>
<th>Classifier</th>
<th>MED</th>
<th>AED</th>
</tr>
</thead>
<tbody>
<tr>
<td>$LBP_{8,1}$</td>
<td>L-SVM</td>
<td>63.9%</td>
<td>61.1%</td>
</tr>
<tr>
<td>$LBP_{8,2}$</td>
<td>L-SVM</td>
<td>57.6%</td>
<td>57.6%</td>
</tr>
<tr>
<td>$LBP_{16,2}$</td>
<td>L-SVM</td>
<td>60.4%</td>
<td>61.1%</td>
</tr>
</tbody>
</table>

Table 4.11: Results for the 4-class cross-database recognition trained and tested on CK+ and MMI datasets respectively using L-SVM classification.

4.4 Discussion

There are several factors that can have an affect on a vision-based expression recognition system such as the facial expression database used, degree of variance between
training and test datasets, and the experimental methodology. The lack of a common evaluation protocol and lack of sufficient details to reproduce the reported individual results make it difficult for a direct comparison between systems. Recent efforts like the first facial expression recognition and analysis challenge [101] are a promising step to facilitate a standardized and a fair comparison. Nevertheless, we attempt to evaluate our system performance alongside the state-of-the-art in this section, against the three performance measures: accuracy, generalization, and speed as described in Section 4.2.2.

**Accuracy**

For the 6-class person-dependent recognition tests performed on the CK+ dataset, the system achieved an excellent recognition accuracy of 99.5% even when using automatic eye detection. For many real-world applications, it is desirable that the system has the flexibility to adapt to and incorporate new samples into its trained model, thereby improving its generalized performance. Therefore, the eye region based automatic emotion recognition system can be applied to such scenarios with very high reliability.

With regards to person-independent classification, Shan et al. [90] reported RA = 79.1% using $\chi^2$-NN and = 92.6% using SVM (RBF kernel) of their LBP feature-based facial expression recognition system for 6-class recognition using 10-fold cross-validation on the Cohn-Kanade dataset. Further, they integrated Adaboost to learn the discriminative features of LBP histograms and reported an increased RA of 85.3% and 93.1% for $\chi^2$-NN and AdaSVM (RBF kernel) respectively [93]. This is also comparable to the best results (93.3%) reported in [70], where the authors used Gabor features selected by Adaboost and SVM classifiers. Here, for our eye region system, we are able to achieve RA = 68.6% for $LBP_{16,2} + \chi^2$-NN and 70.4% for $LBP_{8,1} +$
Table 4.12: Comparisons between different methods for 6-class person-independent recognition using 10-fold cross-validation. **CK**: Cohn-Kanade.

<table>
<thead>
<tr>
<th>Methodology</th>
<th>Input</th>
<th>Dataset</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>$LBP_{8,2} + \chi^2$-NN [91]</td>
<td>Face</td>
<td>CK</td>
<td>79.1%</td>
</tr>
<tr>
<td>$LBP_{8,2} +$ Adaboost + $\chi^2$-NN [93]</td>
<td>Face</td>
<td>CK</td>
<td>85.3%</td>
</tr>
<tr>
<td>$LBP_{16,2} + \chi^2$-NN (Proposed)</td>
<td>Eye Region</td>
<td>CK+</td>
<td>68.6%</td>
</tr>
<tr>
<td>$LBP_{8,2} +$ SVM (RBF) [91]</td>
<td>Face</td>
<td>CK</td>
<td>92.6%</td>
</tr>
<tr>
<td>$LBP_{8,2} +$ Adaboost + SVM (RBF) [93]</td>
<td>Face</td>
<td>CK</td>
<td>93.1%</td>
</tr>
<tr>
<td>Gabor + Adaboost + SVM (RBF) [70]</td>
<td>Face</td>
<td>CK</td>
<td>93.3%</td>
</tr>
<tr>
<td>$LBP_{8,1} +$ SVM (Linear) (Proposed)</td>
<td>Eye Region</td>
<td>CK+</td>
<td>70.4%</td>
</tr>
</tbody>
</table>

SVM configurations respectively. The comparison is provided in Table 4.12, and a couple of distinctions and conclusions can be made by comparing these results. First, our proposed system operates on eye region images as opposed to the entire face images thereby inherently containing much lesser information in the input data. In spite of this, we are still able to consistently achieve two-thirds, and in some cases even more, of the performance reported by other systems validating our claim that the eye region contains significant information regarding various expressions or emotions. Secondly, we believe that further exploration into techniques such as Adaboost, including temporal information over time, and experimenting with different SVM kernels (e.g., RBF, Polynomial) can lead to an increased performance.

**Generalization**

In order to visualize the true generalized performance of the system, we performed 4-class recognition by training the system on the CK+ dataset, and testing it on
MMI dataset. Our proposed system achieved the cross-database recognition accuracy of 67.4% for $LBP_{16,2} + \chi^2$-NN and 63.9% for $LBP_{8,1} + L$-SVM configurations respectively. We observe that the system performs consistently when tested on a different corpus that includes images taken under different environment conditions (distance from the camera, illumination, stimulus). The recognition rates are also consistent with previous cross-database performance previously reported by Bartlett et al.’s Gabor-wavelet feature-based [70] (60%) and Shan’s LBP feature-based [90] (51.1%) facial expression recognition systems, although in their case, a significant drop in performance (30-35%) is observed when moving from a single corpus to cross-database testing. A comparison for cross-database testing accuracy results between various systems is presented in Table 4.13. This advocates the fact that a system operating on eye region images might be able to generalize much better than a system that uses face images. Besides exploring different machine learning techniques, one way to proceed forward in order to increase the generalization performance of this system, will be to incorporate images and samples taken under different conditions into the training model.

<table>
<thead>
<tr>
<th>Methodology</th>
<th>Input</th>
<th>Train</th>
<th>Test</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>$LBP_{8,2} +$ Adaboost + SVM (RBF) [90]</td>
<td>Face</td>
<td>CK</td>
<td>MMI</td>
<td>51.1%</td>
</tr>
<tr>
<td>Gabor + Adaboost + SVM (RBF) [70]</td>
<td>Face</td>
<td>CK</td>
<td>POFA</td>
<td>60.0%</td>
</tr>
<tr>
<td>$LBP_{16,2} + \chi^2$-NN (Proposed)</td>
<td>Eye Region</td>
<td>CK+</td>
<td>MMI</td>
<td>67.4%</td>
</tr>
<tr>
<td>$LBP_{8,1} +$ SVM (Linear) (Proposed)</td>
<td>Eye Region</td>
<td>CK+</td>
<td>MMI</td>
<td>63.9%</td>
</tr>
</tbody>
</table>

Table 4.13: Generalization performance for cross-database recognition for different approaches.
4.4 Discussion

**Speed**

We compare the time and memory costs of the LBP feature extraction process in Table 4.14 for both 8 and 16-pixel neighborhoods applied on a $110 \times 60$ eye region image. The time costs were measured using the MATLAB implementation of LBP on a 64-bit machine running at 2.67GHz CPU speed and a 4GB of RAM.

<table>
<thead>
<tr>
<th>Methodology</th>
<th>Image (Size)</th>
<th>Feature Length</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>$LBP_{8,R}$ [90]</td>
<td>Face (110×150)</td>
<td>2478</td>
<td>0.03s</td>
</tr>
<tr>
<td>Gabor [90]</td>
<td>Face (110×150)</td>
<td>42650</td>
<td>30s</td>
</tr>
<tr>
<td>$LBP_{8,R}$ (Proposed)</td>
<td>Eye Region (110×60)</td>
<td>885</td>
<td>0.009s</td>
</tr>
<tr>
<td>$LBP_{16,R}$ (Proposed)</td>
<td>Eye Region (110×60)</td>
<td>3645</td>
<td>0.014s</td>
</tr>
</tbody>
</table>

Table 4.14: Time and Memory costs for 8 and 16 neighborhood samples in LBP feature extraction on a $110 \times 60$ eye region image, and comparison with the results listed in [90]. Our metrics are calculated using MATLAB implementation, Machine: 64-bit OS, 2.67GHz CPU, 4GB RAM.

The results show that while both the cases allow for very fast and real-time feature extraction (0.009s for P=8, and 0.014s for P=16), the memory costs are almost quadrupled from 885 dimensional vector for P=8 to a 3645 dimensional vector for P=16 neighboring samples. We also list the time and memory costs in [90] calculated on a $110\times150$ face image. It is observed that LBP features for the eye region image can be calculated much faster as compared to the LBP features for the bigger face image (0.009s versus 0.03s), and lie in a much lower dimensional space over LBP-face (885 versus 2478) and Gabor-face (885 versus 42650). Therefore, for an application that requires real-time recognition and imposes memory constraints, LBP
feature-based emotion recognition on eye region signals brings significant speed and memory benefits.

## 4.5 Summary

In this chapter, we proposed, demonstrated, and evaluated the grayscale LBP feature-based system for automatic emotion recognition from eye expressions. The proposed system consists of three main modules: (1) Pre-Processing, (2) Feature Extraction, (3) Training and Classification. The key findings can be summarized as follows.

For pre-processing, we evaluated the performance of the automatic eye detection as an individual module, and further, its effect on the performance of the entire system. Comparing the performance for both manual and automatic eye detection scenarios revealed the fact that the slight error introduced by automatic eye detection does not have a drastic affect the overall recognition accuracy, especially in the case of using SVMs as the classifier. This proves the robustness of the system in less constrained environments such as the multimedia browsing scenario, where pixel-level eye center detection accuracy might not be achievable.

We demonstrated the choice of the optimal system parameters by applying three LBP operators: $LBP_{8,1}$, $LBP_{8,2}$, and $LBP_{16,2}$ alongside two classification techniques: $\chi^2$-NN and SVM for the different set of experiments. We demonstrated via the second set experiments on the CK+ corpus, the $LBP_{8,1}$ operator combined with the SVM provides the best accuracy (RA = 70.4%), whereas $LBP_{16,2}$ outperforms the others when using $\chi^2$-NN as the classifier (RA = 68.6%).

The first set of person-dependent experiments on the CK+ dataset allowed us to show the functionality and the viability of our proposed system (RA = 99.5%) in application scenarios where samples for new expression-classes and individuals are avail-
able to train the system. Person-independent evaluation on the CK+ corpus showed us the unbiased performance of the $\chi^2$-NN classifier, and the potential of eye region as a significant contributor for emotion recognition. The third set of experiments performed recognition on unseen data taken under completely different environmental conditions, and showed the ability of the system to generalize well. Specifically, we compare the cross-database (Trained on CK+, Tested on MMI) generalization performance of our proposed system with other facial expression recognition systems (Table 4.13) and observe consistent findings with regards to the generalization ability to other datasets.

Based on the obtained results, we then provided a evaluation of our system with the help of key performance measures: Accuracy, Generalization, and Speed, and compared it with other facial expression recognition systems. In the next chapter, the second proposed pipeline, based on a new and sophisticated feature set called LCVBP, is presented that extends the idea of traditional LBP patterns to include color information in color images.
Chapter 5

Color LBP (LCVBP) feature-based System

The previous chapter presented a fully functional emotion recognition system using eye region signals based on grayscale LBP features. The LBP approach has been proven in the past as an effective technique for facial expression recognition (FER) due to its robustness to variations in illumination, and its high discriminative power due to different levels of locality. From the experimental evaluation in the previous chapter, we were able to demonstrate the usefulness of LBP in expression recognition using eye region images as well as its ability to generalize well in different environments through cross-database experiments.

Recently, many research efforts have been dedicated towards extending the idea of grayscale texture patterns to incorporate color information specifically for the purposes of face recognition (FR) [11, 58, 67, 72]. Local Color Vector Binary Patterns (LCVBP) [67] is one such technique that has been proven to increase the performance over other state-of-the-art FR systems. In this chapter, we present our second pipeline for emotion recognition that uses LCVBP features (described in detail in Section 3.2)
operating on color eye region images. While the main focus of the previous chapter was to establish the viability of using eye region images as a feasible signal for automatic emotion recognition using grayscale LBP patterns, our primary objective in this chapter is to investigate and evaluate the effectiveness of LCVBP as a suitable feature descriptor for eye region expression recognition.

5.1 System Architecture

The second system designed under this research work attempts to recognize emotions from eye region expressions on color images using the LCVBP features (described in Section 3.2). The system architecture, as illustrated in Figure 5.1, performs supervised classification comprising of two main phases: training (offline) phase and classification (online) phase.

During the training phase, a collection of input images are processed offline using pre-processing and LCVBP feature extraction modules in order to learn a set of most discriminative features from the dataset and build a model that contains statistical characterization (signature) for each class (e.g., anger, joy, ...). In the classification phase, the input test image undergoes the same pre-processing and LCVBP feature extraction (but in an online manner), and is then classified by examining the signature of this test image against the information from the previously learnt model, making a decision about which of the signatures it resembles most. During pre-processing, the eye region is first manually localized, normalized and cropped within the image. After which, the input three-channel RGB image undergoes color space conversion and is represented in an alternative three-channel color domain, RQCr. During feature extraction, the LCVBP operator is applied to the pre-processed eye region image to create a rich feature set of four feature histograms: one for the color norm patterns,
and three for color angular patterns. These feature histograms are then fused together into a single feature column by the feature selection module in a way that achieves the best possible recognition rate. The feature histograms from the input images are then used to build a model in the training phase, and used by the decision function in the classification phase in order to classify the input test image into one of the emotion classes. Each of the modules are described further below.

### 5.1.1 Pre-Processing

The pre-processing module is responsible for extracting a normalized and aligned eye region image from the input color image in a best suited color domain. Here, the input RGB image is first converted into a ‘RQCr’ color space (R taken from RGB, Q taken from YIQ, and Cr taken from YCbCr). The authors in [12] state that RQCr color space representation achieves the best performance across a number of different color-component configurations for the LCVBP feature-based FR system. The Q and Cr components are calculated using (5.1) and (5.2) respectively as follows:

\[
\begin{bmatrix}
Y \\
I \\
Q
\end{bmatrix} =
\begin{bmatrix}
0.2990 & 0.5870 & 0.1140 \\
0.5957 & -0.2745 & -0.3213 \\
0.2115 & -0.5226 & 0.3111
\end{bmatrix}
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix} \tag{5.1}
\]

\[
\begin{bmatrix}
Y \\
Cb \\
Cr
\end{bmatrix} =
\begin{bmatrix}
16 \\
128 \\
128
\end{bmatrix}
\begin{bmatrix}
65.4810 & 128.5530 & 24.9660 \\
-37.7745 & -74.1592 & 111.9337 \\
111.9581 & -93.7509 & -18.2072
\end{bmatrix}
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix} \tag{5.2}
\]

where R,G,B values are scaled to [0, 1].

The objective of the proposed LCVBP-based pipeline is to investigate a color based
5.1 System Architecture

Figure 5.1: Overview of the system architecture for the LCVBP feature-based system.
approach for eye region expression recognition, and in turn, evaluate the feasibility of LCVBP as a suitable feature descriptor for the same. Therefore, instead of focusing on integrating or building an automatic eye detection module, we manually label the eye center positions as the ground truth for the sake of accuracy. This allows us to bypass any errors that might be introduced by an automatic eye detection module and focus on LCVBP evaluation for the system. Once the eye center locations have been marked, the eye region image is extracted similar to the process described in Section 4.1.1.

The entire image is rotated about the mid-point of the line joining the two eye-centers to remove the head-tilt and align the $y$-coordinates of these two points. This rotated image is then resized to have a fixed interocular distance, $d = 55$ pixels. The eye region is extracted based on this distance, $d$ as follows: $0.5 \times d$ in the horizontal direction on either side, $0.76 \times d$ on the top and $0.31 \times d$ on the bottom extracting a rectangular eye region images of size $110 \times 60$ pixels. As described in Section 4.1.1, the parameters that encode the width and height of the extracted eye region image have been determined empirically to provide the maximum recognition rate.

### 5.1.2 Feature Extraction

The LCVBP operator is applied to the extracted eye region images to extract a rich feature set consisting of four sets of histograms for a three-channel color image: one related to the color norm patterns (CNP), and three related to the color angular patterns (CAP). The LCVBP feature descriptor is described in detail in Section 3.2. Similar to the feature extraction process described for LBP-based system, we undertake the following parameter values for the feature extraction:

- The $110 \times 60$ RQCr eye region image is divided into $m = 15$ ($5 \times 3$) blocks, with
a fundamental block size of 22×20 pixels,

- \((P = 8, R = 1)\), \((P = 8, R = 2)\), and \((P = 16, R = 1)\) sets of values are evaluated for the LCVBP operator in order to determine the most optimal performing set,

- Uniform patterns are made use of. That implies for \(P = 8\), we get a single histogram length of 885 \((59\times15)\), and for \(P = 16\), a histogram length of 3645 \((243\times15)\) respectively.

First, the LCVBP operator is applied locally to each one of these blocks to produce the CNP and CAP histograms at a block level. Figure 5.2 displays an example of the LCVBP operator applied to a block of the eye region image and the corresponding output local histograms. The local CNP histograms and similarly, the local CAP histograms are then concatenated for all the blocks to produce four globals histograms: \(H_{cn}\), the CNP global histogram, and \(H_{ca(i,j)} = (H_{ca(R,Q)}, H_{ca(Q,Cr)}, H_{ca(R,Cr)})\), the CAP global histograms.

### 5.1.3 Feature Fusion

The LCVBP feature descriptor applied to the three-channel color images provides us with a rich set of features (one CNP global histogram, and three CAP global histograms) that can be used to classify the images. Using multiple features in a system opens up several possibilities in which these features can be combined in order to achieve the best performance accuracy. In this section, we discuss some of the existing information fusion techniques that have been employed in the past for fusing multiple feature descriptors. Sanderson and Paliwal [89] classify information fusion in multi-modal biometric systems into two major categories: pre-classification
5.1 System Architecture

Figure 5.2: LCVBP operator applied to a region of an eye image with $P = 8, R = 2$. (a) Color Norm Pattern (CNP) Histogram, (b) R-Q, Q-Cr, R-Cr Color Angular Patterns (CAP) Histograms.
fusion and post-classification fusion.

In pre-classification fusion, the integration of information takes place prior to classification or matching either at the \textit{sensor level} or \textit{feature level}. \textit{Sensor level} fusion combines raw data from sensors into a single feature vector on the pre-condition that that the data obtained from the different sensors must be compatible. \textit{Feature level} fusion refers to combining different feature vectors that are obtained by either using multiple sensors or employing multiple feature extraction techniques on the same sensor data. In post-classification fusion, the information fusion takes place after the classifier stage. This can be divided into four major categories: dynamic classifier selection, fusion at the decision level, fusion at the rank level, and fusion at the matching score level. A thorough review of the post-classification techniques is provided in [54]. One of the more intuitive fusion schemes at the decision level to combine multiple descriptors is the use of boolean operators: Conjunction (AND) rule or Disjunction (OR) rule [16]. The conjunctive (AND) operator strictly requires the satisfaction of all criteria, and can be considered as severely pessimistic. In other words, classification based on the four histograms individually should output the same emotion class. The disjunctive (OR) operator requires atleast one of the criteria is satisfied, and can be considered as optimistic.

Jain \textit{et al.} [54] advocate the fact that integration of information at an early stage of processing is more effective than at a later stage because the features contain richer information about the input data than the matching score or the output decision of a classifier. The authors also support concatenating non-homogenous feature vectors (i.e., feature vectors obtained using different feature extraction techniques) into a single feature vector in order to maximize a complementary effect on recognition. Although, it might result in a feature vector with very large dimensions leading to the ‘curse of dimensionality’ problem [23], this feature level fusion technique is shown to
be effective in terms of both computational cost and performance in face recognition. Therefore, we employ fusion at the feature level as one of the techniques to concatenate the color norm and color angular pattern histograms into a single column feature vector. In section 5.3, we evaluate the system performance and the present the results using both CNP and CAP features separately and using the combination of all the features via the feature level fusion technique.

5.2 Experimental Setup

5.2.1 Dataset Description

Most of the work on facial expression analysis so far has been designed to operate on grayscale images, and in turn, use grayscale facial expression databases (e.g., Cohn-Kanade, JAFFE) not employing color information. Hence, the availability of color facial expression databases is also very limited. The MMI database, as previously described in Section 4.2.1, does contain three-channel (RGB) 24-bit images measuring 720x576 pixels. Figure 4.8(b) shows some sample color images from the MMI database. For experimental evaluation under this work, we manually selected 48 image sequences of 18 subjects belonging to one to four prototypical emotions (anger, disgust, happiness, and surprise). Data from the other two classes (fear and sadness) was not included due to the lack of its availability for a sufficient number of independent subjects. Three peak frames of expression were extracted for each sequences giving rise to a set of 144 images (42 anger, 27 disgust, 30 happiness, and 45 surprise).
5.2 Experimental Setup

Figure 5.3: Sample facial expression color images from the MMI dataset.
5.2 Experimental Setup

5.2.2 Objectives and Framework

The primary objective of the experiments of this chapter is to explore and demonstrate the viability of LCVBP as a suitable feature descriptor for emotion recognition on eye region images. In order to systematically evaluate the performance of the proposed system, the experimental framework is defined as follows:

1. 4-class recognition (anger, disgust, joy, and surprise) is performed on selected images from the MMI dataset. As our results indicated in Tables 4.3, 4.7, and 4.9, images from these four classes can be recognized with much higher accuracy as compared to the other two. Using data from this subset of four emotions instead of the traditional six allow us to perform preliminary evaluation of our LCVBP-based system without significant performance degradation.

2. Three different sets of parameters for the LCVBP operator namely, \(LCVBP_{8,1}\), \(LCVBP_{8,2}\), and \(LCVBP_{16,2}\) are applied to extract feature histograms to observe the most optimal combination for the number of neighborhood samples, \(P\), and the circle radius, \(R\).

3. SVMs with a linear kernel (L-SVM) is chosen as the classification tool due to its proven effectiveness in facial expression recognition in the past, and its reported performance in the grayscale LBP feature-based system in the previous chapter.

4. The total set of 144 images (42 anger, 27 disgust, 30 happiness, and 45 surprise) is divided into approximately two equal groups of half the number images from each expression class using random partitioning. The first group constitutes the training set while the other group is used as the testing set. In order to guarantee stable experimental results, this random partitioning is performed 10 times and therefore, all the results reported below were averaged over 10 runs.
5. The evaluation is carried out by measuring the system performance in the following three cases: (i) using color norm patterns alone, (ii) using color angular patterns alone, and (iii) combining color norm and color angular patterns at the feature level. Evaluation using the color norm and color angular patterns separately will allow us to determine their relative significance and contribution to the combined LCVBP system.

5.3 Results and Evaluation

The experimental results on the MMI dataset images using LCVBP features are reported in Table 5.1 (row 3). Along with the LCVBP features, we also report the system performance when using color norm and color angular pattern features separately (rows 1 and 2). The results reported in the last row of Table 5.1 refer to the performance when using grayscale LBP features as described in the previous chapter.

As shown in Table 5.1, we were able to achieve a maximum recognition accuracy of 86.3\% using $LCVBP_{8,2}$ feature descriptor. As observed in the case of grayscale LBP features, increasing the number of neighborhood samples from $P = 8$ to $P = 16$ actually causes a slight decrease in the overall system performance when SVMs are used as the classification tool, thereby making $(P = 8, R = 2)$ as the optimal choice of parameter values for LCVBP.

The backward compatibility of the LCVBP based system is demonstrated by observing the results obtained using grayscale LBP features in Table 5.1 (row 4). Using color information, such as the color norm or the color angular ratio instead of the grayscale value of the pixels, does provide consistent results across the range of $P$ and $R$ values, and are in the nearby range with those based on grayscale LBP.
5.3 Results and Evaluation

<table>
<thead>
<tr>
<th>Feature Set</th>
<th>LCVBP Parameters (P,R)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(8,1)</td>
</tr>
<tr>
<td>CNP features only</td>
<td>83.4%</td>
</tr>
<tr>
<td>CAP features only</td>
<td>85.7%</td>
</tr>
<tr>
<td>LCVBP (CNP + CAP) (Chapter 5)</td>
<td>86.1%</td>
</tr>
<tr>
<td>Grayscale LBP (Chapter 4)</td>
<td>88.1%</td>
</tr>
</tbody>
</table>

Table 5.1: Comparison of the results for the 4-class recognition on the MMI dataset using LCVBP features and Grayscale LBP features. CNP refers to color norm pattern features only, CAP refers to color angular pattern features only, LCVBP refers to the feature set that includes both CNP and CAP features fused together, and Grayscale LBP is our proposed methodology in chapter 4.

Although we observe that current grayscale LBP features outperform the LCVBP from the shown results, it is important to mention that the experiments are based on preliminary evaluation on a limited dataset, and we suspect that LCVBP performance should increase with a larger training set. This provides us confidence in the new and rich feature set and opens up the possibilities for future research.

From Table 5.1 and Figure 5.4, it can be observed that color angular patterns contribute much more than color norm patterns demonstrating their ability to provide highly discriminative features for eye expression recognition. Also, the fused LCVBP features provide better results than CNP and CAP alone for \((P = 8, R = 1)\) and \((P = 16, R = 2)\), and matching CAP results for \((P = 8, R = 2)\). This highlights the fact that the norm and angular texture features in a color image are able to provide different information and are mutually compensational for improving expression recognition performance from eye region images.
5.4 Summary

Figure 5.4: Comparison of recognition accuracy obtained for color norm and color angular pattern features separately with the fused LCVBP features.

5.4 Summary

In this chapter, we proposed and evaluated a new and rich feature descriptor based on color information called local color vector binary patterns (LCVBP) for emotion recognition from eye region signals operating in ‘RQCr’ color domain. LCVBP feature descriptor extends the traditional grayscale LBP operator by including color information from multiple channel images by extracting two sets of color features: (1) Color Norm Patterns (CNP), and (2) Color Angular Patterns (CAP). To the best of our knowledge, the aforementioned color based technique or any other parallel technique that explores the spatial interactions between different bands of a color image has not been explored previously by researchers in the context of facial or eye expression recognition, thereby outlining the novelty of our proposed system. The
LCVBP feature-based system consists of five main modules: Pre-Processing, Feature Extraction, Feature Selection, Training, and Classification. The key points from this chapter can be summarized as follows:

1. In order to evaluate the feasibility of a LCVBP as a suitable feature descriptor for eye region expression recognition in a systematic manner, we manually labeled the eye center positions as the ground truth for the sake of accuracy instead of focusing on integrating or building an automatic eye detection module.

2. We demonstrated the success of LCVBP as a suitable feature descriptor for emotion recognition on eye region images by performing 4-class recognition on a selected subset (144 images: 42 anger, 27 disgust, 30 happiness, and 45 surprise) of the MMI facial expression database. We were able to achieve an accuracy of 86.3% using $LCVBP_{8,2}$ as the feature extractor and L-SVM as the classifier (See Table 5.1 for results).

3. The backward compatibility of the LCVBP based system was shown by comparing the results obtained using grayscale LBP features in Table 5.1. Using color information, such as the color norm or the color angular ratio instead of the grayscale value of the pixels, we observed consistent results across the range of $P$ and $R$ values with those obtained using grayscale LBP features.

4. By evaluating the system performance on color angular pattern and color norm pattern features separately (as shown in 5.4, we observed that color angular patterns contribute much more than color norm patterns demonstrating their ability to provide highly discriminative features for eye expression recognition.

5. We also observed that the fused or combined LCVBP features provide better results than CNP and CAP separately. This highlights the fact that the norm
and angular texture features in a color image are able to provide different information and are mutually compensational for improving expression recognition performance from eye region images. We used information fusion at the feature level into a single feature vector in order to maximize a complementary effect on recognition.

LCVBP feature descriptor is a rich, novel, and a sophisticated tool for facial analysis in multichannel images that has already been proven effective for face recognition purposes. Through our preliminary evaluation on a rather limited dataset, we have exhibited its potential and backward compatibility (with grayscale LBP) as a suitable feature descriptor for expression recognition on eye or facial images. We have only scratched the surface with LCVBP in this chapter, and there is a great potential of exploration with more thorough and careful experimentation on larger databases and various pre and post-classification fusion techniques.
Chapter 6

Conclusions and Future Directions

This chapter first describes the principle contributions of this research, followed by several directions for future work, and a potential application for this work - *Implicit Tagging of Multimedia Content*.

6.1 Contributions

In this thesis work, we addressed the problem of automated inference of emotional states - the group of prototypic emotions such as anger, disgust, fear, joy, sadness, and surprise - from the eye region signals or eye expressions. This is a challenging endeavour because (1) of the limited amount of information the eye region provides as opposed to the entire face, (2) majority of the work in this context in the literature has addressed automated analysis of the entire face, (3) facial analysis still remains an open machine-vision problem, and (4) real-world applications require robust and very fast processing systems. These challenges and the application motivations for this work were introduced in Chapter 1. In Chapter 2, a review of the prior art in the field of automated facial expression recognition was provided. Chapter 3 described the technical details of the two feature extraction algorithms: local binary patterns (LBP) and local color vector binary patterns (LCVBP) that form the basis of two
pipelines described within the context of this work. The LBP feature-based system was proposed, described, and evaluated in Chapter 4, whereas Chapter 5 introduced and evaluated the LCVBP feature-based system.

We proposed two frameworks for building an emotion recognition system operating on eye region images. The first framework was based on the traditional and more mature LBP feature descriptor applied to grayscale images. This advances LBP operator’s applicability for expression recognition from face images to eye region images. In the second framework, we emphasized the use of a more sophisticated and rich feature descriptor, LCVBP that extends the idea of LBP to include color information. This has the potential to inform future work in the field of color facial analysis for expression recognition. Each of these contributions are further discussed in this section.

6.1.1 Grayscale LBP feature-based system

The first contribution of this thesis is the automatic emotion recognition system based on the traditional LBP features to classify the set of basic emotions from grayscale eye region images. The LBP based approach has been proven in the past as an effective technique for facial expression recognition [92] due to its robustness to variations in illumination, and its high discriminative power due to different levels of locality. And from our experimental evaluation in Chapter 4, we were able to demonstrate its usefulness in expression recognition for eye region images as well. The proposed system consists of three main modules: Pre-Processing, Feature Extraction, and Classification. In the pre-processing stage, any present facial region in the image is first localized using a selected luminance-based skin-tone detector [10]. Next, the eye center locations are detected (both manually, and later on automatically using
the MPT EyeFinder [32]), and then the eye region within the localized facial region is normalized, aligned, and extracted.

The system was evaluated with different LBP parameters \( LBP_{8,1}, LBP_{8,2}, \) and \( LBP_{16,2} \) and two different classifiers: nearest neighbor with chi-square distance metric, and SVMs with a linear kernel. Through our experiments on the widely used CK+ facial expression dataset, we were able to achieve excellent person-dependent recognition rates (\approx 99.5\%) with LBP features and SVM classification. Furthermore, with the integration of automatic eye detection in the pre-processing stage, the results showed that the system performance does not decrease drastically (with a slight error in locating eye centers precisely) proving the effectiveness of the system in less constrained environments such as the multimedia browsing scenario, where pixel-level eye center detection accuracy might not be achievable. For person-independent classification tests on CK+ dataset using 10-fold cross-validation, our proposed system was able to achieve an accuracy of around 70%.

In addition, we performed a third set of experiments to evaluate recognition accuracy on unseen data taken under completely different environmental conditions by training the system on images from CK+ dataset and testing on MMI dataset. We compared the cross-database generalization performance of our proposed system (best RA of 67.4\%) with other facial expression recognition systems (Table 4.13) and observed consistent findings with regards to the generalization ability to other datasets. We also provided an evaluation and comparison of our eye region analysis results with state-of-the-art facial expression recognition systems on the criteria of Accuracy, Generalization, and Speed.
6.1 Contributions

6.1.2 Color LBP (LCVBP) feature-based system

Most of the prior work to-date has been carried out on single channel or grayscale images for facial expression recognition. The emotion recognition system from color eye region images based on LCVBP features is a novel framework in such context, and the second major contribution of this thesis. LCVBP, a recent and an effective feature descriptor for face recognition purposes, extends the idea of grayscale LBP features to include color information extracting two sets of features from color images: (1) color norm patterns (CNP) related to the norm of the multiple channels, and (2) color angular patterns (CAP) related to the spatial interactions between different color channels of the image. In the proposed LCVBP based system, the eye region is first manually localized, normalized and cropped. Since the main objective of the proposed LCVBP-based pipeline was to investigate a color based approach for eye expression recognition and in turn, evaluate the feasibility of a LCVBP as a suitable feature descriptor for the same, therefore, instead of focusing on integrating or building an automatic eye detection module, we manually label the eye center positions as the ground truth to bypass any errors introduced by the pre-processing stage. Then, the eye region image (RGB) undergoes color space conversion and is represented in an alternative three-channel color domain, ‘RQCr’ (R from RGB, Q from YIQ, Cr from YCbCr). The RQCr color space representation is shown to achieve the best performance across a number of different color-component configurations for the LCVBP feature-based FR system [12, 67].

The success of LCVBP as a suitable feature descriptor for emotion recognition on eye region images was demonstrated by performing 4-class recognition on a selected subset (144 images: 42 anger, 27 disgust, 30 happiness, and 45 surprise) of the MMI facial expression database. We were able to achieve an accuracy of 86.3% using
Future Directions

The ultimate goal of this research is the development of an automatic emotion recognition system using eye region signals that can be applied towards building socially intelligent applications. This objective motivates several exciting areas for future work described as follows.

6.2.1 Boosting the Recognition and Accuracy

The problem of emotion recognition from eye region signals belongs to the category of visual pattern recognition systems. Therefore, there are several parameters and machine learning methods that can be investigated to boost the accuracy and robustness of the proposed LBP-based systems.
6.2 Future Directions

- Several works have pointed towards integrating feature selection after the feature extraction process and before classification in order to select the most discriminative features. More specifically, tools such as Boosting (e.g., Adaboost, Conditional Mutual Information based Boosting) and LDA can be applied to the LBP histogram(s) in order to select a subset of features that contain the most discriminative information, and subsequently increased performance [67, 93].

- For classification, different SVM kernels (RBF, Polynomial) should be investigated, along with the optimal kernels parameters, in order to find the best performing SVM configuration.

- As mentioned before, LCVBP is a sophisticated tool that outputs a rich set of multiple features (CAP, CNP). In this work, these features are fused together prior to classification at the feature level to provide maximum benefit based on the literature. As there are several options to combine multiple features (pre-classification: sensor or feature level, post-classification: dynamic classifier selection, fusion at decision, rank or matching score level), feature fusion is an interesting yet inherently challenging problem that makes for an exciting area for future work in this context.

6.2.2 Including Temporal Information from Video

The proposed systems currently recognize emotions from static images. A natural extension to these systems is to include temporal information from image sequences in order to increase recognition accuracy as the suggested works do using dynamic LBP features [113]. Using multiple images from an expression sequence (Neutral→Apex) will not only allow for more training data, but can also provide for additional discriminative information regarding each expression class. When including temporal
information, another extension to the system would be the analysis of eye movements or gaze to gain better information about the mental states. This would increase the ability of the system to go beyond a basic set of emotional states as described below.

### 6.2.3 Beyond the basic emotions

This thesis describes an eye-region based emotion recognition framework and a set of basic emotions - Anger, Disgust, Fear, Joy, Sadness, and Surprise - are classified to present this framework. While the set of basic emotions has received great attention in research, there exist many mental states that the face, and in turn, the eye region is able to communicate such as boredom, stress, confusion, interested, and so forth. Depending on the application context, and the availability of data in such context, the proposed systems can be extended to include emotional/mental states that are beyond the basic set of emotions.

### 6.2.4 Application: Implicit Tagging of Multimedia Content

The growth and popularity of social media platforms (Facebook, Twitter, Flickr, Youtube, and many other web 2.0 services) in the recent years has caused a major paradigm shift in the way users consume digital media; passive consumption during the early years to active participation in creation, sharing, and diffusion of social data. These websites provide the ability for the users to add keywords or tags to the data, which are then indexed for retrieval and interaction purposes. Although the human-centered approach to tagging is a step forward, studies analyzing tagging behavior of users reveal that users are neither motivated nor aim at making retrieval systems work [104]. Also, the existence of huge amount of multimedia content on the Internet makes it impossible for explicit tagging to be the only solution.
In contrast to explicit tagging, implicit tagging involves generating tags for multimedia data such as images, videos, web pages, and advertisements based on user’s nonverbal reactions, such as facial expressions and head gestures. Indeed, psychology studies suggest that people behave in a similar manner with machines as they do with humans [80, 104] i.e; they display their reactions (smile, head shake, frown) in front of the device while interacting with multimedia data as they would in a conversation with another human being. Several types of tags can be generated based on the cues conveyed by nonverbal behavior such as emotional cues, level of interest, and focus of attention. The following are some scenarios where implicit tagging can be used:

- **Information about the content of the new data.** The user behavior might provide information about the content of the data. For example, in images or videos, laughter and disgust facial expressions can be used to tag the content as comedy and horror respectively. Similarly, in marketing, the user behavior might provide to be useful in assessing user reactions to new advertisements and products. Some recent works have been proposed to use physiological or EEG signals to tag multimedia data [94].

- **Assessment of the existing tags.** By analyzing the user behavior upon retrieval of existing data might, we might be able to assess the “correctness” of the existing tags. Jiao and Pantic [57] recently conducted a study to show that user’s facial expressions can be used to convey some information about the correctness of image tags.

- **Delivering personalized content.** Behavior-based or emotional tags might play an important role in delivering personalized content to the user. For example, a user feeling sad may want to watch video clips tagged as funny or happy. Arapakis *et al.* [2] investigated the role of emotions and facial expressions in
the information seeking process concluding that users emotion feedback can be used as good predictors of document relevancy. Also, a system was designed in [109] to acquire users preference of TV shows from users’ temporal patterns in facial changes.

Implicit human-centered tagging of multimedia content is a relatively new area of research under the umbrella of social signal processing. As seen by the aforementioned recent research in this area, facial and eye expressions are effective cues that can be processed to generate relevant tags to annotate multimedia data. Automatic emotion recognition from eye region signals proposed in this work can be applied in multimedia browsing environments at everyday places (e.g., home, office) as it offers real-time processing due to a smaller input signal, and robustness where precise eye center localization might not be feasible. It is important to mention that the proposed system might not be able to function solely by itself in an implicit tagging scenario, i.e., generate completely accurate implicit tags for multimedia data due to limitations such as all users might not fully express their feelings and the users might react differently for different data. Nevertheless, this type of system offers the potential to be used as an assistive technology to augment the tagging process by suggesting the closest matching emotion tag.
The first step of any visual pattern recognition problem is to be able to accurately identify and localize the region of interest; in this case, existence of eyes in an image under constrained and/or unconstrained environments. The appearance of eye regions can change with variation in viewing angles, illumination conditions, occlusion of the eye by the eyelids, reflectivity, head pose, and degree of openness of the eyes. In the field of facial expression analysis to deduce emotional states, eye region has been mostly analyzed by researchers as a part of the facial feature extraction.

The eye region can be roughly described by the intensity distribution of the pupil, iris, cornea, and by their shapes. Previously in literature, common approaches for measuring eye positions determine the location of the pupil or the iris center. An extensive survey on state-of-the-art in eye detection techniques can be found here [46].
A.1 Shape Models

Shape-based or geometric approaches are constructed from either the local point features or the contours of the eye region such as edges, eye corners, eyelids, eyebrows, the iris and the pupil using a prior model. The geometric models contain parameters for similarity and nonrigid transformations, and generally have the ability to handle variations in shape, scale, and rotation.

Simple Elliptical Shape Models.

A rough and a simple shape approximation of the iris and the pupil is an ellipse depending on the viewing angle, and therefore can be modeled by five parameters. Many researchers have proposed eye detection approaches based on this geometric property.

Daugman [17] proposes a pupil and iris detection technique that uses optimization of the curve integral of gradient magnitudes to fit an elliptical shape model. This model is limited by the fact that it does not take contour neighborhood into account, thereby, disregarding useful information. Hansen et al. [47] also use an ellipse model to define the shape of the iris, and is locally fitted to an image through an EM and RANSAC optimization approach. Their technique overcomes the limitation of [17], by incorporating neighboring information into the contour likelihood model, and also avoids explicit feature detection (strongest gray-level gradient and thresholds).

The use of simple elliptical models can prove to be an efficient approach to model features such as iris and pupil under different viewing angles. However, they require explicit thresholds and high-contrast images, and fail to successfully capture the variations in other eye features such as eyelids, eye corners and eyebrows.
Complex Shape Models.

Complex shape models, by definition, can account for more detailed specification of the eye shape. One such example is of deformable templates originally proposed by Yuille et al. [111]. A deformable eye template is specified by a circle, two intersecting parabolic curves and the two whitest points which are geometrically arranged to resemble the shape of the eye. The parameters for the curves are then estimated by minimizing the energy functions defined for the valleys, edges, peaks, and the intensity of the image and the template. Many methods have made use of this approach to build more generic and efficient models to avoid the local minima and improve the convergence [19, 108].

Although deformable template-based models are generically accurate and seem logical for eye-feature extraction, descriptions and tracking, they do come with their own set of limitations: (1) computationally expensive, (2) require an initialization point close to the eye for accurate localization, and (3) do not handle pose variations and eye occlusions well.

A.2 Pupil and Iris Detection

The pupil and the iris are fairly reliable and common features for eye detection, especially when the eye is viewed sufficiently closely. These features are generally darker than their surrounding area - the sclera. Kothari and Mitchell [63] propose a method to detect eyes in monoscopic gray scale images based on gradient direction field around the iris. The gradient field is extrapolated in a direction opposite to the gradient to calculate the largest number of intersection points determining the possible eye locations. Then, heuristics and postprocessing is performed to remove false eye candidates. This approach is limited to work under normal lighting conditions (due
to the requirement of high contrast between the iris and its surrounding area), and can still produce high number of false eye candidates.

Yang et. al [110] introduce an iterative threshold algorithm that looks for two dark regions (pupils) that satisfy anthropometric constraints using a skin-color model. Their approach is constrained by the accuracy of the skin-color model and the presence of other dark regions such as eyebrows and shadows.

### A.3 Wavelet Features with Boosting

#### Haar-like features with AdaBoost

Viola and Jones use the concept of very simple vertically and oriented binary rectangular templates for feature extraction applied mainly to the task of face detection [105]. This is carried out at all possible scales at all possible locations within the detection window. This was extended by Leinhart et al. [69] to include rotated features (45°). Figure A.1 shows the feature prototypes that are used under this framework; the light areas indicating positive weight and dark areas indicating negative weight. The total number of possible features (given different scales and locations) is quite large, thereby motivating the use of a boosting algorithm (e.g., Adaboost) for feature selection.

The Viola and Jones face detector [105] is considered by many to be the current state-of-the-art in fast face detection. This system learns the most discriminative Haar features through AdaBoost [38], a boosting algorithm used to radically reduce the number of features used in classification.

#### Haar-like features with GentleBoost

Haar-like features have also been used by Fasel et al. [33] for face, eye, and blink
A.3 Wavelet Features with Boosting

Figure A.1: Feature prototypes of Haar-like and center-surround templates. Light areas indicate positive weight and dark areas indicate negative weight.

detection under a generative model based on bayesian inferences. Unlike the binary classifier approach used in AdaBoost, the authors developed and trained a likelihood-ratio model using the GentleBoost method. To the original set of Haar-like features proposed by Viola and Jones [105], center-surround type wavelets and mirror image wavelets were added (See figure A.2), and GentleBoost was applied to sequentially choose and combine wavelets to minimizing a chi-square error function. Using this generative model, they propose a real-time context-specific two-stage face and eye detection system, which is made available under the framework of Machine Perception Toolbox [32].

Recently, González-Ortega et al. [42] has proposed a facial feature detection framework in which the facial region is first localized using skin-color modeling in TSL color space, and then rotated Haar-like features with Gentle AdaBoost are used for eye de-
A.3 Wavelet Features with Boosting

Figure A.2: Haar-like and mirror image wavelets used by the MPT face and eye detection system [33].

detection. They report a true positive rate (TPR) of 98.36% with a false positive rate (FPR) of .032% in the case where the eye search is constrained to the upper facial region and an FPR of .262% in an unconstrained search.

Gabor features with GentleBoost

Vukadinovic et al. [106] propose a fully automated facial point detection system using Gabor wavelets and the GentleBoost algorithm. The detected face region is divided into smaller regions of interest (ROIs) corresponding to each facial point. Then, feature vector for each facial point is extracted and compared with GentleBoost templates built from both gray level intensities and Gabor wavelet features. They report a recognition rate of 93% for 20 facial feature points on 300 Cohn-Kanade database samples.
Appendix B

Skin-tone Detection

Skin-tone detection plays a crucial role in wide array of image processing and computer vision applications ranging from face detection, face tracking, face recognition, gesture recognition, content-based retrieval (CBIR) systems, and in various surveillance type and human-computer interaction (HCI) applications. Skin-tone, due to its invariance to rotation and scale in various images [55], generally acts as a preprocessing tool or a cue in early stages of these higher-level systems. Section B.4.1 provides an overview of how skin-tone can be used as a useful cue in automatic face detection systems. Skin-color in an image is sensitive to factors such as varying illumination, camera adjustments, human skin types of different ethnicity, and other external appearance factors (hairstyle, glasses, motion, and background colors), thereby making it a challenging problem in image processing applications. The following comments discuss some of the challenges that impact the performance of any skin-tone detection algorithm.

Illumination

The light distribution in an image greatly affects the skin-color distribution. The po-
B.1 Approaches

Position of the light source (ambient, directed, shadow), and the type of the light source (incandescent, fluorescent) produce different skin-color characteristics; therefore the environment conditions under which an image is taken such as indoors or outdoors, play a major part in effectively detecting skin-tone.

Camera Parameters

A wide variety of cameras perform adjustments such as white-balancing which results in a biased global and/or local color distribution shift. The color reproduced by the CCD camera greatly depends on the spectral reflectance, and the camera sensor’s dynamic range.

Ethnicity

Depending on the person’s ethnic profile, the skin-color varies across different regions. For example, the skin-color belonging to Asian, African, Caucasian, and Hispanic categories varies from white, yellow, brown to dark.

B.1 Approaches

Extensive research has been carried out in the area of skin-color detection, implementing parametric or non-parametric (statistical) global skin color models defined in various color spaces. In these models, skin pixels are classified independently based on their chrominance (defined by the coordinates in the color space). The skin color signals generally tend to occupy a certain distribution in various color spaces. Therefore, in order to decrease the overlap between skin and non-skin pixels, a color space transformation is performed. Skin-tone detection algorithms are then applied in the new color space providing parameters for robust skin-pixel classification.

In literature, skin-tone detection approaches have been categorized into the fol-
following main categories: *explicitly defined skin region*, *nonparametric skin distribution*, *parametric skin distribution*, and *dynamic skin distribution* models applied in several color spaces (HSV, HSI, RGB, NCC-rgb, YCbCr, YIQ, YES, CIE-XYZ, CIE L*u*v).

A comprehensive survey of skin-tone detection techniques can be found here [60], [96]. I have selected to implement five skin-tone detection algorithms covering explicitly defined skin region and nonparametric categories. These algorithms are evaluated on a video dataset that was acquired indoors under varying illumination conditions across three subjects having different skin complexions. The dataset is described in more detail in Section B.3. The following section describes each type of skin-tone detection approach in detail, along with a brief overview about the various color spaces under which these algorithms are designed to operate.

### B.1.1 YCbCr Model

In the YCbCr model, RGB space is linearly transformed into Y ("luminance") and two color difference color components B-Y and R-Y. This color space is motivated by the fact that human visual system has considerably less spatial acuity for color information than for brightness and is mainly used for reduced data transmission for television and video. YCbCr is used for digital video, while YPbPr is the analog version used for analog component video. RGB to YCbCr color space conversion is done using:

\[
\begin{align*}
Y &= 16 + \frac{1}{256} \times (65.738R + 129.057G + 25.064B) \\
Cb &= 128 + \frac{1}{256} \times (-37.945R - 74.494G + 112.439B) \\
Cr &= 128 + \frac{1}{256} \times (112.439R - 94.154G - 18.285B)
\end{align*}
\]
where R, G, B, Y take the typical values from 0-255 (8-bit precision) and Cb, Cr components take the range (16-240).

The facial region segmentation algorithm proposed in [7] defines the skin-region distribution in the CbCr (“chrominance”) plane, discarding the Y (“luminance”) channel as per the following equation:

\[ 77 \leq Cb \leq 127, \quad 133 \leq Cr \leq 173 \] (B.2)

**B.1.2 Luminance-based Model**

The luminance-based skin-tone detection model proposed in [10] uses error signals derived from differentiating the “luminance” map and the non-red encoded “luminance” map. Using a dataset of 147,852 skin labeled pixels, a Gaussian curve fit is determined on the error signal through the expectation maximization (EM) method.

\[
\begin{align*}
    e(x) &= Y(x) - \arg\max(G(x), B(x)) \\
    f_{\text{skin}}(x) &= 1, \text{ if } 0.0251 \leq e(x) \leq 0.1177 \\
    Y(x) &= 0.2989R + 0.5870G + 0.1402B
\end{align*}
\] (B.3)

**B.1.3 NCC rgb Model**

NCC rgb or the normalized rgb model, a non-linear transformation of the RGB space, has been a popular choice amongst researchers for modeling skin-distribution because it removes the intensity information from the image by normalizing the color elements (R,G,B) of the linear RGB:
\[ r = \frac{R}{(R + G + B)}, \quad g = \frac{G}{(R + G + B)}, \quad b = \frac{B}{(R + G + B)} \quad \text{(B.4)} \]

A set of rules for skin pixel classification was derived in the NCC rgb color space using a constructive induction algorithm \[41\]. The image dataset included more than 2000 skin and non-skin subjects with different skin tones with several cameras indoors and outdoors under light conditions with CCTs ranging from 3200 to 5500K. The best performance was reportedly given by the following rule:

\[ \frac{r}{g} > 1.185, \quad \frac{rb}{(r + g + b)^2} > 0.107, \quad \frac{rg}{(r + g + b)^2} > 0.112 \quad \text{(B.5)} \]

### B.1.4 HSV Model

HSV (Hue-Saturation-Value) family of color spaces (HSI, HSV, HSB, HSL) represent the human perception of color, saturation, and luminance, and are primarily used in computer graphics. HSV has been previously studied by researchers for building skin-pixel classifiers \[49\]. RGB to HSV is also a non-linear transformation, and can be computed using:

\[
H_1 = \arccos \left( \frac{\frac{1}{2}[(R - G) + (R - B)]}{\sqrt{(R - G)^2 + (R - B)(G - B)}} \right)
\]

\[
H = \begin{cases} 
H_1, & \text{if } B \leq G \\
360 - H_1, & \text{if } B > G
\end{cases}
\quad \text{(B.6)}
\]

\[
S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)}
\]

\[
V = \frac{\max(R, G, B)}{255}
\]

The shape of skin color subspace in HSV color space was estimated in \[39\] by defining six bounding planes:
B.1 Approaches

\[ V \geq 40 \]
\[ H \leq (-0.4V + 75) \]
\[ S = \frac{\max(R, G, B) - \min(R, G, B)}{\max(R, G, B)} \]  \hspace{1cm} (B.7)
\[ V = \frac{\max(R, G, B)}{255} \]

B.1.5 Non-Parametric Model

In this method, the color space is quantized into a number of 2D or 3D histogram bins, where each bin stores the count associated with the occurrence of the bin color in the training data set. With the availability of large amounts of labeled skin and non-skin pixels, this approach can be used to build a robust and fast skin-pixel classifier. A 3D RGB histogram model was built in [59] examining 13,640 images from the web with over a billion labeled skin/non-skin pixels.

\[ P(c/skin) = \frac{s(c)}{T_s}, P(c/non - skin) = \frac{n(c)}{T_n} \]  \hspace{1cm} (B.8)

where, \( s(c) \) - pixel count in the color \( c \)-bin of the skin histogram, \( n(c) \) - pixel count in the color \( c \)-bin of the non-skin histogram, \( T_s \), \( T_n \) - total counts in the skin and non-skin histogram bins. After training, a Naïve Bayes classifier approach is used to generate the skin map using:

\[ \frac{P(c/skin)}{P(c/non - skin)} \geq \theta \]  \hspace{1cm} (B.9)

For this project, I used the images from FERET database [87] as the training set. Using the available ground truth, over 8.75 million labeled skin pixels from a rectangular region around the nose, and about 180 million labeled non-skin pixels.
were acquired from the web to build a 3D RGB histogram using 323 bins. The threshold, $\theta$, was determined to be 0.2 from experimentation over the test video set.

B.2 Illumination Compensation

As previously mentioned, radically changing lighting conditions greatly affect the skin-color distribution in an image and may render any skin-tone detector unreliable. Therefore, in order to provide robustness against such illumination variation, spectral content of an image or the skin pixels can be adjusted using the color constancy approach as a pre-processing step to the skin-tone detection module. Color constancy approaches work on the assumption of estimating the illuminant source or the underlying distribution of the color values and accordingly adjust the pixel wise values of the image. For the purposes of this project, the grayworld algorithm was implemented as a pre-processing step to the skin-detection, which is described below.

B.2.1 Gray World Assumption (GW)

Gray world algorithms assume given an image with sufficiently varied colors, the average reflectance of the image is gray, and they try to adjust the mean values of each color channel towards a common value. A simple method to apply the Gray World Assumption is to find the average values of the image’s R, G, and B color components and use their average to determine an overall gray value for the image. Each color component is then scaled according to the amount of its deviation from this gray value. The scale factors are obtained by simply dividing the gray value by the appropriate average of each color component. Thus, if an image under normal white lighting satisfies the gray world assumption, putting it under a color filtered lighting would disrupt this behavior. By forcing the gray world assumption on the
image again, we are in essence, removing the colored lighting to reacquire the true colors of the original. The use of grayworld pre-processing before applying skin-tone detection greatly reduced the number of false positive detections, and in some cases, improved the skin detection rates. An example of skin detection with and without grayworld pre-processing is shown in Figure B.1.

Figure B.1: An example exhibiting the effect of grayworld pre-processing. (a) Original Image, (b) Skin map without the grayworld pre-processing (High number of false positives), (c) Output image after grayworld, (d) Skin map with the grayworld pre-processing (Low number of false positives).
B.3 Experiments

The goal of the experiments is to compare in a detailed quantitative way each skin model to determine which one detects better human skin color while minimizing false positives. For generating ground truth and systematic evaluation of the skin-color models, a database was created using a Panasonic Lumix point-and-shoot digital camera and a tripod. 15 short videos at a resolution of 1280x720 at 30fps were taken in controlled indoor environment. Sample images from the database are presented in Figure B.2. After the capture, images were extracted from the videos, and skin pixels inside the faces were labeled as ground truth by fitting an ellipse around the face followed by manual correction (with some precision error because there are some parts which are difficult to classify as skin or non-skin). Below are some notes summarizing the details of the database.

- The illumination conditions represented in the database cover the following scenarios:
  - Fluorescent Ambient Lighting (S1)
  - Incandescent Ambient Lighting (S2)
  - Incandescent Frontal Lighting (S3)
  - Incandescent Profile Lighting (S4)
  - Incandescent Directed Lighting (S5)
  - Low Lighting (S6)
  - Ultra Low Lighting (S7)

- Database contains images of three different participants belonging to different ethnic backgrounds - Caucasian, East Indian, and African - covering different
B.3 Experiments

Skin types - light, medium, and dark skin respectively.

- The videos involve the subject walking towards a fixed camera mainly with a frontal pose to allow for accurate ground truth capture.

Figure B.2: Examples of video frames from the test dataset. Columns 1-3 exhibit subjects for light, medium, and dark skin types respectively. Rows 1-3 exhibit ambient fluorescent, ambient incandescent and directed incandescent lighting conditions respectively.

The experiments involve studying in depth five different skin-tone detection models on the above mentioned database consisting of 12 different sets of images (3 subjects, 4 lighting scenarios (S1-S4)). The classification results of each skin color model are compared against the ground truth at pixel level (Table B.1). Also, the false positives or the non-skin pixels falsely identified as skin pixels are measured (Table B.2).
### B.4 Results and Recommendations

Comparing the results of all the skin-tone models, the luminance-based skin-tone detection model [10] and the YCbCr model [7], combined with the grayworld pre-processing, outperform the rest of the skin-tone detection models analyzed. Figure B.3 below provides a comparison of true positive detection rates for one such scenario - Incandescent Frontal (S3) on Subject B. Specifically, the luminance-based model provides the best overall skin detection rates, with the most hits on skin pixels and the least false positives. This model proves to be fairly robust against varying illumin-
### Table B.2: False positive detection rates (%) for each skin model under various illumination scenarios

<table>
<thead>
<tr>
<th>Illumination</th>
<th>Subject</th>
<th>Lum</th>
<th>YCbCr</th>
<th>HSV</th>
<th>NCC-rgb</th>
<th>NonParam</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fluorescent</td>
<td>A</td>
<td>0.4</td>
<td>32.5</td>
<td>30.7</td>
<td>82.2</td>
<td>59.4</td>
</tr>
<tr>
<td>Ambient (S1)</td>
<td>B</td>
<td>9.0</td>
<td>66.9</td>
<td>56.2</td>
<td>84.3</td>
<td>77.1</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>34.1</td>
<td>76.0</td>
<td>72.6</td>
<td>75.5</td>
<td>91.2</td>
</tr>
<tr>
<td>Incandescent</td>
<td>A</td>
<td>14.4</td>
<td>11.6</td>
<td>34.0</td>
<td>43.9</td>
<td>67.2</td>
</tr>
<tr>
<td>Ambient (S2)</td>
<td>B</td>
<td>18.0</td>
<td>5.0</td>
<td>48.8</td>
<td>59.3</td>
<td>74.5</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>23.4</td>
<td>6.2</td>
<td>59.1</td>
<td>63.6</td>
<td>85.1</td>
</tr>
<tr>
<td>Incandescent</td>
<td>A</td>
<td>0.1</td>
<td>1.9</td>
<td>45.8</td>
<td>62.3</td>
<td>65.3</td>
</tr>
<tr>
<td>Frontal (S3)</td>
<td>B</td>
<td>2.8</td>
<td>14.3</td>
<td>56.7</td>
<td>82.6</td>
<td>80.0</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>4.4</td>
<td>41.7</td>
<td>96.2</td>
<td>95.9</td>
<td>98.8</td>
</tr>
<tr>
<td>Incandescent</td>
<td>A</td>
<td>18.9</td>
<td>14.5</td>
<td>23.9</td>
<td>25.9</td>
<td>80.7</td>
</tr>
<tr>
<td>Profile (S4)</td>
<td>B</td>
<td>2.2</td>
<td>16.5</td>
<td>45.2</td>
<td>52.9</td>
<td>72.6</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>0.2</td>
<td>13.6</td>
<td>81.9</td>
<td>85.4</td>
<td>92.1</td>
</tr>
</tbody>
</table>

On the other hand, the previously employed non-parametric model in the current application, trained on the FERET database, fails to perform robustly giving us the least true positive skin detection hits. The limited variation in the lighting conditions and the skin samples of the FERET database contribute to the poor performance of the non-parametric model.

The proposed skin-tone detection model has proven to be fairly robust in detecting skin content in an image. However, there are certain observed cases in which the model falsely classifies any bright patches of “red” present in the image as skin. Also, under
B.4 Results and Recommendations

Figure B.3: Comparison of true positive detection rates for one scenario - Incandescent Frontal Illumination for Subject B.

certain lighting conditions, when the pixel values approach saturation, the skin-tone detector fails by classifying such areas as skin as seen in Figure B.4. The robustness in these conditions can be improved by adding rules in which any bright “red” and saturated “white” patches can be avoided as being classified as belonging to the skin category.

B.4.1 Skin-tone as a Cue in Face Detection

There are many cues such as shape of the face, skin-tone, facial features and motion (in videos) that are used in combination for face detection. Skin-tone is a useful cue for human detection and specifically face detection, in many surveillance-type and HCI applications. By itself, skin-tone detection is inadequate to detect faces with a low positive false rate, however it can provide supplemental information that can be
Figure B.4: Example of non-skin pixels falsely classified as skin by the proposed skin model. (a) Original Image, (b) Skin map exhibiting non-skin pixels (“red” patches and saturated pixels) being falsely classified as skin.

used to increase the robustness of a face detection module. However, in order to get a sense of usefulness of skin-tone in automatic face detection, Figure B.5 depicts the general framework for using skin-tone as a standalone cue for face detection.

First, the image is color corrected for any non-uniform color characteristics introduced by the camera or the illumination source. The next step is to detect skin regions by first converting the image into the appropriate color space, and then applying skin-tone detection. Along with the skin regions, there are also various noise pixels or false positives belonging to the background objects that are classified as skin in many cases. This noise is removed by passing the image through a 5x5 or a 7x7 median filter. Morphological operations such as dilation and erosion are performed on the noise-removed image to fill any holes present due to shadows, reflectance. The final step is to detect any faces present in the image using the geometric properties of the face, and fitting an ellipse that satisfies the aspect ratio of the human face mentioned in the open literature (height/width $\simeq 1.5$) [66].
One of the most powerful and popular technique for face detection is the Viola-Jones face detector based on Adaboost [105]. This technique works on grayscale images and requires training data to learn a face’s appearance using Haar-like features. Figure B.6 below demonstrates how skin-tone can be used to reduce the false positive and increase the true positive rates.

In the first image, the Viola-Jones face detector is able to correctly detect the face present in the image at the expense of multiple false positives. Using the skin-tone
Figure B.6: Appearance-based versus Skin-tone Face Detection. Viola-Jones face detection results (a), (c) and Skin-tone detection face detection results (b), (d)

Using the skin-tone detector, we can remove any false positives. In the second image, the faces present in the image are missed by the Viola-Jones detector and in addition, new false positives are introduced. Using the skin-tone detector, the face regions blobs can be estimated thereby increasing the robustness of the face detector module.
References


[37] E. Fox and L. Damjanovic. The eyes are sufficient to produce a threat superiority effect. Emotion, 6(3):534–539, 2006. 4


[58] C. Jones and A.L. Abbott. Color face recognition by hypercomplex gabor analysis. In Automatic Face and Gesture Recognition, 7th International Conference on, April 2006. 31, 71


REFERENCES


