Machine Analysis of Facial Behaviour for Affective Computing

Yifan Xia

The thesis is submitted in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy of the University of Portsmouth.

April 2021
Abstract

With the popularity of smart devices in our daily lives in recent years, affective computing has attracted increasing attention, which is regarded as the fundamental requirement of human-machine interaction systems, such as smart phones and virtual reality equipment. As the most communicative part of emotion in our body, face contains a lot of behaviours of expressing emotions. Facial behaviour analysis imitates the way humans analyse and understand emotions, which is essential for achieving affective computing. However, automatic facial behaviour analysis is still a very difficult task, especially in the wild environment. This thesis addresses facial behaviour analysis from two fundamental aspects, namely eye analysis and facial expression analysis. Therefore, the proposed work in this thesis deeply explores how to adapt deep learning technologies to address the problems and challenges mainly from these two aspects.

For eye analysis, eye centre localization that occupies the crucial position becomes the first priority to be addressed. Existing methods mainly rely on hand-crafted features, which are not robust enough and are very sensitive to the variation from the wild environment. Moreover, the previous works on eye centre localization have rarely used technologies of deep learning. To address these issues, this thesis proposes a novel method based on a fully convolutional network (FCN) for the task of eye centre localization, which treats eye centre localization as a special subproblem of the task of semantic segmentation. The proposed method has been validated on challenging databases and has competitive performance compared with the state-of-the-art methods in terms of accuracy of eye centre localization, which is an alternative solution for some challenging real-world scenarios.

For facial expression analysis, this thesis proposes a novel relation-aware facial expression recognition method called Relation Convolutional Neural Network (ReCNN). ReCNN adaptively captures the relationship between crucial regions
and facial expressions and focuses on the most discriminative regions for recognition. Comparing with the previous methods that rely on processing the whole face for recognition, the performance of ReCNN is more accurate and robust on two large in-the-wild databases. What’s more, the relationship between crucial regions and facial expressions shows big potential on further improving the performance of facial expression recognition. Inspired by ReCNN, this thesis continues to explore the role of crucial facial regions in facial expression synthesis and proposes a novel method called Local and Global Perception Generative Adversarial Network (LGP-GAN) for facial expression synthesis. It fully utilizes local and global facial information during facial expression synthesis. Extensive experiments on the mainstream database demonstrate that LGP-GAN has superior performance compared with the state-of-the-art methods, which is a feasible solution for the issue of inadequate training data in facial expression recognition. Towards exploring mobile affective computing, this thesis further proposes a light-weight CNN architecture with high performance and low consumption, which is well-suited for mobile applications. The proposed method is capable of real-time performance on an actual mobile device and allows for easy portability and integration with other applications.

In summary, through developing the aforementioned algorithmic solutions, the thesis gains the first-hand experience in adapting the technologies of deep learning to the facial behaviour analysis task. This is supposed to be beneficial to propagate facial behaviour analysis to a broader range of applications based on affective computing.
# Contents

Abstract

Declaraton

List of Figures

List of Tables

List of Acronyms

Acknowledgements

Publications

1 Introduction

1.1 Background

1.2 Problems and Challenges

1.2 Research Aim and Objectives

1.3 Contributions

1.4 Outline

2 Literature Review

2.1 Overview of Deep Learning

2.1.1 Basic Concepts

2.2 Application Areas

2.2.1 Smart Vehicles

2.2.2 Smart Healthcare

2.2.3 Smart Education

2.3 Challenges and Directions

2.3.1 Data Collection

2.3.2 Nonfrontal Face

2.3.3 Network Structure Design

2.4 Relation to Our Work

i

vii

viii

x

xii

xiv

xv

1

1

3

6

7

8

11

13

14

16

20

21

21

23

25

27

27

28

29

30
3 Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

3.1 Introduction

3.2 Related Work

3.2.1 Eye Centre Localization

3.2.2 Fully Convolutional Network

3.3 Methodology

3.3.1 Preprocessing

3.3.2 Network Architecture

3.3.3 From Heatmaps to Coordinates

3.4 Experimental Results

3.4.1 Database

3.4.2 Evaluation Criteria

3.4.3 Experimental Settings

3.4.4 Quantitative Results

3.4.5 Qualitative Results

3.4.6 Comparison with Existing Approaches

3.5 Summary

4 Relation-Aware Facial Expression Recognition

4.1 Introduction

4.2 Related Work

4.3 Methodology

4.3.1 Overview

4.3.2 Network Architecture

4.3.3 Loss Function

4.4 Experiment

4.4.1 Databases

4.4.2 Implementation Details

4.4.3 Experiment on AffectNet Database
Declaration

Whilst registered as a candidate for the above degree, I have not been registered for any other research award. The results and conclusions embodied in this thesis are the work of the named candidate and have not been submitted for any other academic award.
## List of Figures

1.1 The applications of facial expression analysis.......................... 2
2.1 Example of image data .......................................................... 14
2.2 Example of CNN architecture ................................................. 16
2.3 Examples of convolution operation ......................................... 17
2.4 Examples of different activation functions ............................... 18
2.5 Examples of Max pooling and Average pooling ......................... 18
2.6 Examples of fully connected layer ......................................... 19
2.7 A general structure of RNN ..................................................... 20
3.1 Overview of our method for eye centre localization using shallow fully convolutional network ........................................... 39
3.2 The sample heatmaps generated using Gaussian kernel ............ 41
3.3 Example output produced by our network ............................... 45
3.4 The generated heatmap using different values of standard deviation $\sigma$ ................................................................. 48
3.5 Experimental results on different values of standard deviation $\sigma$ on BioID database .......................................................... 49
3.6 Normalized error curves of the proposed approach on the BioID and GI4E database .......................................................... 50
3.7 Qualitative results of our approach on the BioID and GI4E database .......................................................... 51
4.1 The differences between facial expressions often appear in crucial regions ........................................................................ 57
4.2 Overview of the proposed ReCNN framework .......................... 58
4.3 The framework of the proposed ReCNN .................................... 63
4.4 Examples of facial action units ............................................... 65
4.5 The example of generating crucial regions ................................ 67
4.6 Examples of facial expression images from the databases .......... 73
4.7 The confusion matrix of the proposed method on the AffectNet database................................................................. 76
4.8 The confusion matrix of the proposed method on the RAF-DB database........................................................................... 77
4.9 Some examples of twelve-class compound facial expressions picked from RAF-DB .......................................................... 79
5.1 The overview of the LGP-GAN for facial expression synthesis........................................................................................... 89
5.2 Crucial region samples of eight facial expressions on RaFD database .................................................................................. 91
5.3 Qualitative results of facial expression synthesis on RaFD database....................................................................................... 100
5.4 Qualitative comparison of facial expression synthesis on RaFD database ............................................................................. 101
6.1 The architecture of our light-weight deep model for facial affective computing ................................................................. 109
6.2 Difference between fully connected layer and global average pooling layer......................................................................... 111
6.3 Examples of the eight categories of facial expressions .................................................................................................. 114
6.4 Facial expression classification confusion matrix of the proposed network architecture on AffectNet validation set.............. 117
6.5 Implementation of the proposed model on the iOS platform ............................................................................................. 122
List of Tables

3.1 The performance of our proposed approach on BioID and GI4E database................................................................. 49
3.2 Comparison of our method with other methods on BioID database.............................................................................. 53
3.3 Comparison of our method with other methods on GI4E database.............................................................................. 54
3.4 Comparison of our method with other methods in average processing time ................................................................. 54
4.1 List of AUs involved in six basic facial expressions ................. 66
4.2 The details of database for facial expression recognition including the expression categories, training and test samples ....... 73
4.3 Comparison of accuracies (%) using different methods on AffectNet ............................................................................. 76
4.4 Comparison of accuracies (%) using different methods on RAF-DB ........................................................................ 78
4.5 Comparison of accuracies (%) using different methods on compound facial expressions from RAF-DB ......................... 80
4.6 Comparison of accuracies (%) using different methods on eleven compound facial expressions from RAF-DB .......... 80
4.7 Evaluation of effect of different crucial regions in RECNN on AffectNet database............................................................ 82
4.8 Evaluation of effect of attention modules and weighted loss function in RECNN on AffectNet database......................... 82
5.1 Generator network architecture ......................................................... 94
5.2 Discriminator network architecture ................................................ 95
5.3 Comparison of our method with other methods on RaFD database using IS and FID........................................................ 102
6.1 Details of parameters in proposed network ........................................ 109
6.2 The numbers of each category of facial expressions in AffectNet
for training .................................................................................................. 114
6.3 Classification accuracy (%) of facial expressions on AffectNet
validation set ............................................................................................. 118
6.4 Comparison of our method with other methods in AffectNet
validation set for valence and arousal estimation ........................................ 119
6.5 Number of parameters and model size of different networks........ 123
6.6 Running speed and memory consumption of our implemented
mobile application ....................................................................................... 123
### List of Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAM</td>
<td>Active Appearance Model</td>
</tr>
<tr>
<td>ANN</td>
<td>Artificial Neural Network</td>
</tr>
<tr>
<td>AR</td>
<td>Augmented Reality</td>
</tr>
<tr>
<td>ASD</td>
<td>Autism Spectrum Disorder</td>
</tr>
<tr>
<td>ASM</td>
<td>Active Shape Model</td>
</tr>
<tr>
<td>AU</td>
<td>Action Unit</td>
</tr>
<tr>
<td>CC</td>
<td>Pearson’s Correlation Coefficient</td>
</tr>
<tr>
<td>CCC</td>
<td>Concordance Correlation Coefficient</td>
</tr>
<tr>
<td>CCE</td>
<td>Categorical Cross Entropy</td>
</tr>
<tr>
<td>CDA</td>
<td>Clustering-based Discriminant Analysis</td>
</tr>
<tr>
<td>CDAAE</td>
<td>Conditional Difference Adversarial Autoencoder</td>
</tr>
<tr>
<td>CDF</td>
<td>Cumulative Density Function</td>
</tr>
<tr>
<td>CGAN</td>
<td>Conditional GAN</td>
</tr>
<tr>
<td>CNN</td>
<td>Convolutional Neural Network</td>
</tr>
<tr>
<td>DECGAN</td>
<td>Double Encoder Conditional GAN</td>
</tr>
<tr>
<td>DFSTN</td>
<td>Deep Facial Spatiotemporal Network</td>
</tr>
<tr>
<td>DNN</td>
<td>Deep Neural Network</td>
</tr>
<tr>
<td>FACS</td>
<td>Facial Action Coding System</td>
</tr>
<tr>
<td>FCN</td>
<td>Fully Connected Network</td>
</tr>
<tr>
<td>FID</td>
<td>Fréchet Inception Distance</td>
</tr>
<tr>
<td>G2GAN</td>
<td>Geometry-Guided Generative Adversarial Network</td>
</tr>
<tr>
<td>GAN</td>
<td>Generative Adversarial Network</td>
</tr>
<tr>
<td>GAP</td>
<td>Global Average Pooling</td>
</tr>
<tr>
<td>GC-GAN</td>
<td>Geometry-Contrastive Adversarial Network</td>
</tr>
<tr>
<td>GPU</td>
<td>Graphics Processing Unit</td>
</tr>
<tr>
<td>GRU</td>
<td>Gated Recurrent Unit</td>
</tr>
<tr>
<td>HMMs</td>
<td>Hidden Markov Models</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>HoG</td>
<td>Histogram of Oriented Gradient</td>
</tr>
<tr>
<td>IN</td>
<td>Instance Normalisation</td>
</tr>
<tr>
<td>IS</td>
<td>Inception Score</td>
</tr>
<tr>
<td>LBP</td>
<td>Local Binary Pattern</td>
</tr>
<tr>
<td>LGP-GAN</td>
<td>Local and Global Perception GAN</td>
</tr>
<tr>
<td>LSTM</td>
<td>Long Short-Term Memory</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Squared Error</td>
</tr>
<tr>
<td>MTL</td>
<td>Multi-task Learning</td>
</tr>
<tr>
<td>NIR</td>
<td>Near-infrared</td>
</tr>
<tr>
<td>RaFD</td>
<td>The Radboud Faces Dataset</td>
</tr>
<tr>
<td>ReCNN</td>
<td>Relation Convolutional Neural Network</td>
</tr>
<tr>
<td>ReLU</td>
<td>Rectified Linear Unit</td>
</tr>
<tr>
<td>RMSE</td>
<td>Root Mean Square Error</td>
</tr>
<tr>
<td>RNN</td>
<td>Recurrent neural network</td>
</tr>
<tr>
<td>SAGR</td>
<td>Sign Agreement</td>
</tr>
<tr>
<td>SGD</td>
<td>Stochastic Gradient Descent</td>
</tr>
<tr>
<td>SIFT</td>
<td>Scale Invariant Feature Transform</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machines</td>
</tr>
<tr>
<td>SVR</td>
<td>Support Vector Regression</td>
</tr>
<tr>
<td>VR</td>
<td>Virtual Reality</td>
</tr>
<tr>
<td>VRF</td>
<td>Valid Receptive Field</td>
</tr>
</tbody>
</table>
Acknowledgements

Firstly, I am extremely grateful to my supervisor Prof. Hui Yu for providing me this wonderful PhD opportunity to be a member of the Visual Computing Group at the University of Portsmouth. His profound knowledge and rich experience have helped me a lot during my time as his student. All the academic achievements I have achieved are inseparable from his patient guidance and invaluable advice. I would especially like to thank all the wonderful conversations and discussions we had. My supervisor is always patient to listen to my thoughts and responded to me promptly. His endless energy and creative ideas helped me to stay motivated. This work could not have been possible without his constant support and guidance over the past three years. I am really lucky to meet a supervisor who cared so much about my academic research and daily life. His guidance and help, from research to life, are worth my lifelong gratitude. Also, I would like to thank Prof. Fei-Yue Wang and Prof. Junyu Dong for their kind support with their broad knowledge and insights. I would also like to thank Dr. Brett Stevens and Dr. Charles Nduka for their kind support and collaboration during my PhD study.

I would like to further thank other colleagues of the Visual Computing Group for their kind help and continuous support including Miss. Qiongdan Cao, Mr. Jianwen Lou, Mr. Martin Kearl, Mrs. Xiaoxu Cai, Dr. Xin Liu, Dr. Hao Fan, Dr. Jianyuan Sun, Dr. Yiming Wang, Mrs. Weihong Gao and Dr. Shu Zhang. We have many good moments together in Portsmouth.

Finally, I dedicate this thesis to my parents. I would like to express gratitude to my parents for their encouragement and support during my study. They are staying by my side all the way through this journey.
Publications


Chapter 1

Introduction

1.1 Background

In recent years, smart devices such as computers and mobile phones have quickly become indispensable to people's lives, which we spend a lot of time using and interacting with them every day. Currently, these smart devices are emotionally blind which means that they don’t consider human affective states when interacting. However, for humans, successful communication between people depends largely on understanding signals of affective state. Human-machine interaction without considering users’ affective states will cause most of the information in the interaction to be lost. Considering the importance of affective states in our daily lives, it is recognized that the ability to recognize and understand human affective states is beneficial for smart devices to achieve advanced intelligence and promote the progress of human-machine interaction.

Recently, the research topic of affective computing has been widely studied to address the issues of emotional lack of smart devices. The purpose of affective computing is to develop intelligent algorithms for smart devices to read and understand affective states mimicking the human way. Humans usually utilize multi-modal and non-verbal affective behaviour such as facial expressions and hand gestures to express themselves. The face serves as the most visible and crucial channel of nonverbal communication revealing emotions and communicates intent, which has attracted the attention of researchers from the field of psychology and affective computing. Therefore, machine understanding and recognition of affective states based on facial behaviours have gained increasing attention and interest from researchers.
Facial behaviour analysis has been implemented for affective computing in many emerging applications to facilitate human-machine interaction (See Figure 1.1). There are many fields such as healthcare, automotive industries, education, and entertainment that could benefit from facial behaviour analysis. The feasibility of facial behaviour analysis for affective computing has been demonstrated in many applications: driver assistant systems that monitor the driver’s state to improve safety; smart healthcare systems that can help diagnose disease and assess the pain of patients based on facial behaviours; and online learning systems that adjust the teaching process according to students' affective states and engagement.

In this work, we define facial behaviour analysis as consisting of two subtasks, namely eye analysis and facial expression analysis. The eye plays a vital role in the perception of the world. And it is important when evaluating things such as attraction and attentiveness and makes a significant contribution in the expression of emotion and social signal. In addition to the eye, facial expressions as one of the most vital affect signals for humans is a core module in facial behaviour analysis systems, which can reveal intent, display affection, express emotion during communication. Both modalities contain important emotional information, which helps humans to read and understand the affective states of others.

Since the ability of facial behaviour analysis largely determines the performance of most affective computing applications, the approaches for facial
behaviour analysis must work normally under challenging real-world scenarios. Moreover, for some real-time applications that must process and analyse large amounts of data, it is necessary to maintain a high level of computational efficiency. All these requirements make the implementation of facial behaviour analysis systems for affective computing applications a very challenging task. In recent years, the emerging technologies of deep learning have led to an explosive rise in performance for the tasks of various research fields such as pattern recognition and computer vision. Some state-of-the-art approaches using deep learning models achieve near or even super-human performance. It is thus interesting to extend deep learning technologies to facial behaviour analysis task, which motivates the thesis to utilize deep learning to solve problems and challenges in facial behaviour analysis for affective computing in the wild.

1.2 Problems and Challenges

Although the main focus in affective computing has shifted to analyse facial behaviours, existing methods cannot be used in fully unconstrained environmental conditions effectively. The problems and challenges preventing this are mainly from two subtasks, eye analysis and facial expression analysis. Each task has its own issues, which is elaborated as follows:

First, eye centre localization is a fundamental step for eye gaze analysis. It estimates the pupil centres of a given facial image. There have been many works in this area recently, but due to some challenges from illumination, viewing angles and occlusions, the accuracy needs to be further improved for some applications under real-world conditions. There are a number of tools and commercial systems utilizing specialized head-mounted devices for eye centre localization, but they are expensive and uncomfortable for users. Although, there exist a couple of systems available to localize eye centre based on the low-cost webcam, they have limited performance when moving from the constrained lab scenarios to the in-the-wild
situations. These methods generally rely on hand-crafted features extracted from the appearance and geometry information of the eye to localize the eye centre through technologies of image processing and machine learning. However, these hand-crafted features are not robust enough, which are very sensitive to the variation from the wild environment such as illumination and image noises. For the task of eye centre localization, it can be found that the deep learning-based approach has rarely been studied. As an alternative, the deep learning-based approach can directly extract features from the raw data. These highly discriminative features are more robust than the hand-crafted features under real-world scenarios. Considering deep learning technologies have been successfully applied to various tasks in the field of computer vision recently, it is very timely and of great potential to develop a robust deep learning-based eye centre localization method.

Second, facial expression analysis is the main task of facial behaviour analysis since for humans the most universal and powerful function for expressing their affective states is the facial expression. There have been many studies on facial expression recognition as it is an essential module for many human-machine interaction applications such as smart healthcare diagnose, driver monitoring systems and robots. In recent times, the research of facial expression recognition has been moving from the constrained lab scenarios to in-the-wild situations and has been widely studied utilizing deep learning technologies. However, there are still problems and challenges to be addressed, which can be summarized as follows:

- Much progress in facial expression recognition has been made using deep learning techniques, but existing methods mainly take the whole face as a uniform source of features for facial expression analysis. The performance may be affected by redundant facial image information. In fact, according to the physiology and psychology research, the differences between facial expressions often appear in crucial regions such as eye and mouth, which have close relationships with emotion expression. Moreover, studies have been proven that the attention of humans naturally focus on specific facial regions
rather than the whole face when they recognize and distinguish different facial expressions. Therefore, it can be inferred that there are certain relationships between these crucial regions and facial expressions can help recognize facial expressions. However, most state-of-the-art approaches still recognize facial expressions depending on the whole face and deploy few efforts in exploiting relations. As a result, it is necessary to consider and utilize the potential relations between crucial regions and facial expressions to recognize facial expressions.

- As the focus of research of facial expression recognition has been shifted to the challenging wild scenarios, many researchers have utilized deep learning technologies to address challenges from the wild environment such as large poses and occlusion as well as illumination and intensity variations. Given that deep learning is a data-driven technology that needs a great deal of data to train the network, the major challenge of deep learning-based facial expression recognition is the lack of sufficient training data. However, it is difficult to collect a large annotated facial expression database, which requires specific expertise and a long time to complete. A reasonable solution to address the problem of insufficient training data is facial expression synthesis. One of the most successful methods for data synthesis is Generative Adversarial Network (GAN), which has been used for a variety of tasks in the field of computer vision such as image-to-image translation, image synthesis and image inpainting. Since most existing GAN-based methods are designed for general synthesis tasks without considering the characteristics of facial expressions, it is obvious that they are not appropriate to synthesize facial expressions. Therefore, how to extend GAN to its unfamiliar facial expression synthesis task has been an interesting research topic and still remains largely unexplored.

- In recent years, mobile devices with cameras have become an important part of people’s daily lives, which has promising applicability in various areas such as entertainment, healthcare and education due to its popularity and portability. Many researchers have started to develop real-time systems on mobile devices
for ordinary users to recognize facial expressions improving the human-machine interaction experience. The performance of facial expression recognition has been reached state-of-the-art levels because of the technologies of deep learning, especially convolutional neural networks (CNNs). Although impressive results have been obtained, it has also led to a significant increase in the computational complexity of CNNs which means that high-performance hardware must be used. It is difficult for most existing CNN-based approaches to be deployed on mobile devices with limited storage space and calculation capacity. Thus, designing a light-weight CNN architecture that can be easily ported to mobile platforms for real-time applications to recognize facial expressions is an urgent need to be addressed.

1.2 Research Aim and Objectives

The aim of this thesis is to address the aforementioned problems and challenges in facial behaviour analysis and develop easy to use methods suitable for the wild environment using deep learning technologies. It aims to bridge the gap between existing state-of-the-art research and practical applications. This thesis consists of 4 objectives as follows.

1. Investigate methods to improve robustness of eye centre localization which utilize hand-crafted features.
2. Investigate the relationship between crucial regions and facial expressions for emotion analysis.
3. Consider the importance of crucial facial regions for facial expression synthesis.
4. Develop and implement a network architecture that can be deployed on mobile devices for real-time facial expression recognition.
1.3 Contributions

Accordingly, this thesis provides research works with the following four main contributions:

1. This thesis proposes a novel deep learning-based method using fully convolutional network (FCN) to accurately locate the eye centre, which treats eye centre localization as a special subproblem of the task of semantic segmentation. Experiments on challenging databases show that the proposed method has higher accuracy of eye centre localization compared with the state-of-the-art approaches and thus can be an alternative solution for wild environments.

2. A study of the relationship between crucial regions and facial expressions is presented. A novel relation-aware approach for facial expression recognition called Relation Convolutional Neural Network (ReCNN) is proposed in this thesis, which can adaptively capture the relationships between crucial regions and facial expressions and focus on the most discriminative regions. ReCNN computes the relation weight from coarse to fine through a two-level relation module and then uses the relation weight to generate weighted features as a final representation for facial expression. The evaluation on the in-the-wild databases of facial expression shows that ReCNN has superior performance over state-of-the-art approaches and this demonstrates the relationship is beneficial for further improving facial expression recognition performance.

3. Inspired by ReCNN, considering the role of crucial facial regions in in facial expression synthesis, this thesis further proposes a novel end-to-end method called Local and Global Perception Generative Adversarial Network (LGP-GAN) for facial expression synthesis. The proposed method utilizes the local network to capture texture details of these crucial facial regions and the global network to learn the general structure and profile of the face. The proposed LGP-GAN has a two-stage cascaded architecture that divides the facial expression synthesis process into local facial region generation and global facial image generation, which can synthesize facial expressions step by step. The qualitative and
quantitative experiments show that LGP-GAN has superiority on the public database compared with the state-of-the-art methods.

4. Towards developing and implementing real-time facial expression recognition on mobile devices, this thesis designs a light-weight CNN architecture with low complexity and high performance. The proposed method can be easily ported and deployed on mobile devices with limited computing capacity without taking up too much memory or storage space. This thesis also implements a real-time facial expression recognition system on the mobile device using the proposed network, which demonstrates the feasibility of the proposed method for mobile development.

1.4 Outline

The rest of the thesis is structured as follows:

Chapter 2 - Literature Review: This chapter provides an extensive review on facial behaviour analysis for affective computing. Special focus will be put on the two facial behaviour analysis subtasks, namely eye analysis and facial expression analysis. In the end, this chapter outlines some application areas, current challenges and future directions. This chapter aims at providing a systemic and comprehensive understanding of the background of this thesis for readers.

Chapter 3 - Accurate and Robust Eye Centre Localization via Fully Convolutional Networks: It proposes a novel end-to-end approach based on a fully convolutional network (FCN) for the task of eye centre localization, which can accurately locate the eye centre. The key idea of the proposed method is to treat eye centre localization as a special subproblem of the task of semantic segmentation. Evaluations on challenging databases show that the proposed method is superior to the state-of-the-art approaches on the performance of eye centre localization. A great performance improvement achieved by the proposed
method demonstrates that it is an alternative solution for some challenging real-world scenarios.

Chapter 4 - Relation-Aware Facial Expression Recognition: This chapter proposes a novel relation-aware facial expression recognition method called Relation Convolutional Neural Network (ReCNN), which can adaptively capture the relationship between crucial regions and facial expressions leading to the focus on the most discriminative regions for recognition. Specifically, the relationship is quantified via a value termed as the relation weight which can reflect the importance of crucial regions to facial expressions. The proposed ReCNN can compute the adaptive relation weight of the crucial region and thus learn weighted features to represent the facial expressions according to the relation weight. Extensive experiments on two large in-the-wild databases show that ReCNN has superior recognition accuracy over state-of-the-art approaches and the relationship between crucial regions and facial expressions can facilitate further improvements in the performance.

Chapter 5 - Local and Global Perception Generative Adversarial Network for Facial Expression Synthesis: It develops a novel end-to-end facial expression synthesis method with a two-stage cascaded structure, called Local and Global Perception Generative Adversarial Network (LGP-GAN). LGP-GAN can combine the generated results from the global network and local network into the corresponding facial expressions. In Stage I, LGP-GAN utilizes local networks to capture the local texture details of the crucial facial regions and generate local facial regions, which fully explores crucial facial region domain information in facial expressions. And then LGP-GAN uses a global network to learn the whole facial information in Stage II to generate the generate final facial expressions building upon local generated results from Stage I. The qualitative and quantitative experiments on the commonly used public database show that LGP-GAN outperforms state-of-the-art approaches on the performance of face expression synthesis.
Chapter 1: Introduction

Chapter 6 - Real-Time Facial Affective Computing on Mobile Devices:
This chapter aims to design and implement a network architecture for real-time mobile facial expression recognition. It proposes a light-weight CNN architecture for mobile development. Evaluations demonstrate that the proposed method outperforms state-of-the-art approaches in terms of recognition accuracy and computational complexity. Moreover, the implementation of the proposed method on an actual mobile device also demonstrates that it can maintain high running speed without taking up too much memory or storage space.

Chapter 7 - Conclusions: This chapter summarises this thesis with an in-depth discussion on its contributions and the future work.
Chapter 2

Literature Review

The pioneer work of affective computing can be traced back to 1997 by Rosalind Picard, who introduced the human affect and its practical applications in the field of human-machine interaction (Picard, 1997). Recently, affective computing has been received increasing attention from researchers, which has great potential for many applications in healthcare, sociable robots, autonomous-driving cars, etc. Research has shown that humans rely largely on nonverbal behaviours rather than verbal ways to express their emotions when communicating. Among the nonverbal means of communication, the face is considered to be the most visible and crucial channel since it contains a wealth of identity information such as age, gender and race, which are vital to people's daily lives. Moreover, some facial behaviours are also relevant in the people’s affective states. The facial behaviour of a person may indeed provide clues to read and understand the affective states. Therefore, researchers from the fields of image processing and computer vision have been studying how to automatically analyse facial affect and apply them to emerging applications to avoid subjective biases from humans.

With the development of hardware technology, smart devices, e.g., computers and mobile phones, have been an important part of people’s lives. Facial data collected from these devices can be processed and analysed through the facial behaviour analysis system. Numerous smart applications based on facial behaviour analysis have been developed such as driver monitoring systems and healthcare diagnosis systems. Traditional methods for facial behaviour analysis usually rely on professional knowledge and machine learning technology. However, with the advent of the "big data" era, these conventional approaches are not powerful enough to handle massive and explosive data. Moreover, most of the traditional
approaches use specially designed hand-crafted features, which are not robust enough and are very sensitive to the variation from the wild environment. The recent breakthroughs in deep learning and the computing capability of the hardware have been advanced some research fields such as computer vision and pattern recognition to a novel state-of-the-art. The technology of deep learning can directly learn multi-level abstract representations from raw data without feature engineering, which are more robust for the wild environment.

The advantages of deep learning make it attractive for facial behaviour analysis, which have made it possible to cope with wide variations in uncontrolled real-time environments and perform at par with human ability. Reviewing the previous work, the evolution of facial behaviour analysis can be divided into two different levels: The first level is the theoretical research which main focus is on how to design a new model to extract more robust features. The development of this level is very rapid, resulting in many solutions that achieve excellent accuracy and robustness on benchmark databases. The second level is applied research, trying to utilize the research results of the first level to address more specific issues in a particular field such as healthcare, education and autonomous driving. In the applied research, the application problems in a specific field are usually solved by researchers from multiple fields. They must find the best balance between the domain-related constraints and the existing technology in order to establish a very effective framework. For instance, deep learning-based facial expression recognition systems have achieved impressive performance recently. However, they are usually designed for normal human faces and they cannot be directly utilized to recognize facial expressions of facial palsy patients without considering the characteristics of facial palsy. Therefore, it is very meaningful to summarize works in the literature that use the technologies of deep learning to address issues of facial behaviour analysis related to specific applications. In addition, there is no comprehensive review of various facial behaviour analysis applications using deep learning. Motivated by this, the aim of this chapter is to fill up this research gap.
Chapter 2: Literature Review

This chapter summarizes the latest research on applying deep learning to facial behaviour analysis applications. As mentioned in chapter 1, this chapter surveys the relevant research of facial behaviour analysis from two perspectives: eye analysis and facial expression analysis. The eyes and facial expressions are the most flexible way to express emotions, which help us to quickly convey emotional information to others. We focus on three representative application scenarios of facial behaviour analysis, including smart vehicles, smart healthcare and smart education. We aim to investigate how to combine deep learning and facial behaviour analysis to develop a large number of emerging practical applications for various fields. The challenges and future research directions of deep learning in the application of facial behaviour analysis are also discussed in this chapter. We believe that this research will promote more practical deep learning research to realize smart applications of facial behaviour analysis.

The rest of this chapter is as follows: Section 2.1 explains some basic deep learning concepts. Section 2.2 summarizes studies focusing on the deep learning-based facial behaviours analysis and their application areas. Section 2.3 points out current challenges and future directions of facial behaviour analysis. Section 2.4 discusses the relation to the methods proposed in this thesis. Section 2.5 concludes the literature review in facial behaviour analysis.

2.1 Overview of Deep Learning

The origin of deep learning is the emergence of Artificial Neural Networks (ANNs) inspired by the neuronal networks of the human brain. Deep learning is part of the machine learning family, which can model extremely sophisticated functions and directly extract multi-level abstract representations from raw data. The major challenge of deep learning is that it needs powerful computing resources and the training process is very time-consuming. With the development and popularity of high-performance hardware such as Graphics Processing Units (GPUs), we can
establish a deep learning-based model to process data and solve the issues of real-world applications. Therefore, deep learning has become a powerful tool for processing massive and explosive data in the era of big data.

2.1.1 Basic Concepts

Compared with conventional machine learning methods, deep learning methods have recently reached a state-of-the-art level and even super-human performance in various application areas. This section gives a brief introduction of some basic deep learning concepts.

a) Input and output

The input of the network in deep learning can be any form of data, however, due to its success in the field of computer vision and image processing, the most common input data is image data. An image is usually represented as a matrix of pixel values (see Figure 2.1). The pixel refers to the point in an image, which its value ranges from 0 to 255 denoting the colour (0 indicating black and 255 indicating white). The output of the network depends on its task. Generally, deep learning tasks can be roughly divided into two categories: classification task and regression task, whose outputs are categorical value and continuous value respectively.

![Figure 2.1: Example of image data.](image)

b) Loss function

The loss function is responsible for supervising network training, which evaluates the error between the predicted value of the network and the ground truth of
training data. Categorical Cross Entropy (CCE) is commonly used in classification tasks, which is defined as:

\[
\text{loss}_{\text{CCE}} = - \sum_{x} g(x) \log p(x)
\]  (2-1)

where \( x \) denotes the number of samples in the training set; \( g(x) \) and \( p(x) \) are ground truth value and predicted categorical value, respectively.

For regression tasks, a widely used loss function is Mean Squared Error (MSE) which can be formulated as:

\[
\text{loss}_{\text{MSE}} = \frac{1}{x} \sum_{x} (g(x) - p(x))^2
\]  (2-2)

where \( x \) represents the number of samples in the training set; \( g(x) \) is the ground truth value and \( p(x) \) is the predicted continuous value.

c) Training algorithm

During the process of training, the network is trained to minimise the loss function using backpropagation and gradient descent algorithms. Backpropagation is a kind of automatic differentiation algorithm used to calculate the gradients for the weights in the networks. The gradient descent algorithms such as Stochastic Gradient Descent (SGD) (Robbins & Monro, 1951) and Adam (Kingma & Ba, 2014) use the gradients to adjust and update the weights.

d) Generalization

A good model can usually be generalized and applied to other data that has not been seen before. One of the challenges of training a network is overfitting which means that the network leans too well the training data and has a poor generalization to unseen data. The technologies of generalization are commonly used in training the network to avoid overfitting. This section introduces three representative generalization techniques used in the training process of deep learning.

**Data augmentation:** The problem of overfitting usually occurs if training data is too small and the network structure is deep with too many parameters to learn. Considering that it is not easy to collect large amounts of data or enlarge database
Chapter 2: Literature Review

for some learning tasks, data augmentation is an alternative solution which have been widely in deep learning to avoid overfitting. There are several operations in data augmentation to generate new data, such as rotation, scaling, cropping, random erasing and noise injection.

**Transfer learning:** Another solution for avoiding overfitting from insufficient training data is transfer learning. Transfer learning enables the small dataset to train an effective deep model without starting from scratch. It uses a small database of the specific task to fine-tune the network which pre-trained on a large related database.

**Early stopping:** Generally, test errors of the deep model cannot keep decreasing as the training time increases during the training process. After a certain period of training, over-fitting problems may occur, which means test errors start to increase. The technology of early stopping allows training to be stopped in time to obtain a model with minimal test error when test errors are found to start increasing and over-fitting occurs.

![Figure 2.2: Example of CNN architecture.](image)

2.1.2 Convolutional Neural Network

Convolutional Neural Network (CNN) is a typical network structure in deep learning which is inspired by the functioning of the human brain. CNN models became popular when it was successfully applied to handwritten digit classification. They are suitable for processing image data, and they have been successfully applied to a variety of computer vision and image processing tasks.
Recently, CNN has developed very rapidly, giving rise to various variants such as AlexNet (Krizhevsky, Sutskever, & Hinton, 2012), VGGNet (Simonyan & Zisserman, 2014) and ResNet (K. He, Zhang, Ren, & Sun, 2016). As shown in Figure 2.2, a typical CNN contains following four types of layers including convolutional layer, non-linearity layer, pooling layer and fully connected layer. The first three layers of the network extract discriminative features, and the last layer is responsible for specific tasks such as classification or regression. In this way, CNN converts the pixel values of the input image layer by layer into the final classification or regression values.

a) Convolutional Layer

A convolutional layer usually has several kernels filters, also called kernels, which perform the convolution operation sliding over the input image data. For instance, as shown in Figure 2.3, when an image is fed into a network, each filter slid over the width and height of the input image and extract features through computing dot product between them. Each filter contains a set of weight parameters. In the training step, weights in the filter start with random values and will be learned based on the training data. The backpropagation algorithm is used to calculate the gradient for each weight, which is then utilized by gradient descent algorithms to update the network weights for minimizing the loss function.

b) Non-linearity Layer
Chapter 2: Literature Review

After applying the convolutional function, a non-linearity layer is added to the output. The non-linearity layer contains a non-linear activation function that is responsible for whether to activate neurons of the network. The activation function applies an element-wise non-linear transformation on the input data to help the network to learn more complex relationships in the database. Generally, the Rectified Linear Unit (ReLU) activation function is commonly used in the non-linearity layer due to its robust performance (Glorot, Bordes, & Bengio, 2011), which conducts an element-wise operation and replaces all negative pixel values in the input with 0. There are other forms of non-linear activation functions as well such as Leaky ReLu, Sigmoid and Tanh (see Figure 2.4).

![Activation Functions Diagram](image)

Figure 2.4: Examples of different activation functions.

c) Pooling Layer

![Pooling Diagram](image)

Figure 2.5: Examples of Max pooling and Average pooling.
The pooling layer, or down-sampling layer, is applied to reduce the dimensionality of the input to save the most relevant information. In the pooling layer, a filter slides over the input data and performs the pooling operation. There are two main types of pooling operations, Max pooling and Average pooling. As shown in Figure 2.5, given a $4 \times 4$ input matrix, a $2 \times 2$ filter with a stride of 2 slides over the input matrix. For each $2 \times 2$ region in the $4 \times 4$ input matrix, Max pooling and Average pooling calculate maximum and average values respectively to obtain a new matrix.

**d) Fully Connected Layer**

The fully connected layer consists of several neurons which is responsible for the final classification or regression. As shown in Figure 2.6, it first converts the output from the previous layers into a single long continuous linear vector through flattening operation. The input linear vector is connected to all neurons and is performed linear and non-linear transformation to obtain the output. The whole process can be formulated as:

$$Z = \alpha(W^T \cdot V + b)$$  

(2-3)

where $Z$ presents the output of this fully connected layer, $\alpha$ is the non-linear activation function, $V$ refers to the input linear vector, $b$ and $W$ denote the network’s weight matrix and bias vector in this fully connected layer.

![Figure 2.6: Example of fully connected layer.](image)
2.1.3 Recurrent Neural Network

Recurrent neural network (RNN) is another network structure in deep learning which focuses on learning temporal features of the input data. Compared with CNN, RNN mainly processes sequential data such as video sequences instead of a single image. As shown in Figure 2.7, a general framework of RNN can be defined as:

$$h(t) = f(h(t-1), x(t))$$  \hspace{1cm} (2-4)

where $h(t)$ presents the output of current hidden unit, $h(t-1)$ is the output of previous hidden unit, $x(t)$ denotes the current input unit. Therefore, it can be found that RNN can remember and utilize the output results from the previous unit to compute the output of the current unit. However, RNN cannot process very long sequences and has gradient vanishing or exploding problems. To address these limitations, some variants of RNN such as Long Short-Term Memory (LSTM) (Hochreiter & Schmidhuber, 1997) and Gated Recurrent Unit (GRU) (Chung, Gulcehre, Cho, & Bengio, 2014) have been proposed. These variants can avoid the long-term dependency problem and determine what to remember in the previous and current unit to obtain better performance.

Figure 2.7: A general structure of RNN.
Chapter 2: Literature Review

2.2 Application Areas

Facial behaviour analysis is a multi-disciplinary research field with close links to other research fields such as psychology and computer vision. Over the past years, the research of automatic facial behaviour analysis including eye analysis and facial expression analysis has been extensively investigated and many approaches have been developed. Conventional machine learning methods used in facial behaviour analysis rely on hand-crafted low-level features extracted from images, which are not accurate and robust enough under real-world conditions. Due to the recent development of hardware, deep learning especially Convolutional Neural Network (CNN) is rapidly emerging and developing in facial behaviour analysis. The technologies of deep learning directly extract and utilize high discriminative features from images, which dramatically improve the performance without the use of hand-crafted features. Some existing deep learning-based methods have systematically been employed in various practical applications. This section recaps the studies regarding deep learning-based facial behaviour analysis including eye analysis and facial expression analysis with a focus on three representative application scenarios: smart vehicles, smart healthcare and smart education.

2.2.1 Smart Vehicles

In the field of smart vehicles, the driver monitoring system is becoming more and more popular, as it is highly important in ensuring the safety of drivers in the vehicle, and it is generally accepted as the first issue and essential step in realizing smart vehicles. Currently, most vehicle crashes happened because of the driver’s bad state, such as distracted, asleep or fatigued, or lost in thought. The advanced driver monitoring system with facial behaviour analysis can analyse and monitor the driver’s state, which will enhance driving safety. Smart vehicles usually can rely on artificial intelligence systems in conjunction with computer-controlled devices to gather and analyse facial information of the driver from the driver monitoring system to obtain emergency warnings. The facial behaviours of the
driver, such as eye gaze and facial expression, provide cues about their emotional state, fatigue, or abnormal behaviour. This kind of information could help smart vehicles to predict the driver’s state thereby avoid accidents. However, many challenges are faced by facial behaviour analysis system for monitoring the driver’s state such as significant changes in facial appearance and lighting conditions. The advent of deep learning provides an ideal solution to these issues of facial behaviour analysis.

Some deep learning-based facial behaviour analysis systems have been developed for driver monitoring (Q. Bi, Ji, & Sun, 2020; Bublea & Câleanu, 2020; Dari, Kadrileev, & Hüllermeier, 2020; C.-M. Kim, Hong, Chung, & Park, 2020; Lollett, Hayashi, Kamezaki, & Sugano, 2020; Lyu, Wang, & Meng, 2020; Rangesh, Zhang, & Trivedi, 2020; Vaca-Recalde, Pérez, & Echanobe, 2020). These systems usually set an ordinary camera on the dashboard to monitor changes in visual features of the driver’s face inferring the driver’s state. Choi et al. (I.-H. Choi, Hong, & Kim, 2016) proposed a simple CNN structure to classify 8 driver gaze zones. Vora et al. (Vora, Rangesh, & Trivedi, 2017) also proposed a CNN-based method for estimating driver gaze zones, but they only considered 6 gaze regions in this study. After that, they took a step forward by performing a comprehensive analysis of systems of driver gaze zone classification by using multiple CNN architectures (Vora, Rangesh, & Trivedi, 2018). Tayibnapis et al. (Tayibnapis, Choi, & Kwon, 2018) proposed a method for driver gaze zone estimation utilizing transfer learning technology, which fine-tuned a CNN model pretrained on a larger database to address the lack of training data. Kim et al. (W. Kim, Jung, & Choi, 2019) proposed a lightweight system to monitor the driver status like distraction and fatigue according to facial behaviours through multi-task MobileNets. Verma et al. (Verma & Choudhary, 2018) proposed a real-time driver emotion monitoring system based on VGG16 network. Hu et al. (Z. Hu, Lv, Hang, Huang, & Xing, 2021) used a multi-resolution network for the driver's attention area estimation based on the analysis of the gaze and the scenario image. Cha et al. (Cha et al., 2020) performed eye tracking of driver through a
multichannel CNN. A multi-task learning CNN is proposed by Yang et al. (D. Yang et al., 2020) to detect eye gaze and head pose for monitoring drivers’ attention. Oh et al. (Oh et al., 2021) combined facial expressions and biophysiology signals using deep learning to recognize the driver’s real emotion.

However, ordinary cameras or visible-light cameras cannot work normally in a dark environment or in areas with low visibility. An alternative solution is using near-infrared (NIR) cameras which are low-cost and can work well in a dark environment. Lee et al. (Lee, Yoon, Song, & Park, 2018) utilized multi-modal cameras including NIR and thermal camera to recognize facial emotions of drivers. Yoon et al. (Yoon, Baek, Truong, & Park, 2019) analysed the facial image captured from two NIR cameras calculating the driver gaze via a CNN framework. Naqvi et al. (Naqvi, Arsalan, Batchuluun, Yoon, & Park, 2018) used a NIR camera to directly detect the driver’s eye gaze through deep learning without any calibration. In another study (Naqvi et al., 2020), the same authors proposed a CNN-based method for aggressive driving detection, which utilized NIR cameras to detect the changes from drivers' gaze and facial expressions.

### 2.2.2 Smart Healthcare

The progress and popularity of deep learning have made it possible to improve traditional healthcare systems. Recently, many emerging smart healthcare applications combined with deep learning have been developed to provide better healthcare service and support. Automatic facial behaviour analysis has great potential in smart healthcare, and patients’ health status information can be obtained through the analysis of facial images. This section introduces some applications that perform facial behaviour analysis applying deep learning in the field of healthcare.

Nowadays, a variety of smart applications have been used for health monitoring which analyses facial behaviours to determine human health conditions. Yolcu et al. (Yolcu et al., 2017) proposed a novel neurological disorders monitoring system by recognizing facial expressions through deep
learning. Li et al. (C. Li, Pourtaherian, Van Onzenoort, a Ten, & De With, 2020) utilized Fast R-CNN to analyse facial expressions and states for monitoring infants. Wang et al. (F. Wang, Chen, Kong, & Sheng, 2018) deployed a CNN-based facial expression recognition system on real service robot to monitor health conditions of elderly people. Codina-Filbà et al. (Codina-Filbà et al., 2021) designed a mobile eHealth platform for home monitoring of bipolar disorder based on ResNet.

The systems of automatic facial behaviour analysis also play an important role in healthcare diagnosis and assessment. Pain assessment is an important clue to the patient’s condition in clinical situations, which can help treat and improve the patient's pain. Considering the pain can cause facial deformation and produce various spontaneous expressions, many works have been proposed using facial behaviour analysis for pain assessment. Salekin et al. (Salekin et al., 2021) assessed neonatal postoperative pain by using a multimodal spatio-temporal network. Zamzmi et al. (Zamzmi, Paul, Goldgof, Kasturi, & Sun, 2019) also designed CNN to detect neonatal pain from facial expression. Rodriguez et al. (Rodrigue et al., 2017) exploited LSTM for pain assessment based on facial expression classification. Bargshady et al. (Bargshady et al., 2020) proposed an automatic system analysing facial expressions by a hybrid deep model to assess levels of pain. Huang et al. (Y. Huang, Qing, Xu, Wang, & Peng, 2021) proposed a hybrid network structure which can extract multidimensional features from image sequences to estimate pain intensity. Tavakolian et al. (Tavakolian & Hadid, 2018) used CNN to classify levels of pain based on facial expressions and they further proposed a spatiotemporal CNN to automatically assess levels of pain according to facial dynamics in (Tavakolian & Hadid, 2019). Huang et al. (D. Huang, Xia, Mwesigye, & Feng, 2020) proposed a novel network called PainAttentive for assessing pain intensity levels, which is a kind of spatio-temporal attention model and focuses on the saliency when extracting dynamic features.

In addition to pain assessment, deep learning-based facial behaviour analysis has been employed in the diagnosis applications of other diseases. For instance, Jiang et al. (Jiang & Zhao, 2017) proposed a deep learning-based method to
diagnose Autism Spectrum Disorder (ASD) through eye tracking. Leo et al. (Leo et al., 2018) used CNN to evaluate the ability of children with ASD to produce facial expressions. Jin et al. (Jin, Qu, Zhang, & Gao, 2020) utilized LSTM to diagnose Parkinson disease through facial expression recognition. Fei et al. (Fei et al., 2020) proposed a novel deep learning-based system to detect and diagnose mental state through emotion analysis. Melo et al. (de Melo, Granger, & Hadid, 2019) proposed a deep learning method to assess depression based on facial expressions using convolutional 3D networks. Recently, the same authors performed depression detection according to facial behaviours using a multiscale spatiotemporal network (de Melo, Granger, & Hadid, 2020).

2.2.3 Smart Education

Smart education is the combination of education and the technologies of artificial intelligence to improve traditional education. The state of the students is closely related to their learning efficiency, which can be inferred from their facial behaviours. In recent years, a myriad of the deep learning-based facial behaviour analysis system has been developed to apply in the field of smart education. The facial behaviour analysis system can automatically analyse students’ facial behaviours to identify changes of their emotion and attention during the learning process. This allows teachers to have feedback in time and then adjust the teaching progress and teaching contents to meet students’ need.

In traditional classroom teaching, teachers determine the state of students observing students’ facial behaviours including eye gaze and facial expressions. Teachers usually make the visual inspection to get a comprehensive judgment of the student's state, which is based on individual experience. However, it is not easy to recognize the state of all students in the classroom. Thus, the automatic facial behaviour analysis system for the classroom learning environment is required, which can evaluate the state of students in real-time. Recently, some studies introduced deep learning-based facial behaviour analysis for classroom learning. A novel hybrid CNN is proposed to evaluate the affective states of students using
Chapter 2: Literature Review

facial expressions (Ashwin & Guddeti, 2020). Sümer et al. (Sümer et al., 2021) used deep learning-based method to analyse student's engagement from facial videos in the classroom. Pei et al. (Pei & Shan, 2019) proposed a CNN-based method to recognize students’ micro-expressions during the process of classroom learning.

With the prevalence of smart devices such as smartphones and tablets and the development of network technology, online learning gradually becomes a hot topic in the field of education. Recently, online learning has been gradually replaced traditional classroom learning and accepted by students because of the COVID-19 pandemic. Online learning makes it possible for many learners who have difficulties in having a face-to-face class to get an opportunity to study at home. One of the challenges of online learning is the lack of enough interaction between students and teachers. Therefore, when moving to online learning, it becomes of extreme importance to assess and monitor the level of engagement of students for providing useful feedback to teachers. Many works have utilized deep learning-based facial behaviour analysis to meet this need. Revadekar et al. (Revadekar, Oak, Gadekar, & Bide, 2020) utilized CNN to gauge the attention of students in an online learning environment through emotion-based attention detection. Pise et al. (Pise, Vadapalli, Sanders, & Applications, 2020) proposed a Temporal Relational Network to recognize the emotion changes of students in online learning. Carolis et al. (De Carolis, D'Errico, Macchiarulo, & Palestra, 2019) used LSTM to measure and monitor student engagement according to facial behaviours. Zakka et al. (Zakka & Vadapalli, 2020) utilized CNN to estimate student learning affect using facial emotions. Mukhopadhyay et al. (Mukhopadhyay et al., 2020) built a CNN model to classify the emotions and then identify the states of mind of the learners in an online learning system. Liao et al. (J. Liao, Liang, & Pan, 2021) proposed a Deep Facial Spatiotemporal Network to capture spatial and temporal information of the face evaluating student engagement in online learning. Nezami et al. (Nezami et al., 2019) utilized transfer learning to improve the performance of student engagement prediction. Zhu et al. (B. Zhu, Lan, Guo, Barner, &
Boncelet, 2020) used hybrid deep learning models to implement automatic student engagement detection. Wang et al. (W. Wang, Xu, Niu, & Miao, 2020) designed a framework based on CNN, which can monitor the emotions of students in real-time during the process of online learning. Dubbaka et al. (Dubbaka & Gopalan, 2020) detected learner engagement in massive open online courses using automatic facial expression recognition based on CNN. Wang et al. (Yuehua Wang, Kotha, Hong, & Qiu, 2020) investigated how to implement a real-time system for student engagement prediction under the wild environment and developed a CNN model to assess student engagement.

### 2.3 Challenges and Directions

In section 2.2, we have reviewed the state-of-the-art deep learning-based facial behaviour analysis methods in three application scenarios. In this section, we point out current challenges and future directions of this field from three main aspects.

#### 2.3.1 Data Collection

Since the current research of facial behaviour analysis has been moving from the constrained lab scenarios to the in-the-wild situations, many researchers have applied deep learning to address various challenges, such as large poses and occlusions. The technology of deep learning heavily depends on large databases with elaborately annotated label to enable better performance. Considering the complexity of annotation, it is difficult to collect a large annotated facial database, which requires specific expertise and a long time to complete. Moreover, another problem that usually occurs is imbalanced class distribution. This is determined by the nature of the human face. For instance, it is very easy to collect and label a facial image with an expression of happy but for other rare expressions it is difficult. Therefore, how to collect large high-quality data for facial behaviour analysis is a critical research issue.
Data synthesis is a feasible solution to the lack of training data, which has been a hot research topic and gained increasing attention recently. Much of the recent research emphasises on generating various facial data for facial behaviour analysis, for example, Wood et al. (Wood, Baltrušaitis, Morency, Robinson, & Bulling, 2016) proposed a method called UnityEyes to generate eye region images based on a generative 3D model. The synthesized images can be used to estimate gaze in difficult in-the-wild scenarios. The advent of Generative Adversarial Networks (GAN) provides a novel solution for facial expression synthesis. The typical GAN consists of a generator for generating fake images and a discriminator for distinguishing the generated results. They are simultaneously trained in the same framework, which is so-called adversarial learning. A conditional difference adversarial autoencoder (CDAAE) proposed by (Y. Zhou & Shi, 2017) generated certain target emotion states. A Double Encoder Conditional GAN (DECGAN) is proposed to generate facial images with desired facial expressions in (Mingyi Chen, Li, Li, Zhang, & He, 2018). There are also some methods generating facial expressions based on geometry information such as G2GAN (Song, Lu, He, Sun, & Tan, 2018) and GC-GAN (Qiao et al., 2018). ExprGAN proposed by (Ding, Sricharan, & Chellappa, 2018) edited the facial expressions based on the controllable expression intensity.

2.3.2 Nonfrontal Face

Previous research on facial behaviour analysis mainly focuses on images of the frontal or near-frontal faces, which may limit its practical application. In real life, the smart device may capture facial data from various poses. The ideal method of facial behaviour analysis should work normally for multiview facial images, especially nonfrontal faces. With the progress of hardware technology and deep learning, the performance of facial behaviour analysis on frontal faces has reached to a novel state-of-the-art. However, many existing facial behaviour analysis methods still have limited performance when analysing non-frontal faces because texture information of non-frontal face is more difficult to recognize and there are
very limited common features between frontal faces and nonfrontal faces. Moreover, for nonfrontal faces, the facial shapes and position relationship of facial organs are seriously changed and distorted. Therefore, how to analyse facial behaviours from a non-frontal face image is an urgently needed research issue.

Face frontalization refers to reconstructing the frontal faces from nonfrontal viewpoints. Recent research has demonstrated that face frontalization is an effective solution to dealing with nonfrontal faces in facial behaviour analysis. Recently, some works have proposed for face frontalization. Wang et al. (Yiming Wang, Yu, Dong, Stevens, & Liu, 2016) proposed a novel method for face frontalization based on a multi-template model which can well recover facial expressions from nonfrontal faces. After that, they presented a cascade support vector regression-based approach to learn the relationships between nonfrontal faces and their reconstructed frontal images (Yiming Wang, Yu, Dong, Jian, & Liu, 2017). Recently, the same authors further proposed a cascade regression-based method to recover frontal faces with facial expressions from nonfrontal images in real-time which can improve the performance of dynamic facial expression recognition (Yiming Wang, Dong, Li, Dong, & Yu, 2021).

2.3.3 Network Structure Design

Facial behaviour analysis plays an important role in many fields, such as smart vehicles, healthcare and education. Researchers have developed various deep network structure for facial behaviour analysis and achieved impressive performance in practical applications. As we know, it is necessary to maintain the high real-time performance of facial behaviour analysis for practical applications. However, the primary task of researchers is usually to focus on how to obtain higher performance and computation complexity has been largely ignored. A deep network with more parameters is often designed to reach the higher performance of facial behaviour analysis, which means that it has high computation complexity and cannot meet the real-time requirements. Moreover, deep learning generally has high requirements for hardware. Therefore, some high-performance devices
are usually performed the deep models. But these devices are often expensive and not portable enough. Therefore, how to design a lighter but high performance network to perform real-time analysis on a resource-limited portable device such as the smartphone is still a challenge.

Research on deep learning is still in an early stage. Much of the recent research has focused on improving the performance of the network by designing a deep structure with more parameters and high consumption of computing resources. How to optimize the network to make it implement on a resource-limited embedded device is a promising direction and trend for future research. Generally, there are two representative solutions that can meet this need. One is to utilize the cloud server for remote analysis. In this solution, mobile devices are only responsible for collecting data and uploading data to the server. However, this solution may have potential problems in privacy security and can be heavily affected by network latency when uploading data and receiving responses from the server. Another solution is to design a lighter network to maintain low computational complexity at the slight expense of network performance, which can enable deep models to be performed on mobile devices. For example, Guo et al. (Guo, Xia, Wang, Yu, & Chen, 2020) proposed a light-weight CNN architecture with low complexity and high performance which can be deployed on mobile devices to estimate facial affects in real-time.

2.4 Relation to Our Work

Despite numerous efforts made in recent years to implement facial behaviour analysis, there are still many unsolved issues that may affect the performance of existing methods. Therefore, we outline the issues addressed through our proposed methods in this thesis:

1. The task of eye centre localization occupies the crucial position of eye analysis. Most existing methods mainly rely on traditional handcrafted features,
which are not robust enough for the wild environment. To address this issue, we propose a novel deep learning-based method with high performance in Chapter 3.

2. For facial expression analysis, existing methods mainly take the whole face as a uniform source of features. The performance may be affected by redundant facial image information. We address this issue by proposing a novel relation-aware approach which can focus on the most discriminative regions. Moreover, we propose a novel facial expression synthesis method to solve the issue of the lack of training data. The experimental results of Chapter 4 and 5 show outstanding performance of the above proposed methods.

3. Most existing deep learning-based approaches are difficult to be deployed on mobile devices with limited calculation capacity. Therefore, we design a network architecture for mobile devices that can perform real-time facial expression recognition. The experimental results in Chapter 6 demonstrate the efficiency and feasibility of the proposed method for mobile development.

2.5 Summary

In this chapter, we have comprehensively reviewed how deep learning-based facial behaviour analysis applied in various application scenarios. Deep learning can efficiently process massive data which directly extracts high discriminate and robust features from raw data for the wild environment without feature engineering like traditional hand-crafted features. It offers us a new perspective to improve and solve traditional problems. From the perspective of applied research, we demonstrated how to combine various deep learning techniques and facial behaviour analysis to implement specific applications. We further point out current challenges and future directions of this field and discuss the relation to our work. We hope that this study will bring more interested investigators into this field and more effective deep learning methods for practical applications.
Chapter 3
Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

3.1 Introduction

Eye centre localization refers to localizing the centres of human’s pupil on given face images. Locating these centres means that we could establish correspondence between two eyes of the person and the focused targets, which has been proven to be useful for computer vision and human computer interaction tasks such as eye gaze estimation and eye tracking (Leo, Cazzato, De Marco, & Distante, 2013; Xia, Lou, Dong, Li, & Yu, 2018). Eye centre localization is the first step towards eye gaze tracking and estimation in images and video (H. Cai, Yu, Zhou, & Liu, 2016). During the process eye gaze estimation and tracking, we need to determine the precise pixel location of important key points of the eye centre for a single given RGB image. Moreover, achieving accurate eye centre localization is useful for higher level tasks (Cyganek & Gruszczynski, 2014; Jang, Mallipeddi, Lee, Kwak, & Lee, 2014; Lin, Li, & Liu, 2014; Zhentao Liu et al., 2017; Xing et al., 2017; H. Yu & Liu, 2014) such as human attention control, driver monitoring system and sentiment analysis, and also serves as a fundamental tool in fields such as human computer interaction and animation.

Eye centre localization has been an interesting topic in the field of computer vision in recent years. There are many factors that can affect performance of the eye centre localization such as the significant variability situation of eye appearance from different illumination, shape, colour and viewing angles. A good eye centre localization system must be accurate and robust to these factors. Early
works tackle such difficulties using specialized devices like infrared cameras or head-mounted devices. This kind of devices is very popular in commercial areas since they could apply infrared illumination to localize the eye centres through corneal reflections. In that case, these devices could obtain a high accurate eye centre location. However, it has some limitations in applications such as the high cost devices and the uncomfortable wearing experience. Compared with these specialized devices, the approaches which directly localizing key point positions of eye centre through computer vision and image processing techniques are more efficient since they only need a low-cost webcam instead of specific hardware devices and can be easily implemented. This method is often used as an alternative approach of infrared illumination in terms of the high accuracy and robustness.

The success of deep learning methods for various computer vision tasks in recent years motivates us to investigate it in the task of eye centre localization (F.-Y. Wang et al., 2018; B. Zhang, 2019; J. J. Zhang et al., 2018; N. ZHENG, 2019). Traditional methods have recently been reshaped by emerging deep learning techniques, which are the main driver behind an explosive rise in performance across many computer vision tasks (L. Chen et al., 2018; R.-C. Chen, 2019; L. Li, Lv, & Wang, 2016; Y. Tian, Li, Wang, & Wang, 2018; Q. Wang et al., 2018; S. Wang, Cai, Lin, & Guo, 2019). Fully convolutional network (FCN) has been proved to be successful not only in object semantic segmentation tasks, but also in other applications such as image classification or object detection. However, deep learning has rarely been mentioned and used for eye centre localization. Therefore, in this chapter, we introduce a novel end-to-end and pixel-to-pixel method for the eye centre localization via FCN.

The designed FCN takes an entire image of face as input and the predicted heatmaps as output. And then we transform the predicted heatmaps to landmark coordinates to get the eye centre location. The designed network follows two design principles: 1) we design a shallow structure rather than a deep one, which makes a good balance between performance and computational resources due to limited publicly available databases with accurate eye centre annotations. 2)
Chapter 3: Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

inspired by (Peng, Zhang, Yu, Luo, & Sun, 2017; B. Zhou, Khosla, Lapedriza, Oliva, & Torralba, 2014), we use a large kernel convolutional block instead of stacking small size (1×1 or 3×3). The key idea is based on the assumption that the eye centre localization can be considered as a special semantic segmentation problem. For the eye centre localization and semantic segmentation task, images are taken as the input, but the output of the former task is coordinates of landmarks and the latter one is the object’s class at each pixel. Thus, the key to implementing this assumption is that we need to establish correspondence between coordinates of landmarks and object's class at every pixel. To this end, we preprocess the images in which the coordinates of the eye centres are first transformed to a heatmap using Gaussian kernels. Then the problem becomes estimating the value of the heatmap at each pixel, which is equivalent to the semantic segmentation problem, where the goal is to estimate the object’s class at each pixel. Thanks to the strong performance of FCN for semantic segmentation, we design a shallow FCN network, which is similar to the one in (Long, Shelhamer, & Darrell, 2015) with a large kernel convolutional block and fine-tune it to transfer their performance from semantic segmentation to the eye centre localization task. The detailed experimental results show that the proposed approach outperforms state-of-the-art methods for eye centre localization in terms of accuracies and reliability.

The major contributions of this work are as follows:

- We design a fully convolutional network (FCN) with a shallow structure and a large kernel convolutional block to accurately locate the eye centre, which well balances the performance and the computational costs.
- We regard the problem of eye centre localization as a special semantic segmentation problem, which is a novel and important solution regarding the key and future directions for this area of research.
3.2 Related Work

3.2.1 Eye Centre Localization

Localizing the eye centre is a critical requirement for eye gaze estimation and eye tracking and has attracted a huge interest in recent years. Existing works for eye centre localization can be roughly divided into three categories: 1) appearance-based methods, 2) model-based methods, and 3) hybrid methods. Early works tackle this problem mainly using appearance-based methods, which use priori eye knowledge about appearance information such as the colour, circle structure and other geometric characteristics of the eye to localize the eye centre (Asteriadis, Nikolaidis, Hajdu, & Pitas, 2006; Bai, Shen, & Wang, 2006; Z.-H. Zhou & Geng, 2004). Valenti and Gevers (Valenti & Gevers, 2008) proposed a method using the isophote curvature method according to circle shape of eye to localize the eye centre. Moreover, based on the circle property of the eye, the means of gradient method proposed by Timm and Barth (Timm & Barth, 2011) is a milestone in the development of eye centre localization tasks. It can localize the eye centre by calculating the dot product of gradient vector and displacement vector. Based on means of gradient method, there are many improved or similar methods over recent years like (H.-B. Cai, Yu, Yao, Chen, & Liu, 2015; Soelistio, Postma, & Maes, 2015). Asadifard et al. (Asadifard & Shanbezadeh, 2010) proposed a method based on the cumulative density function (CDF), which mainly filters the image to determine which pixel is the eye centre. A method proposed by Leo et al. (Leo, Cazzato, De Marco, & Distante, 2014) used the local variability of the appearance and image intensities to determine the eye centre. Araujo et al. (Araujo, Ribeiro, Silva, & Goldenstein, 2014) described an Inner Product Detector for eye localization based on correlation filters. The appearance-based methods have achieved good performance, but under some challenging scenarios like poor illumination they are not robust and accurate enough. Zhang et al. (W. Zhang, Smith, Smith, & Farooq, 2016) introduced a modular approach making use of
Chapter 3: Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

isophote and gradient features simultaneously to estimate the eye centre locations. Villanueva et al. (Villanueva et al., 2013) proposed a method to detect the eye centre using a multiresolution and topographic method. George et al. (George & Routray, 2016) used geometrical characteristics for eye centre localization. Choi et al. (I. Choi & Kim, 2017) reviewed the local structure patterns LSPs and extended them by using several hybrid LSPs for accurate eye detection.

Model-based and hybrid methods are alternative solutions for eye centre localization. Model-based methods mainly use machine learning algorithms. It first extracts key features of images to train a model regarding appearance or structures of eye and then fit the learned model to determine eye centres. Many machine learning algorithms have been used for eye centre localization such as Bayesian models (Everingham & Zisserman, 2006), hidden Markov models (HMMs) (Samaria & Young, 1994), support vector machines (SVM) (Campadelli, Lanzarotti, & Lipori, 2009; S. Chen & Liu, 2015; Hamouz et al., 2005) and AdaBoost (Niu, Shan, Yan, Chen, & Gao, 2006). Kim et al. (S. Kim et al., 2007) localized eye centres using a multi-scale approach, which was based on Gabor vectors. A multi-layer perceptron was used by Jesorsky et al. (Jesorsky, Kirchberg, & Frischholz, 2001) to determine the position of eye centre. Kroon et al. (Kroon, Hanjalic, & Maas, 2008) employed a Fisher Linear Discriminant to filter the face image and selected the highest responses as the eye centre. Chen et al. (D. Chen, Tang, Ou, & Xi, 2006) used a hierarchical FloatBoost and MLP classifier simultaneously to localize the eye centre. Cristinacce et al. (Cristinacce, Cootes, & Scott, 2004) used Active Appearance Model (AAM) to find the eye centre positions. Behnke (Behnke, 2002) proposed a hierarchical network with local recurrent connectivity for this task. A cascade regression model was trained by Gou et al. (Gou, Wu, Wang, Wang, & Ji, 2016; Gou et al., 2017; Gou, Zhang, Wang, Wang, & Ji, 2019) using synthetic photorealistic data, which was used to determine the eye centre. Markus et al. (Markuš, Frljak, Pandžić, Ahlberg, & Forchheimer, 2014) localized the eye pupil by using an ensemble of randomized regression trees. Chen et al. (S. Chen & Liu, 2013) used clustering-based
discriminant analysis (CDA) models to localize the eye centre. Ren et al. (Ren, Wang, Hou, & Ma, 2013) proposed a codebook of invariant local features and a pyramid-like sparse representation classifier to locate the eyes. Hamouz et al. (Hamouz, Kittler, Kamarainen, Paalanen, & Kalviainen, 2004) used a GMM-based feature detector and an enhanced appearance mode to localize the eye centre. Compared with appearance-based methods, the model-based method is more robust. However, this kind of methods relies on lots of annotated training data, which is difficult to obtain in many cases. Hybrid methods integrate the advantages of appearance-based and model-based method simultaneously in one method like (Campadelli, Lanzarotti, & Lipori, 2006; Türkan, Pardas, & Cetin, 2007). In order to deal with occlusions of the eyelids under certain lighting conditions, Valenti et al. (Valenti & Gevers, 2011) proposed a hybrid method using mean shift and machine learning algorithm to improve their previous isophote method (Valenti & Gevers, 2008).

3.2.2 Fully Convolutional Network

In the domain of deep learning, FCN is widely used for semantic segmentation to predict object’s class at each pixel in an image according to its semantic meaning. Semantic segmentation is one of the most active research areas over recent years in computer vision. Early works (Deng, Todorovic, & Jan Latecki, 2015; Silberman, Hoiem, Kohli, & Fergus, 2012) mainly relied on handcrafted features to generate the label map to solve this problem. FCN proposed by Long et al. (Long et al., 2015) is a special variant of convolutional neural networks (CNNs). This method is an encoder-decoder architecture taking the existing CNNs model like VGG-16 as powerful tools to learn hierarchical features, which transform these models into a fully convolutional form by replacing the original fully connected layers with convolutional layers. Then upsampling or deconvolution is used to output the class of prediction for each pixel. FCN is the first end-to-end and pixel-wise predicting model, which provides a novel and milestone solution and opens a new research area for semantic segmentation. It is also the foundation for other
contemporary semantic segmentation algorithms. Based on the principle of FCN, many variations have been proposed for semantic segmentation over recent years (L.-C. Chen, Papandreou, Kokkinos, Murphy, & Yuille, 2014, 2017; Noh, Hong, & Han, 2015; S. Zheng et al., 2015). Note that all the aforementioned methods are used for semantic segmentation. Recently, however, the FCN-like network structure has been also applied successfully to other keypoint detection problems such as human pose estimation (Newell, Yang, & Deng, 2016), facial landmark detection (J. Yang, Liu, & Zhang, 2017) and eye gaze estimation (Park, Spurr, & Hilliges, 2018; Park, Zhang, Bulling, & Hilliges, 2018). They all have an encoder-decoder architecture and used an FCN-like network structure called hourglass network which borrows the idea from FCN.

The method proposed in this chapter is inspired by both the semantic segmentation task and FCN, which regard eye centre localization as a special semantic segmentation task. Therefore, we design a shallow FCN network with a large kernel convolutional block to overcome the limitations of previous works for eye centre localization. It is a feasible and high-efficiency solution for eye centre localization, which leads to high performance outperforming many existing state-of-the-art methods.

3.3 Methodology

In this chapter, we mainly focus on designing a network to achieve the task of localization of left eye centre and right eye centre. In this section, we give a detailed description of the proposed deep learning approach for eye centre localization. Figure 3.1 shows the brief flowchart of the proposed method. We design a shallow FCN network inspired by (Long et al., 2015; Peng et al., 2017; B. Zhou et al., 2014) with a large kernel convolutional block. The major advantage of our approach is that we regard the eye centre localization as a special semantic
Chapter 3: Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

Figure 3.1: Overview of our method for eye centre localization using shallow fully convolutional network. First, given an input image, the image is cropped to the size of face bounding box provided by face detection algorithm. Then, the face image is fed into shallow FCN with a large kernel convolutional block. Then the feature map is mapped to each pixel by deconvolution operation to predict the per-pixel eye region. And the network outputs the heatmap. Finally, the heatmap generated by the network is transformed back to the normal landmark coordinates. In this way, we can get the final results of eye centre position.

The key to transforming eye centre localization to the semantic segmentation problem is the preprocessing stage. The images of the training set are first cropped based on the face bounding box provided by the database. Knoche et al. (Knoche, Merget, & Rigoll, 2017) researched the effect of the image resolution on segmentation problem. And the transformation of the image to a heatmap allows the network of semantic segmentation to focus on the landmark detection of the eye centre.

3.3.1 Preprocessing

The key to transforming eye centre localization to the semantic segmentation problem is the preprocessing stage. The images of the training set are first cropped based on the face bounding box provided by the database. Knoche et al. (Knoche, Merget, & Rigoll, 2017) researched the effect of the image resolution on
performance of facial landmark prediction and found that there was a decline of performance when the image resolution is smaller than 50×50 px. We thus, resize all the cropped face images to be an equal size of 96×96 px. And then we transform all the processed images to a grey level for a stable performance. This can also improve the efficiency in processing and training.

Finally, according to the landmarks of eye centres, we transform these images to a heatmap as inputs of the network. Note that the successful use of the network of semantic segmentation on eye centre localization heavily depends on the generation of heatmap. We transform each landmark to a single heatmap using Gaussian kernel. For the eye centre localization problem, there are two landmarks (left and right eye centre). This means that we need to generate 2 heatmaps for each eye image, which can be interpreted as a grayscale image in the range [0, 1]. The ground-truth landmark coordinates are set to white and the other position as black. In other words, a black heatmaps indicates that some landmarks are not recorded, so all pixels on this heatmap are set to 0. We use two formulas based on the Gaussian kernel to generate heatmaps of eye centre landmarks:

\[
H_l = \frac{1}{2\pi\sigma^2} e^{\exp\left(-\frac{(x-x_l)^2 + (y-y_l)^2}{2\sigma^2}\right)} \quad (3-1)
\]

\[
H_r = \frac{1}{2\pi\sigma^2} e^{\exp\left(-\frac{(x-x_r)^2 + (y-y_r)^2}{2\sigma^2}\right)} \quad (3-2)
\]

where \((x_l, y_l)\) and \((x_r, y_r)\) are the ground truth landmarks of left and right eye centre, \(H_l\) and \(H_r\) are corresponding values of the heatmap at position \((x, y)\) of the image. And \(\sigma\) is the standard deviation. The value of \(\sigma\) is an important parameter, which needs to be appropriately adjusted. The choice of \(\sigma\) is important to get sensible results. If the value of \(\sigma\) is too small, the heatmap becomes too sparse (mostly zero). If the value of \(\sigma\) is too big, the trained model focuses too much on estimating coordinates of other positions instead of eye centre positions. For generating heatmaps, we set \(\sigma = 3\), which achieves the best results in our experiment. The further discussion can be found in Section 3.4. Figure 3.2 shows the examples of the generated heatmaps of left and right eye centre.
3.3.2 Network Architecture

In this section, we introduce the proposed network architecture. We use the VGG16-FCN (Long et al., 2015) architecture as a basis for developing our eye centre localization network. Classical CNN uses the convolutional layers to extract local features in an image. On the top of convolutional layers, the fully connected layers use the inner product operation to integrate high-level local feature maps into a single feature vector to predict the label of each image. Therefore, it is not able to predict the label for each pixel. Recently, the trend has shifted towards using FCN to solve the dense prediction of each pixel. FCN is a special type of CNN, which replaces all fully connected layers with the convolutional layers and adds additional upsampling or deconvolution layers.

After upsampling or deconvolution layers, the output feature maps of the network can be transformed to probability maps with sigmoid outputs $f_{y_1}$ by...
passing through a perceptron layer. $f_{y_i}$ represents the probability of predicting class $y_i$ at pixel $i$. The final results of class prediction $\hat{y}_i$ can be represented as the formula:

$$\hat{y}_i = \arg\max_{y \in \mathcal{Y}} f_{y_i}$$

(3-3)

where $\mathcal{Y}$ is a set of possible categories. Unlike a typical CNN, FCN could perform end-to-end and pixel-to-pixel classification and output a tensor of pixel-wise class predictions without additional post-processing. The spatial size of the tensor is equal to the input image, which is implemented by using several upsampling or deconvolution layers. Since the output of the deep layer lacks location and edge clues, the FCN combines feature maps of deep and shallow layers to obtain finer results called “FCN-xs” (like FCN-8s). For more details on FCN, see (Long et al., 2015).

Our network architecture is a shallow and simplified version of FCN with a large kernel convolutional block, which is also an encoder-decoder structure shown in Figure 3.1. In this work, we use the first two convolutional blocks from VGG16-FCN (Long et al., 2015) for encoders. Each convolutional block includes two convolutional layers and one Maxpooling layer. The parameters are set as the same as those in (Long et al., 2015). And the remaining layers of VGG16-FCN are discarded. The numbers of channels of two convolutional blocks at different resolutions are 64 and 128 respectively.

For traditional network architectures, stacking convolutional blocks with small size kernels (1×1 or 3×3) in the entire network is more efficient than using large kernels. However, in the experiment, Zhou et al. (B. Zhou et al., 2014) proposed the concept of valid receptive field (VRF) and claimed that the sizes of the actual receptive were always smaller than the theoretical receptive fields for traditional network architectures. Based on this work, Peng et al. (Peng et al., 2017) concluded that the large kernel size which could lead to more effective receptive
field played an important role in the field of semantic segmentation and could improve the performance.

Inspired by (Peng et al., 2017), we propose to use large kernel convolutional blocks in our network after the outputs from previous encoder convolutional blocks. However, the direct use of a large kernel size will increase the computational burden due to the large number of parameters. In our method, we employ a simulation of a K×K convolutional kernel including a combination of one K×1 convolutional kernel and one 1×K convolutional kernel to replace the direct use of a large kernel size. For the large kernel convolutional block, we set K=15 and use two path convolutional operations shown in Figure 3.1. Each path contains two convolutional layers with kernel the size of 15×1 and 1×15 respectively. Therefore, the output feature maps of the convolutional block in the encoder pass through a large kernel convolutional block with a kernel size of 15 and a filter number of 256 for a large receptive field. After two path convolutional operations, we aggregate the feature maps of two paths. Finally, the output of the large kernel convolutional block is upsamled with a deconvolution layer, which is used to output the prediction results. The input of the network is the cropped face image and the heatmap. The cropped face image input to the network is a grey level image with a resolution of 96×96 px. The ground truth label generated by using the Gaussian kernel is to generate heatmaps of two eye centre landmarks. The output of the network is the heatmap with the size of 96×96 px.

The training procedure for eye centre localization is similar to the one training FCN for semantic segmentation, which uses the images and labels of each pixels as input. We use heatmaps as the labels, which are generated by using landmarks of eye centres. Landmarks of eye centres are encoded using the Gaussian kernel to generate heatmaps at the provided location of the eye centre landmarks. Each eye centre landmark has its own heatmap and allows the network to distinguish between two points more easily. The eye centre localization network training is formulated as a per-pixel regression problem based on the ground-truth
Chapter 3: Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

Segmentation masks. Formally, the objective function can be represented as the following formula:

\[ \varepsilon(\theta) = \sum_p e(X_\theta(p), l(p)) \]  (3-4)

where \( p \) is the index of the pixel, \( l(p) \) is the ground truth heatmap which represents ground truth label of the pixel, and \( X_\theta(p) \) is the predicted heatmap which indicates estimated label predicted by the fully convolutional network with parameters \( \theta \). The network parameters \( \theta \) are updated by using RMSprop optimizer. And \( e(X_\theta(p), l(p)) \) is the loss function.

During the training stage, all parameters \( \theta \) are learned and updated via minimizing a loss function, which are computed as errors between the predicted heatmap and the ground truth data. Usually, the mean squared error (MSE) loss function is used for this kind of problems. During the training procedure, the MSE loss function of the proposed network is thus given by,

\[ \text{loss} = \frac{1}{N} \sum_{n=1}^{N} \| h_n - g_n \|^2 \]  (3-5)

where \( h_n \) represents the heatmap of \( n \)-th landmark predicted; \( g_n \) refers to the ground truth heatmap of \( n \)-th landmark; \( N \) is the total number of target facial landmarks.

3.3.3 From Heatmaps to Coordinates

In order to evaluate the training model performance, we need to transform heatmaps generated by the network to the normal landmark coordinates as shown in Figure 3.3. To this end, a straightforward method is to use the landmark coordinates of the pixel with the largest estimated density in the heatmap as the estimated landmark coordinates. We find that this method usually works well, but sometimes it is not accurate enough and thus results in outliers. To solve this issue, we use the weighted average of these coordinates corresponding to the pixels instead. To improve the accuracy and reduce the impact of outliers, the result of weighted average is further refined by considering only those pixels with the top
Chapter 3: Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

Figure 3.3: Example output produced by our network. On the left we see the final eye centre positions provided by weighted average method across each heatmap. On the right we show sample heatmaps (From left to right: right eye centre and left eye centre).

$N$ largest estimated density. Therefore, the problem becomes how to determine the value of $N$, which is used to calculate the weighted average of the coordinates. In our experiment, we set $N = 36$ which achieves the best performance for eye centre localization.

3.4 Experimental Results

In this section, we first introduce the database including the training and test set, experimental settings and the evaluation metric. We have also compared with other existing state-of-the-art methods on the public database including BioID (Jesorsky et al., 2001) and GI4E (Villanueva et al., 2013).
Chapter 3: Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

3.4.1 Database

In the experiment, we use the database from (Y. Sun, Wang, & Tang, 2013) as the training set. This dataset consists of 13,466 face images from real-world conditions, among which 5,590 images are selected from LFW database (G. B. Huang, Mattar, Berg, & Learned-Miller, 2008) while the remaining 7,876 images are downloaded from the web. These facial images have a clear difference in shape, expression and occlusions. Each face in this database is manually labelled with 5 landmarks including left and right eye centre. We only use landmarks of left and right eye centre in our experiment.

Moreover, for a fair comparison with other existing state-of-the-art methods, we choose two public databases as the test set and evaluate the proposed method on this database. The test set BioID (Jesorsky et al., 2001) is composed of 1,521 grey level images taken from 23 different subjects under various illumination, poses and locations. This database is regarded as the most challenging and realistic databases and then widely used for eye centre localization. Images of this database have a low-resolution size of 286×384 px. The left and right eye centre of each image are labelled in this database.

Another test set GI4E (Villanueva et al., 2013) contains 1,236 high quality RGB images from 103 subjects with 12 different gaze directions. These images have a resolution of 800×600 px, which are similar to images acquired by a normal camera. The eye centres are also labelled in this database. Both of the training and test sets are challenging and realistic in terms of appearance, pose, expression and occlusion.

3.4.2 Evaluation Criteria

In the testing stage, performance is measured with the maximum normalized error (Jesorsky et al., 2001), which is the standard evaluation metric for eye centre localization. It indicates the accuracy and reliability of each algorithm by
calculating the maximum error from the worst estimations of both eyes. The
detection error is measured as

\[
err = \max \left( \frac{\sqrt{(x'_l - x_l)^2 + (y'_l - y_l)^2}}{\sqrt{(x_l - x_r)^2 + (y_l - y_r)^2}} \right) \quad (3-6)
\]

where \((x'_l, y'_l)\) and \((x_l, y_l)\) are the estimated position and the ground truth of left
eye centre, and \((x'_r, y'_r)\) and \((x_r, y_r)\) refer to that of the right eye centre. During
evaluation, if the maximum normalized error is larger than 0.25, it is regarded as
failure. There are some special thresholds which are meaningful and usually used
to evaluate algorithms for eye centre localization: \(err = 0.05 \approx \) the diameter of
pupil; \(err = 0.10 \approx \) the diameter of iris; \(err = 0.25 \approx \) the distance between the
eye centre and the eye corners. Therefore, in order to estimate the eye centre point
located in the eye region, the error should be less than or equal to 0.25.

### 3.4.3 Experimental Settings

All images used in our experiment including the training and test set are cropped
using a bounding box to obtain a clear face area. And then the cropped face images
are further processed to grey level images with a size of 96×96 px. And the ground
truth heatmap of two eye centre landmarks is generated by using the Gaussian
kernel with a size of 96×96 px.

During the training process, we use 10,000 images from database (Y. Sun et
al., 2013) for training and 3,466 images for validation. But for deep learning
methods, the amount of data has a significant impact on the performance. But to
the best of our knowledge, only a few databases provide annotations of eye centres,
which are not enough to support the FCN network. Therefore, we use a data
augmentation method to augment the available training images to improve the
model performance on validation data. After splitting, the training set is augmented
via affine transformation including rotation (+/− 30 degrees) and scaling (0.75−
1.25) and horizontal flipping to increase size of the training set.
Chapter 3: Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

![Generated heatmaps using different values of standard deviation \( \sigma \).](image)

In our experiment, the network is trained using TensorFlow on a desktop PC with the specification of an Intel Core i7 at 4.20GHz processor, 16 GB of RAM memory and 8 GB NVIDIA GeForce GTX 1080 GPU. And we use RMSprop with a leaning rate of 5e-4 for optimization and set the batch size to 32. A Mean Squared Error (MSE) loss is computed comparing with the predicted heatmap to the ground truth heatmap generating from a 2D Gaussian kernel (with standard deviation \( \sigma = 3 \)) on the eye centres. To improve the performance of transforming heatmaps to coordinates, we use a weighted average of the top \( N = 36 \) largest estimated density instead of the largest one.

### 3.4.4 Quantitative Results

We first explore the impact of standard deviation \( \sigma \) on generating the heatmaps for our method. As mentioned in Section 3.3, we use the Gaussian kernel to generate heatmaps of eye centre landmarks according to (3-1) and (3-2) and set \( \sigma = 3 \) in our experiment. The impact on the generated heatmap using different parameters \( \sigma \) shown in Figure 3.4. From Figure 3.4, we can find that the size of heatmaps of the eye centre consistently increases with standard deviation \( \sigma \). It demonstrates the changes of performance of eye centre localization with different values of \( \sigma \). We
Chapter 3: Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

Figure 3.5: Experimental results on different values of standard deviation $\sigma$ on BioID database. The eye centre detection rate is evaluated by maximum normalized error. And Y coordinate denotes the rate with the normalized error less than 0.05.

Table 3.1: The performance of our proposed approach on BioID and GI4E database.

<table>
<thead>
<tr>
<th></th>
<th>$\text{err} \leq 0.05$</th>
<th>$\text{err} \leq 0.10$</th>
<th>$\text{err} \leq 0.25$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>BioID-Max</strong></td>
<td>94.4%</td>
<td>99.9%</td>
<td>100%</td>
</tr>
<tr>
<td><strong>BioID-Min</strong></td>
<td>98.9%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td><strong>BioID-Avg</strong></td>
<td>96.9%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td><strong>GI4E-Max</strong></td>
<td>99.1%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td><strong>GI4E-Min</strong></td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td><strong>GI4E-Avg</strong></td>
<td>99.8%</td>
<td>100%</td>
<td>100%</td>
</tr>
</tbody>
</table>

We have tested different values of standard deviation $\sigma$ ranging from 1 to 6 and obtained the performance on the BioID database shown in Figure 3.5. From the results we can see that for eye centre localization, when $\sigma \leq 3$, a larger value of standard deviation $\sigma$ will result in better performance yet for $\sigma \geq 4$ the performance drops. The possible reason for this is that a very high value of standard deviation $\sigma$ could lead to too many values of non-eye centre positions in the generated heatmap which reduce the effectiveness of heatmaps.

Based on the previous choice of standard deviation $\sigma$, we have obtained the overall performance of the proposed approach on BioID and GI4E database using three metrics shown in Table 3.1 including the maximum normalized error, the...
For quantitative results, we mainly focus on metric of the maximum normalized error in this work. For the BioID database, the maximum normalized error in Table 3.1 shows that our proposed approach can reach an accuracy of 94.4% ($err \leq 0.05$) which means that the estimated eye centres are located within the pupil with a high probability. Moreover, Table 3.1 also shows the accuracy of 99.9% ($err \leq 0.10$) indicating that eye centres estimated by our approach well lie within the iris. Finally, our approach yields an accuracy of 100% ($err \leq 0.25$) for localizing eye centre, which means that the method could meet the minimum standards of the eye centre localization and all the estimated eye centre points locate exactly within the eye region. Table 3.1 shows that our method has good performance with the accuracy of 99.1% ($err \leq 0.05$), 100% ($err \leq 0.10$), 100% ($err \leq 0.25$) on the GI4E database.
Chapter 3: Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

Figure 3.7: Qualitative results of our approach on the BioID and GI4E database. The images are sorted according to the maximum normalized error. The red points represent the estimated eye centre positions by our proposed approach. And the green points represent the ground truth. (a) For the BioID database, the first two rows are the best which $\text{err} \leq 0.05$ and $0.05 \leq \text{err} \leq 0.10$, and the bottom row is the worst which $0.10 \leq \text{err} \leq 0.25$. Note that all the results within the eye region which meet minimum standards of the eye centre localization ($\text{err} \leq 0.25$). (b) For the GI4E database, the first two rows are the best which $\text{err} \leq 0.05$, and the bottom row is the worst which $0.05 \leq \text{err} \leq 0.10$. Note that all the results of this database meet $\text{err} \leq 0.10$.

To further demonstrate the overall performance of our method, we also use the minimum normalized error and the average normalized error to evaluate the performance to give an upper bound and an average error. The minimum normalized error and the average normalized error replace the maximum function in (3-6) with the minimum and average function respectively. In Table 3.1, we can find that accuracies of almost all errors are 100%, indicating a reliable accuracy for eye centre localization.

3.4.5 Qualitative Results

The qualitative results of the proposed approach on BioID and GI4E database are shown in Figure 3.7. The red points are used to represent the estimated eye centre positions by the proposed approach. And the green points represent the ground truth positions of eye centre provided by the database. The first two rows show a selection of images of different subjects with various poses, facial expressions,
occlusions and lighting conditions. Row three shows the worst results estimated by the proposed approach due to occlusion from glasses, strong reflection and shadows making pupils invisible. Nevertheless, our method could obtain accurate eye centre points locating exactly within the eye region and meet the minimum standards of the eye centre localization ($err \leq 0.25$).

For BioID database, the results of the first two rows demonstrate that the proposed method is very accurate and robust under different challenging situations such as closed eyes, occlusion from glasses or hair, affection from shadows and far away from the camera. All estimated eye centres using the proposed method fall within the corresponding eye region, which meets the minimum standards ($err \leq 0.25$) of the eye centre localization. It is worth noting that even the four worst examples demonstrated are not failure cases. For GI4E database, it has similar performance on qualitative results to BioID database, reaching an accuracy of 100% when $err \leq 0.10$ which is more accurate than BioID database.

### 3.4.6 Comparison with Existing Approaches

We have extensively compared the proposed approach with the state-of-the-art methods on BioID and GI4E database using the maximum normalized error as the metric. The comparison results are shown in Table 3.2 and Table 3.3.

BioID is one of the most widely used databases with low quality images for eye centre localization. Many previous research results are available and easy to compare with using the same experiment protocol. In order to further investigate the overall performance of the proposed method, we first show results of 36 state-of-the-art methods for eye centre localization including appearance-based, model-based and hybrid method which include almost all-important eye centre localization methods published in recent years. Furthermore, we also increase the number of thresholds $err$ of the evolution metric of previous research. We employ five types of $err$ thresholds including $\{0.05, 0.10, 0.15, 0.20, 0.25\}$ instead of $\{0.05, 0.10, 0.25\}$. 
Chapter 3: Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

The comparison results between our approach and state-of-the-art methods on BioID database are shown in Table 3.2. From Table 3.2, we can make the following observations. Firstly, it is obvious that the proposed approach achieves the best performance for all kinds of thresholds err on BioID database compared with existing methods. Secondly, it is worth noting that the proposed approach obtains an accuracy of 94.4% at $\text{err} \leq 0.05$. This is a milestone achievement, since the most majority of existing methods maintains an accuracy of around 80%

Table 3.2: Comparison of our method with other methods on BioID database (bold value indicates best accuracy).

<table>
<thead>
<tr>
<th>Method</th>
<th>$\text{err} \leq 0.05$</th>
<th>$\text{err} \leq 0.10$</th>
<th>$\text{err} \leq 0.15$</th>
<th>$\text{err} \leq 0.20$</th>
<th>$\text{err} \leq 0.25$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Asteriadis et al., 2006)</td>
<td>44.0%</td>
<td>81.7%</td>
<td>92.6%</td>
<td>96.0%</td>
<td>97.4%</td>
</tr>
<tr>
<td>(Zhou et al., 2004)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>94.8%</td>
</tr>
<tr>
<td>(Bai et al., 2006)</td>
<td>37.0%</td>
<td>64.0%</td>
<td>-</td>
<td>-</td>
<td>96.0%</td>
</tr>
<tr>
<td>(Timm and Barth, 2011)</td>
<td>82.5%</td>
<td>93.4%</td>
<td>95.2%</td>
<td>96.4%</td>
<td>98.0%</td>
</tr>
<tr>
<td>(Cai et al., 2015)</td>
<td>84.1%</td>
<td>95.6%</td>
<td>-</td>
<td>-</td>
<td>99.8%</td>
</tr>
<tr>
<td>(Xia et al., 2018)</td>
<td>87.1%</td>
<td>98.7%</td>
<td>-</td>
<td>-</td>
<td>99.9%</td>
</tr>
<tr>
<td>(Valenti et al., 2008)</td>
<td>84.1%</td>
<td>90.9%</td>
<td>93.8%</td>
<td>97.0%</td>
<td>98.5%</td>
</tr>
<tr>
<td>(Soelistio et al., 2015)</td>
<td>80.8%</td>
<td>95.2%</td>
<td>97.8%</td>
<td>98.9%</td>
<td>99.4%</td>
</tr>
<tr>
<td>(Leo et al., 2014)</td>
<td>80.7%</td>
<td>87.3%</td>
<td>88.8%</td>
<td>90.9%</td>
<td>-</td>
</tr>
<tr>
<td>(Leo et al., 2013)</td>
<td>78.0%</td>
<td>86.0%</td>
<td>-</td>
<td>-</td>
<td>90.0%</td>
</tr>
<tr>
<td>(Asadifard et al., 2010)</td>
<td>47.0%</td>
<td>86.0%</td>
<td>89.0%</td>
<td>93.0%</td>
<td>96.0%</td>
</tr>
<tr>
<td>(Araujo et al., 2014)</td>
<td>88.3%</td>
<td>92.7%</td>
<td>94.5%</td>
<td>96.3%</td>
<td>98.9%</td>
</tr>
<tr>
<td>(Niu et al., 2006)</td>
<td>75.0%</td>
<td>93.0%</td>
<td>95.8%</td>
<td>96.4%</td>
<td>97.0%</td>
</tr>
<tr>
<td>(Chen et al., 2006)</td>
<td>-</td>
<td>89.7%</td>
<td>-</td>
<td>-</td>
<td>95.7%</td>
</tr>
<tr>
<td>(Jesorsky et al., 2001)</td>
<td>38.0%</td>
<td>78.8%</td>
<td>84.7%</td>
<td>87.2%</td>
<td>91.8%</td>
</tr>
<tr>
<td>(Gou et al., 2016)</td>
<td>89.2%</td>
<td>98.0%</td>
<td>-</td>
<td>-</td>
<td>99.8%</td>
</tr>
<tr>
<td>(Behnke, 2002)</td>
<td>37.0%</td>
<td>86.0%</td>
<td>95.0%</td>
<td>97.5%</td>
<td>98.0%</td>
</tr>
<tr>
<td>(Markus et al., 2014)</td>
<td>89.9%</td>
<td>97.1%</td>
<td>-</td>
<td>-</td>
<td>99.7%</td>
</tr>
<tr>
<td>(Kim et al., 2007)</td>
<td>96.4%</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>98.6%</td>
</tr>
<tr>
<td>(Everingham et al., 2006)</td>
<td>45.87%</td>
<td>81.35%</td>
<td>-</td>
<td>-</td>
<td>91.21%</td>
</tr>
<tr>
<td>(Ren et al., 2014)</td>
<td>77.08%</td>
<td>92.25%</td>
<td>-</td>
<td>-</td>
<td>98.99%</td>
</tr>
<tr>
<td>(Campadelli et al., 2009)</td>
<td>80.7%</td>
<td>93.2%</td>
<td>-</td>
<td>-</td>
<td>99.3%</td>
</tr>
<tr>
<td>(Chen et al., 2015)</td>
<td>88.79%</td>
<td>95.2%</td>
<td>-</td>
<td>-</td>
<td>98.98%</td>
</tr>
<tr>
<td>(Chen et al., 2014)</td>
<td>87.3%</td>
<td>94.9%</td>
<td>-</td>
<td>-</td>
<td>99.2%</td>
</tr>
<tr>
<td>(Hamouz et al., 2005)</td>
<td>58.6%</td>
<td>75.0%</td>
<td>80.8%</td>
<td>87.6%</td>
<td>91.0%</td>
</tr>
<tr>
<td>(Kroon et al., 2008)</td>
<td>65.0%</td>
<td>87.0%</td>
<td>-</td>
<td>-</td>
<td>98.8%</td>
</tr>
<tr>
<td>(Cristinacce et al., 2004)</td>
<td>57.0%</td>
<td>96.0%</td>
<td>96.5%</td>
<td>97.0%</td>
<td>97.1%</td>
</tr>
<tr>
<td>(Hamouz et al., 2004)</td>
<td>50.0%</td>
<td>66.0%</td>
<td>-</td>
<td>-</td>
<td>70.0%</td>
</tr>
<tr>
<td>(Turkan et al., 2007)</td>
<td>18.6%</td>
<td>73.7%</td>
<td>94.2%</td>
<td>98.7%</td>
<td>99.6%</td>
</tr>
<tr>
<td>(Campadelli et al., 2006)</td>
<td>62.0%</td>
<td>85.2%</td>
<td>87.6%</td>
<td>91.6%</td>
<td>96.1%</td>
</tr>
<tr>
<td>(Valenti et al., 2012)</td>
<td>86.1%</td>
<td>91.7%</td>
<td>-</td>
<td>-</td>
<td>97.9%</td>
</tr>
<tr>
<td>(Zhang et al., 2016)</td>
<td>85.7%</td>
<td>93.7%</td>
<td>-</td>
<td>-</td>
<td>99.2%</td>
</tr>
<tr>
<td>(Gou et al., 2017)</td>
<td>91.2%</td>
<td>99.4%</td>
<td>99.6%</td>
<td>-</td>
<td>99.8%</td>
</tr>
<tr>
<td>(Gou et al., 2019)</td>
<td>92.3%</td>
<td>99.1%</td>
<td>99.7%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>(George et al., 2016)</td>
<td>85.1%</td>
<td>94.3%</td>
<td>96.7%</td>
<td>98.1%</td>
<td>-</td>
</tr>
<tr>
<td>(Choi et al., 2017)</td>
<td>91.1%</td>
<td>98.4%</td>
<td>-</td>
<td>-</td>
<td>99.7%</td>
</tr>
<tr>
<td><strong>Our Method</strong></td>
<td><strong>94.4%</strong></td>
<td><strong>99.9%</strong></td>
<td><strong>100%</strong></td>
<td><strong>100%</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>
or lower. Finally, it shows that with the increasing maximum normalized error metric, the performance of the proposed method gets better. Compared with other methods, the accuracy of our method is the first one close to 100% as the maximum normalized error increase. Except for at $\text{err} \leq 0.05$, the accuracy is almost 100% when it comes to $\text{err} \leq 0.10$, $\text{err} \leq 0.15$, $\text{err} \leq 0.20$ and $\text{err} \leq 0.25$.

GI4E is another evaluation database for eye centre localization, which contains images with high quality taken by normal cameras. The results on GI4E are listed in Table 3.3. We have compared the proposed method with 7 state-of-the-art methods. As shown in Table 3.3, the performance on GI4E database is better than that on BioID in general. The proposed approach still achieves the best performance on GHE database with the accuracy of 94.4% ($\text{err} \leq 0.05$), 99.9% ($\text{err} \leq 0.10$), 100% ($\text{err} \leq 0.15$), 100% ($\text{err} \leq 0.20$) and 100% ($\text{err} \leq 0.25$) respectively. What’s more, the accuracy of our method is the only one that can achieve 100% accuracy at $\text{err} \leq 0.10$.

Another important consideration in evaluating the algorithm for eye centre localization is its computational complexity. The computational complexity is measured by average processing time for each input image. We have conducted a comparison in the processing time of locating the eye centres on BioID database.

Table 3.3: Comparison of our method with other methods on GI4E database (bold value indicates best accuracy).

<table>
<thead>
<tr>
<th>Method</th>
<th>$\text{err} \leq 0.05$</th>
<th>$\text{err} \leq 0.10$</th>
<th>$\text{err} \leq 0.15$</th>
<th>$\text{err} \leq 0.20$</th>
<th>$\text{err} \leq 0.25$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Timm and Barth, 2011)</td>
<td>92.4%</td>
<td>96%</td>
<td>96.9%</td>
<td>-</td>
<td>97.5%</td>
</tr>
<tr>
<td>(Villanueva et al., 2013)</td>
<td>93.9%</td>
<td>97.3%</td>
<td>98%</td>
<td>-</td>
<td>98.5%</td>
</tr>
<tr>
<td>(Zhang et al., 2016)</td>
<td>97.9%</td>
<td>99.6%</td>
<td>-</td>
<td>-</td>
<td>99.9%</td>
</tr>
<tr>
<td>(Gou et al., 2016)</td>
<td>98.2%</td>
<td>99.8%</td>
<td>-</td>
<td>-</td>
<td>99.8%</td>
</tr>
<tr>
<td>(George et al., 2017)</td>
<td>94.2%</td>
<td>99.1%</td>
<td>99.6%</td>
<td>-</td>
<td>99.8%</td>
</tr>
<tr>
<td>(Gou et al., 2016)</td>
<td>89.3%</td>
<td>92.3%</td>
<td>93.6%</td>
<td>94.2%</td>
<td>-</td>
</tr>
<tr>
<td>(Gou et al., 2019)</td>
<td>98.3%</td>
<td>99.8%</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td><strong>Our Method</strong></td>
<td><strong>99.1%</strong></td>
<td><strong>100%</strong></td>
<td><strong>100%</strong></td>
<td><strong>100%</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>

Table 3.4: Comparison of our method with other methods in average processing time.

<table>
<thead>
<tr>
<th>Method</th>
<th>(Araujo et al., 2014)</th>
<th>(Leo et al., 2013&amp;2014)</th>
<th>(Gou et al., 2016&amp;2017)</th>
<th>(Gou et al., 2019)</th>
<th>Our Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time(ms)</td>
<td>83</td>
<td>333</td>
<td>67</td>
<td>63</td>
<td><strong>5</strong></td>
</tr>
</tbody>
</table>
Chapter 3: Accurate and Robust Eye Centre Localization via Fully Convolutional Networks

We train a network model of the proposed method first through a desktop PC. And then deploy it on a standard laptop with an Intel Core i5 at 2.50GHz processor and 16GB of RAM memory for eye centre localization. The comparison of our method and other methods in average processing time is shown in Table 3.4. In our experiment, the proposed method is more efficient and faster than all other state-of-the-art methods, taking 5ms per image on average. This shows that our proposed method is suitable for real time applications and embedded systems.

3.5 Summary

In this chapter, we propose an accurate and robust network architecture for eye centre localization via a shallow FCN with a large kernel convolutional block. The key idea is regarding the eye centre localization as a special semantic segmentation problem, which leads to the transformation of heatmaps of eye centre positions. In the preprocessing stage, we first use the Gaussian kernel to generate heatmaps of eye centre landmarks, which are then used to train the network. In the testing stage, we transform the heatmaps generated by the network to coordinates to evaluate the performance. Our experimental results on testing database show that the proposed approach outperforms the state-of-the-art methods.
Chapter 4

Relation-Aware Facial Expression Recognition

4.1 Introduction

Facial expressions serve as one of the most vital affect signals for humans, which are the external representation of internal emotions and feelings. Hence, the task of facial expression recognition has been a hot research topic in various research fields such as computer vision, artificial intelligence and human-computer interaction (Gunes, Celiktutan, & Sariyanidi, 2019; Lian, Li, Tao, Huang, & Niu, 2020; Zhentao Liu et al., 2017; H. Yu & Liu, 2014). Currently, the research results of facial expression recognition have been used and played a crucial role in many applications, for instance, health care (Amminger et al., 2012), virtual reality (VR) (Lou et al., 2019), augmented reality (AR) (C.-H. Chen, Lee, & Lin, 2015), driver assistant systems (Vicente et al., 2015), and entertainment (Mourão & Magalhães, 2013).

Whilst facial expression recognition accuracy under laboratory settings has been improved significantly, it still remains a great challenge for facial images in the wild due to its significant changes in illumination conditions, head poses and subjects’ attributes, etc. One of the key solutions to overcome the challenge is to represent facial expressions through more robust features (Sariyanidi, Gunes, & Cavallaro, 2014). Conventional methods mainly utilize hand-crafted features to represent facial expressions but they lack the required generalization to the wild environment. Recently, as the high performance of deep learning for various computer vision tasks, features generated by deep learning networks have been demonstrated to be more robust and effective. Therefore,
Chapter 4: Relation-Aware Facial Expression Recognition

Figure 4.1: The differences between facial expressions often appear in crucial regions. All facial images are from the AffectNet (Mollahosseini, Hasani, & Mahoor, 2017) which is an in-the-wild database. (a) shows the example of the same facial expression (Happy) with high similarity in the eye and mouth regions from different subjects. (b) shows the example of two different facial expressions (Sad and Neutral) with low similarity in eye and mouth regions from different subjects.

Emerging deep features have gradually replaced traditional hand-crafted features, which obtain an explosive rise in the performance of facial expression recognition.

Much progress in facial expression recognition has been made using deep learning techniques, but most existing works only depend on processing the whole face for recognition. The performance may be affected by redundant facial image information. In fact, according to the physiology and psychology research such as Facial Action Coding System (FACS) (Ekman, Friesen, & Hager, 2002; Friesen & Ekman, 1978), it is obvious that the differences between facial expressions usually locate in some certain crucial regions such as eye and mouth instead of evenly appearing in the whole face (see Figure 4.1). Due to different facial expressions have explicit local variations, which correspond to crucial regions, it motivates us to investigate and utilize crucial regions in the task of facial expression recognition. Moreover, studies have been proven that the attention of
Chapter 4: Relation-Aware Facial Expression Recognition

Figure 4.2: Overview of the proposed ReCNN framework. It contains three parts for facial expression recognition: feature extraction module, crucial region generation module and relation module. The feature extraction module extracts the features using the pre-trained model. The crucial region generation module obtains the feature maps of the eye and mouth regions by cropping the feature maps of the whole face with a fixed position cropping strategy. Finally, the two-level relation module automatically computes the relation weights ($r_i$ and $R_i$) of each crucial region from coarse to fine via the Relation Extraction Net and then use the relation weights to generate weighted features as a final representation for facial expression.

Humans naturally focus on specific facial regions when they recognize and distinguish different facial expressions (Wegrzyn, Vogt, Kireclioğlu, Schneider, & Kissler, 2017), for example, eyes are more important for fear and the mouth is vital for recognizing happiness. Therefore, we can infer that there are certain relationships between these crucial regions and facial expressions can help facial expression recognition, which is strongly supported by physiological and psychological research. Although such relationships are crucial and beneficial for making accurate recognition, most state-of-the-art methods for facial expression recognition still recognize facial expressions depending on the whole face and deploy few efforts in exploiting relations. Inspired by this observation, the main goal of this work is to consider and utilize the potential relations between crucial regions and facial expressions to recognize facial expressions.

In this chapter, we propose a novel relation-aware method for facial expression recognition called Relation Convolutional Neural Network (ReCNN). Figure 4.2 shows an overview of the ReCNN framework. ReCNN is an end-to-end architecture with three components: feature extraction module, crucial region
generation module and relation module, which can adaptively capture the relationships between crucial regions and facial expressions and focus on the most discriminative regions such as eye and mouth. The value called relation weight is used to quantify relations, which can reflect the importance of crucial regions to facial expressions. Each final representation for facial expression recognition is weighed via Relation Extraction Net in relation module that computes the adaptive relation weight of the crucial region.

ReCNN aims at extracting the discriminative representations to improve the performance of facial expression recognition through the relationship between crucial regions and facial expressions. The contributions include the following:

- We propose a novel relation-aware facial expression recognition method called Relation Convolutional Neural Network (ReCNN) based on physiology and psychology research. ReCNN is an end-to-end architecture which can adaptively capture the relationships between crucial regions and facial expressions and focus on the most discriminative regions.

- We propose a two-level relation module, using Relation Extraction Net as the crucial component, to compute the relation weight from coarse to fine and then use the relation weight to generate weighted features as a final representation for facial expression.

- We have explored the relationship between crucial regions and facial expressions. The evaluation on the in-the-wild databases of facial expression including the basic facial expressions and compound facial expressions shows that the proposed ReCNN has superior recognition accuracy compared with state-of-the-art methods. It means that the relationship is beneficial for further improving the facial expression recognition performance.
4.2 Related Work

For facial expression recognition tasks, one common solution is first to extract robust features to represent facial expressions and then directly feed these features into the specific classifier such as support vector machines (SVM) (Shan, Gong, & McOwan, 2009), AdaBoost (Yubo Wang, Ai, Wu, & Huang, 2004) and hidden Markov models (HMMs) (Uddin, Lee, & Kim, 2009) to recognize and classify facial expressions. Hence, according to the type of the extracted features, existing works for facial expression recognition can be divided into two categories: hand-crafted feature-based or deep learning-based methods.

Previous approaches recognized six basic facial expressions defined by Ekman et al. (Ekman & Friesen, 1971) (Happy, Sad, Surprise, Disgust, Anger and Fear) mainly utilizing the hand-crafted features extracted from facial information. Hand-crafted features based on the type of the facial information can be classified into two categories: geometric features and appearance features. By describing the shapes and positions of the face and other facial components, geometric features can utilize the relationship between these facial components for facial expression recognition. Therefore, one of the key steps for geometric feature-based methods is to detect facial components. Most geometric feature-based methods utilize Active Shape Model (ASM) (Cootes, Taylor, Cooper, & Graham, 1995) and Active Appearance Model (AAM) (Edwards, Taylor, & Cootes, 1998) to detect facial components or track facial points. Appearance features describe the facial texture information caused by facial expressions such as histogram of oriented gradients (HoG) (Carcagnì, Del Coco, Leo, & Distante, 2015; Y. Hu et al., 2008), local binary pattern (LBP) (Carcagnì et al., 2015; Mistry, Zhang, Neoh, Lim, & Fielding, 2016; L. Wang, Li, Wang, & Chen, 2014), scale invariant feature transform (SIFT) (Tariq et al., 2011), and Gabor features (Gu, Xiang, Venkatesh, Huang, & Lin, 2012; Littlewort, Bartlett, Fasel, Susskind, & Movellan, 2004). Although hand-crafted features have high performance under laboratory settings, these features are still not generalizable enough for facial images in the wild.
because of significant variations in lighting, camera view, head pose, age, gender and ethnicity, etc. Moreover, it is difficult to infer high-level categories of facial expressions through low-level hand-crafted features since it has a semantic gap.

Recent progress in deep learning, especially deep neural networks (DNNs), has led to an explosive rise in performance for the tasks of various research fields such as computer vision and pattern recognition (F.-Y. Wang et al., 2018; Xia, Yu, & Wang, 2019; Xing et al., 2017; N. ZHENG, 2019). A large body of works using DNNs such as convolutional neural networks (CNNs) (Hasani & Mahoor, 2017; Mollahosseini, Chan, & Mahoor, 2016), autoencoder (Majumder, Behera, & Subramanian, 2016), recurrent neural network (RNN) (Rodriguez et al., 2017; T. Zhang, Zheng, Cui, Zong, & Li, 2018) and generative adversarial networks (GANs) (H. Yang, Ciftci, & Yin, 2018; F. Zhang, Zhang, Mao, & Xu, 2018) to recognize facial expressions have been proposed in recent years. Currently, DNNs have been widely utilized in the field of facial expression recognition, which are especially efficient for the task of facial expression recognition with high performance in terms of accuracy rate of recognition. The main reason is that DNNs can directly extract highly discriminative and high-level deep features of the raw data layer by layer which overcome the limitations of hand-crafted features. Moreover, the advent of large in-the-wild annotated databases of facial images further facilitates the development of the facial affective computing field, such as AffectNet (Mollahosseini et al., 2017) and RAF-DB (S. Li & Deng, 2019; S. Li, Deng, & Du, 2017).

Unlike the methods mentioned above that use features based on the whole face for facial expression recognition, another solution divides the whole face into several parts based on the pre-defined action units (AUs) from Facial Action Coding System (FACS) (Ekman et al., 2002; Friesen & Ekman, 1978) and then decode the facial expressions to single AU or AU combinations. In the FACS, AUs represent the facial muscle movements and the specific facial expressions can be represented by the movements of one or more AUs. Therefore, the discriminative features of facial expression usually appear in the regions corresponding to some
certain AUs which indicates the relationship between crucial regions and facial expression exists. Tian et al. (Y.-I. Tian, Kanade, & Cohn, 2001) implemented AU recognition through facial landmarks and geometrical modelling and then use detected AUs for facial expression analysis according to the FACS. Yang et al. (P. Yang, Liu, & Metaxas, 2010) and Zhong et al. (Zhong et al., 2012) proposed feature selection schemes to automatically extract the features as the final representation for facial expression recognition from the local facial regions corresponding to some AUs, but it still used hand-crafted features. Recently, Li et al. (Y. Li, Zeng, Shan, & Chen, 2019) proposed a patch-based attention network to recognize facial expressions under occlusion which divided the whole face into several patches around the facial landmarks and selected the un-occluded regions for recognition. However, this method mainly focusses on the occlusion aware facial expression recognition which has limited performance in non-occlusion situations.

Based on above analysis, inspired by the work in (Y. Li et al., 2019), we propose a Relation Convolutional Neural Network (ReCNN) which utilizes the relationship between facial regions and facial expressions. The major advantages of the proposed ReCNN are as follows: First, ReCNN extracts the high-level deep features layer-by-layer instead of hand-crafted features, which are more discriminative and robust for real-world scenarios; Second, ReCNN is an end-to-end network that adaptively captures the relationships between crucial regions and facial expressions and focuses on the most discriminative regions.

### 4.3 Methodology

We first present an overview of the proposed Relation Convolutional Neural Network (ReCNN) in this section and then introduce the details of each module in proposed network architecture. Finally, we give a description of the weighted loss function.
4.3.1 Overview

Figure 4.3: The framework of the proposed ReCNN. Given a facial image, ReCNN extracts global feature maps of the whole face through a pre-trained VGGFace [4] model. Then these feature maps from the Maxpooling layer (pool4 in VGGFace) are cropped into sub-feature maps including the eye and mouth region using a fixed position cropping strategy. These sub-feature maps are then processed by two convolutional layers and the global feature maps are also encoded using three convolutional layers and one Maxpooling layer. All the feature maps are fed into the relation module to obtain the final representation for facial expression recognition through a two-level structure and Relation Extraction Net. Finally, we use the weighted loss function as the softmax loss to optimize the whole network.

As illustrated in Figure 4.3, the proposed ReCNN consists of three modules: feature extraction module, crucial region generation module and relation module. After face detection, ReCNN firstly extracts global feature maps of the whole face by feeding the given facial image into a pre-trained CNN model (VGGFace (Parkhi, Vedaldi, & Zisserman, 2015)). Using a fixed position cropping strategy, these global feature maps are cropped into sub-feature maps of two facial regions (eye and mouth) to obtain two local patches. Then, these sub-feature maps are fed into the two-level relation module.

In the relation module, we use the value called relation weight to quantify the relationship between facial regions and facial expression. The first-level relation module uses the Relation Extraction Net to adaptively extract the coarse relation weights of two local patches and uses the relation weights to generate two local weighted vectors. The relation weights are used to denote the importance of the
region to facial expression recognition. For the global feature maps of the whole face, we also do the same operation to generate a global weighted vector. And we aggregate two local weighed vectors and one global weighted vector to a coarse global representation. Using a similar operation on the concatenation of each weighed feature vector and the coarse global representation, the second-level relation module further obtains a refined final representation. Finally, the whole network is optimized by minimizing the weighted loss function. also do the same operation to generate a global weighted vector.

4.3.2 Network Architecture

As mentioned in Section 4.2, the progress made in facial expression recognition so far mainly focuses on the whole face. In fact, the discriminative features of facial expression usually lie in specific crucial regions such as eye and mouth instead of the whole face. It is reasonable to infer that there is a certain relationship between sub-facial regions and facial expressions as described in FACS. To discover and utilize this relationship, inspired by (Y. Li et al., 2019), we design a ReCNN for facial expression recognition including three modules: feature extraction module, crucial region generation module and relation module (see Figure 4.3).

a) Feature extraction module

Feature extraction module is the first step of our proposed ReCNN. To obtain discriminative feature representation and reduce the number of parameters, a pre-trained CNN model is usually applied to extract feature maps. For the proposed ReCNN, a pre-trained VGGFace (Parkhi et al., 2015) model is used to extract global feature maps of the whole face in the feature extraction module. Since VGGFace is pre-trained for the face recognition task from 2D facial images, it fits well for the task of facial expression recognition. The extracted global feature maps \( M_G \) of an input facial image \( I_F \) is defined by:

\[
M_G = H(I_F)_{pool4}
\]  

(4-1)
where $H(\cdot)_{pool4}$ refers to the feature maps extracted from the pool4 layer of the VGGFace model.

b) Crucial region generation module

In order to recognize facial expressions from relationships between crucial regions and facial expressions, the crucial region generation is a key and fundamental step. Since discriminative features of facial expression usually appear in some specific facial regions, to avoid unrepresentative regions, we utilize crucial region generation module to extract representative crucial regions for our ReCNN based on motions of facial muscles.

The Facial Action Coding System (FACS) (Ekman et al., 2002; Friesen & Ekman, 1978) is a well-known taxonomy of human facial expressions. FACS defines 44 Action Units (AUs) (see Figure 4.4) which are the basic motions of single or groups of muscles. According to FACS, each specific facial expression can be expressed by a set of muscle contractions of AUs. The AU examples for six basic facial expressions is shown in Table 4.1. From Table 4.1, we can find that for six basic facial expressions most AUs happen around the eye and mouth region which means that these two regions have a major contribution to the facial expressions. Moreover, some facial expressions have totally different AUs combination with others. For example, happy can be expressed with a combination of cheek raise (AU6) and lip corner puller (AU12). Therefore, muscles around the eye and mouth form the crucial regions of facial expressions.

![Figure 4.4: Examples of facial action units [8.]](image-url)
Chapter 4: Relation-Aware Facial Expression Recognition

Table 4.1: List of AUs involved in six basic facial expressions.

<table>
<thead>
<tr>
<th>Facial Expression</th>
<th>Action Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Happy</td>
<td>AU6+AU12</td>
</tr>
<tr>
<td>Sad</td>
<td>AU1+AU4+AU15</td>
</tr>
<tr>
<td>Surprise</td>
<td>AU1+AU2+AU5+AU26</td>
</tr>
<tr>
<td>Disgust</td>
<td>AU9+AU15+AU16</td>
</tr>
<tr>
<td>Anger</td>
<td>AU4+AU5+AU7+AU23</td>
</tr>
<tr>
<td>Fear</td>
<td>AU1+AU2+AU4+AU5+AU7+AU20+AU26</td>
</tr>
</tbody>
</table>

To utilize the eye region and mouth region as crucial regions in our proposed ReCNN, it needs to design a strategy to automatically locate and obtain the proper eye and mouth region since too small or large regions can lead to less discriminative features or more redundant information from facial regions. Different from the facial landmark-based region generation scheme in (Y. Li et al., 2019), we utilize a fixed position cropping strategy in our crucial region generation module to extract the eye region and mouth region.

The fixed position cropping strategy mainly has three advantages. Firstly, according to FACS, the crucial region generation module focuses on the most important and salient regions, namely eye and mouth, and reduces the interferences from other unrepresentative facial regions. Secondly, the fixed position cropping is a simple but effective strategy. This is because the facial images can be easily detected and aligned by the recently advanced face detection and alignment methods which make it easy to locate and obtain proper eye region and mouth region with a fixed position cropping. And these two fixed regions located on the upper face and lower face usually contain the eye and mouth in most cases. Compared with (Y. Li et al., 2019), the proposed method can extract the crucial facial regions on two fixed positions rather than rely on detected landmarks. And it can also reduce the complexity and size of the training model but achieve high performance. Finally, the proposed method conducts on the feature maps of the
Figure 4.5: The example of generating crucial regions (the facial image is from the AffectNet (Mollahosseini et al., 2017)). (a) illustrates all the obtained 21 facial patches. (b) shows 4 boundaries that we take to cover the facial region around the eye. (c) donates 5 facial patches that we select from all 21 facial patches to take as the mouth region. (d) and (e) display the selected eye and mouth region respectively.

whole face rather other than on the input image, which is not sensitive to crucial regions misalignment (Y. Li et al., 2019).

Figure 4.5 shows an example of generating crucial regions. The details of the strategy are described as follows.

a) We divide the whole face into 21 facial patches. Specifically, the width and height of the whole face are divided into seven and five equal parts respectively (shown in Figure 4.5a).

b) The horizontal midlines of $H_1$ and $H_2$ are considered as the upside and bottom boundaries of eye region and the vertical midlines of $W_2$ and $W_6$ represent the left and right boundaries of eye region (see red line in Figure 4.5b). Then, the rectangle region consists of above four red lines is taken as an eye region, shown in Figure 4.5d.
c) For the mouth region, we select 5 facial patches from all 21 facial patches of the whole face to cover the mouth region. The selected facial patches are $H_3W_2, H_3W_3, H_3W_4, H_3W_5, H_3W_6$ (see green line in Figure 4.5c).

Using a fixed position cropping strategy, the crucial region generation module can automatically extract the crucial regions from the whole face in a fixed position and scale. To reduce the size of the training model and enlarge the receptive fields, we extract the crucial region of the eye and mouth on the feature maps from the feature extraction module (see Figure 4.3) as in (Y. Li et al., 2019), which can obtain sub-feature maps of the eye and mouth region. These sub-feature maps are then fed into two convolutional layers. To ensure the performance, we use both the sub-feature maps of the eye and mouth region to represent the input facial image and the global feature maps of the whole face which can provide complementary information to maintain the performance. As shown in Figure 4.3, we use three convolutional layers and one Maxpooling layer to encode the global feature maps $M_G$. The feature map set $M$ which is used to represent the input facial image is thus given by

$$M = \{M_1, M_2, M_3\} \quad (4-2)$$

where $M_1$ and $M_2$ are sub-feature maps of the eye and mouth region respectively, each with a size of $512\times10\times5$; $M_3$ is the global feature maps of the whole face which the size is $512\times7\times7$.

c) Relation module

In order to discover and utilize the relationship between crucial regions and facial expressions, our ReCNN uses a two-level relation module which mainly consists of Relation Extraction Net as illustrated in Figure 4.3. Inspired by (Y. Li et al., 2019), the first-level relation module is designed to compute first-level relation weights of the regions and obtain a coarse global representation. Then, the second-level relation module further computes the second-level relation weights and obtains a refined final representation by using the results from the first-level relation module.
First-level relation module. In the first-level relation module, both of the feature maps of the whole face, eye and mouth are processed in two branches (see Figure 4.3). In the first branch, we use a fully connected layer and a ReLU function to encode the input feature maps to the unweighted feature vectors. The second branch uses a Relation Extraction Net (yellow dashed rectangle in Figure 4.3) which applies two fully connected layers, a ReLU function and a Sigmoid function to compute the relation weights reflecting the importance of the region to facial expression. Then, the unweighted feature vectors are weighed by the relation weights. Different from the concatenation operation in (Y. Li et al., 2019), we finally aggregate all weighed feature vectors to a coarse global representation.

Mathematically, for instance, \( M_i \in M \) is the feature maps of the \( i \)-th region which are extracted from the crucial region generation module. Taking the feature maps \( M_i \) as the input of the first-level relation module, we firstly flatten \( M_i \) to a vector \( v_i \) which can be represented as follows:

\[
v_i = \text{Reshape}(M_i)
\]

(4-3)

In the first branch, we take \( v_i \) as input and use a fully connected layer as well as a ReLU function to obtain the unweighted feature vector which can be computed as:

\[
f_i = \text{ReLU}(v_i W_1 + b_1)
\]

(4-4)

where \( W_1 \) and \( b_1 \) denote the network’s weight matrix and bias vector in this fully connected layer; \( f_i \) is an unweighted 512-dimensional feature vector as shown in Figure 4.3.

In the second branch, a Relation Extraction Net applies two fully connected layers, a ReLU function and a Sigmoid function to compute the first-level relation weight of the \( i \)-th region which is defined as:

\[
r_i = \text{Sigmoid}(W_3(\text{ReLU}(v_i W_2 + b_2)) + b_3)
\]

(4-5)

where \( \{W_2, b_2\} \) and \( \{W_3, b_3\} \) are a set of parameters (weight matrix and bias vector) of two fully connected layers in order respectively; \( r_i \) denotes the first-
level relation weight which ranges from 0 to 1 because of the Sigmoid function. If the relations score is 1, it means that the region has a strong relationship and great importance to the facial expression. And the value 0 represents the region is completely unimportant.

Once the relation weight is computed, the first-level relation module utilizes the first-level relation weight $r_i$ to weight the feature vector $f_i$ which can obtain a weighted feature vector of the $i$-th region as follows:

\[ F_i = r_i \cdot f_i \]  \hspace{1cm} (4-6)

where $F_i$ is the weighted feature vector of the $i$-th region; $\cdot$ means the inner production.

Finally, different from the concatenation operation in (Y. Li et al., 2019), we aggregate all weighed feature vectors of three regions including eye, mouth and whole face to a coarse global representation. The obtained global representation is given by,

\[ F = Add(F_1, F_2, F_3) \]  \hspace{1cm} (4-7)

where $F$ is the global representation with a size of 512 dimensions.

**Second-level relation module.** To obtain the refined final representation, the second-level relation module (shown in Figure 4.3) utilizes the relationship between unweighted feature vector of the region and the global representation which are both obtained from the first-level relation module.

With an unweighted 512-dimensional feature vector $f_i$ of the $i$-th region and the global representation $F$, the second-level relation module firstly takes a concatenation operation as follows:

\[ f_i' = Concat(f_i, F) \]  \hspace{1cm} (4-8)

where $f_i'$ is the new feature vector with a size of 1024 dimensions as shown in Figure 4.3.

Then, we use a Relation Extraction Net as in the first relation module to compute the second-level relation weight $r_i'$ between the feature vector of the $i$-th
region and the global representation. Therefore, the final relation weight \( R_i \) can be represented as the product of priori first-level relation weight \( r_i \) and second-level relation weight \( r_i' \). In the following stage, we generate a new weighted feature vector of \( i \)-th region utilizing an inner production operation with the final relation weight \( R_i \). The new weighted feature vector is defined as:

\[
F_i' = R_i \cdot f_i' = r_i r_i' \cdot f_i'
\]  

(4-9)

Finally, we take a concatenation operation for all weighted vectors of eye, mouth and whole face to generate the refined final representation. The obtained final representation is given by,

\[
\tilde{F} = Cncat (F_1', F_2', F_3')
\]  

(4-10)

where \( \tilde{F} \) represents the final representation with a size of 3072 dimensions which can be fed into the classifier to classify and recognize facial expressions.

### 4.3.3 Loss Function

The tasks of Facial expression recognition can be considered as a multi-class classification problem which usually utilize Categorical Cross Entropy (CCE) loss function in most existing works. However, the phenomenon of data imbalance commonly appears in most existing databases due to the nature of expressions in the real world, which can lead to poor performance on the less represented category. Therefore, to overcome the influence of imbalance problem and improve the recognition rate, we modify the CCE loss function by assigning higher priority and weight to the less represented categories. The weighted CCE loss function is given by,

\[
loss_{WCCE} = -\sum_{i=1}^{N} w_i g_i log(p_i)
\]  

(4-11)

where \( N \) is the number of training images; \( p_i \) and \( g_i \) are the labels of prediction and ground truth; \( w_i \) is the weight of the class which ground truth \( g_i \) belongs to. We define the weight \( w_i \) as:
where $n_i$ is the number of images in the category which ground truth $g_i$ belongs to; $n_c$ represents the number of categories in the training set. When the data is balanced, the weighted loss function is equal to the conventional CCE loss function. During the training, compared with other categories, this loss function will highly penalize the wrong classification results from less represented categories.

### 4.4 Experiment

In this section, we firstly give a description of experimental settings including the databases and some implementation details. We then evaluate our proposed ReCNN on the databases including basic and compound facial expressions and compare with other state-of-the-art approaches. Finally, we provide the analysis of ablation experiment.

#### 4.4.1 Databases

To evaluate our approach, two large in-the-wild facial expression databases are used in our experiment, namely AffectNet (Mollahosseini et al., 2017) and RAF-DB (S. Li & Deng, 2019; S. Li et al., 2017). These two databases are challenging since they provide large-scale of facial images from real-world scenarios. The sample images from AffectNet and RAF-DB databases are illustrated in Figure 4.6 and their details are provided in Table 4.2.

**AffectNet**: The AffectNet database which provides over 1M in the wild facial images is one of the largest databases in the research field of facial affective computing. About 440,000 facial images have been manually annotated with the
Chapter 4: Relation-Aware Facial Expression Recognition

Figure 4.6: Examples of facial expression images from the databases. From left to right: (a) AffectNet, (b) RAF-DB.

Table 4.2: The details of database for facial expression recognition including the expression categories, training and test samples.

<table>
<thead>
<tr>
<th>Database</th>
<th>AffectNet</th>
<th>RAFF-DB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Training</td>
<td>Test</td>
</tr>
<tr>
<td>Neutral</td>
<td>74,874</td>
<td>500</td>
</tr>
<tr>
<td>Happy</td>
<td>134,415</td>
<td>500</td>
</tr>
<tr>
<td>Sad</td>
<td>25,459</td>
<td>500</td>
</tr>
<tr>
<td>Surprise</td>
<td>14,090</td>
<td>500</td>
</tr>
<tr>
<td>Fear</td>
<td>6,378</td>
<td>500</td>
</tr>
<tr>
<td>Disgust</td>
<td>3,803</td>
<td>500</td>
</tr>
<tr>
<td>Anger</td>
<td>24,882</td>
<td>500</td>
</tr>
<tr>
<td>Total</td>
<td>283,901</td>
<td>3500</td>
</tr>
</tbody>
</table>
category or intensity of facial expression. In our experiment, as shown in Table 4.2, we use the images with 7 facial expression categories as the training set which contains about 283,901 samples. We conduct the experiment on the validation set of the AffectNet database to evaluate the proposed method due to the test set is still unavailable. The validation set totally includes about 3,500 facial images, 500 samples for each category.

**RAF-DB:** RAF-DB database includes seven basic facial expressions and twelve compound facial expressions, which includes around 30,000 facial images in total. It is a large-scale database since it is collected from the wild environment with significant changes in subjects’ attributes, illumination, occlusions, etc. As shown in Table 4.2, the facial images with 7 categories of facial expressions are used in our experiment, which contain about 12,271 facial images in training set and 3,068 facial images in test set.

### 4.4.2 Implementation Details

During the stage of preprocessing, all the images were first roughly aligned using similarity transformation according to the annotated facial landmarks from the database. According to the annotated face bounding box from the database, we then crop these images and resize all the cropped images to the size of 224×224px. We implemented the network structure of proposed ReCNN illustrated in Figure 4.3 using the TensorFlow framework. For ReCNN, to improve performance and reduce computational complexity, we used pre-trained VGGFace (Parkhi et al., 2015) as the backbone network in the feature extraction module since its excellent performance in face recognition. The first ten convolutional layers were chosen as the feature maps extractor. And then the extracted feature maps were fed into the crucial region generation module to extract the sub-feature maps of two crucial regions (eye and mouth) and the feature maps of the whole face with a fixed position cropping strategy.

The proposed ReCNN was trained through an 8 GB NVIDIA GeForce GTX 1080 GPU on a desktop PC with a 4.20GHz Intel Core i7 processor, 16 GB of
RAM memory. Since the phenomenon of data imbalance appears in the training samples of databases (shown in Table. 2), the weighted Categorical Cross Entropy loss function were used in our experiment which can overcome the influence of data imbalance and improve the recognition rate. The stochastic gradient descent (SGD) method was utilized to optimize the whole network. During the training process, the momentum of SGD was set as 0.9. For the learning rate, we initially set the learning rate as 0.01 and reduced it by step decay policy which divided the learning rate by 10 after each 20K iterations. The batch size was set as 64 and the total iterations in the training stage was 50K. It took about 48 hours to complete the training of the network. During the evaluation stage, the metric of accuracy which commonly used in the evaluation of the classification task was used to evaluate the proposed method.

4.4.3 Experiment on AffectNet Database

To evaluate our proposed ReCNN, seven facial expressions (Neutral, Happy, Sad, Surprise, Fear, Disgust and Anger) in the AffectNet validation set which includes 500 samples for each category were used. The examples of seven facial expressions are shown in Figure 4.6. Figure 4.7 illustrates the confusion matrix based on ReCNN for facial expression recognition and the accuracy for each category of facial expressions by using our method. The accuracy of Happy is about 87.6% which is the highest among the seven facial expressions. There are two facial expressions (Disgust and Anger) with the lowest accuracies under 60% which are 54.4% and 59.0% respectively. The accuracy of other facial expressions is Neutral (62.4%), Sad (59.4%), Surprise (60.0%), Fear (65.6%) respectively. The average accuracy of the proposed ReCNN on the AffectNet validation set is about 64.06%.
Chapter 4: Relation-Aware Facial Expression Recognition

Figure 4.7: The confusion matrix of the proposed method on the AffectNet database.

Table 4.3: Comparison of accuracies (%) using different methods on AffectNet. The highest results are highlighted in bold.

<table>
<thead>
<tr>
<th>Method</th>
<th>Year</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simonyan et al.</td>
<td>2014</td>
<td>51.11</td>
</tr>
<tr>
<td>Zhao et al.</td>
<td>2017</td>
<td>54.47</td>
</tr>
<tr>
<td>Yu et al.</td>
<td>2018</td>
<td>52.97</td>
</tr>
<tr>
<td>Li et al.</td>
<td>2018</td>
<td>55.33</td>
</tr>
<tr>
<td>Li et al.</td>
<td>2019</td>
<td>58.78</td>
</tr>
<tr>
<td>Hua et al.</td>
<td>2019</td>
<td>62.11</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td>2021</td>
<td><strong>64.06</strong></td>
</tr>
</tbody>
</table>

The further evaluation is to compare the performance of proposed ReCNN with the current state-of-the-art approaches including VGG16 (Simonyan & Zisserman, 2014), DLP-CNN (L. Zhao, Li, Zhuang, & Wang, 2017), GAN-Inpainting (J. Yu et al., 2018), pACNN (Y. Li, Zeng, Shan, & Chen, 2018), and gACNN (Y. Li et al., 2019), and ensemble CNN (Hua, Dai, Huang, Xiong, & Gui, 2019). Table 4.3 illustrates the comparison of accuracies using different methods on the AffectNet. The last column in Table 4.3 shows the average accuracy of our method. As shown
in Table 4.3, it is obvious that the proposed method has superior performance in terms of recognition accuracy compared with state-of-the-art methods with a 12.95% to 1.95% improvement in seven categories.

4.4.4 Experiment on RAF-DB Database

The RAF-DB database including 3,068 facial image samples with six basic facial expressions and neutral was also used to evaluate the proposed ReCNN. Figure 4.6 shows samples of seven facial expressions in RAF-DB database. We illustrated the confusion matrix of the proposed method on the RAF-DB database in Figure 4.8. The confusion matrix shows that our ReCNN achieves the average accuracy of 87.06%. Moreover, the accuracy of Happiness can reach 94.35% which is the most easily recognized among all facial expressions. These two expressions with lowest accuracy in seven facial expressions are Fear (64.86%) and Disgust (58.75%). Additionally, the accuracy of other facial expressions is 86.93% (Surprise), 85.36% (Sadness), 79.01% (Anger) and 86.62% (Neutral) respectively.

![The confusion matrix of the proposed method on the RAF-DB database.](image)

Figure 4.8: The confusion matrix of the proposed method on the RAF-DB database.
Table 4.4: Comparison of accuracies (%) using different methods on RAF-DB. The highest results are highlighted in bold.

<table>
<thead>
<tr>
<th>Method</th>
<th>Year</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simonyan et al.</td>
<td>2014</td>
<td>80.96</td>
</tr>
<tr>
<td>Zhao et al.</td>
<td>2017</td>
<td>80.89</td>
</tr>
<tr>
<td>Li et al.</td>
<td>2017</td>
<td>82.84</td>
</tr>
<tr>
<td>Liu et al.</td>
<td>2017</td>
<td>79.70</td>
</tr>
<tr>
<td>Kuo et al.</td>
<td>2018</td>
<td>72.21</td>
</tr>
<tr>
<td>Yu et al.</td>
<td>2018</td>
<td>81.87</td>
</tr>
<tr>
<td>Zhao et al.</td>
<td>2018</td>
<td>81.10</td>
</tr>
<tr>
<td>Li et al.</td>
<td>2018</td>
<td>83.27</td>
</tr>
<tr>
<td>Fan et al.</td>
<td>2018</td>
<td>82.63</td>
</tr>
<tr>
<td>Liu et al.</td>
<td>2018</td>
<td>83.86</td>
</tr>
<tr>
<td>Lian et al.</td>
<td>2019</td>
<td>82.69</td>
</tr>
<tr>
<td>Li et al.</td>
<td>2019</td>
<td>84.13</td>
</tr>
<tr>
<td>Florea et al.</td>
<td>2019</td>
<td>84.50</td>
</tr>
<tr>
<td>Li et al.</td>
<td>2019</td>
<td>85.17</td>
</tr>
<tr>
<td>Li et al.</td>
<td>2019</td>
<td>85.07</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td><strong>2021</strong></td>
<td><strong>87.06</strong></td>
</tr>
</tbody>
</table>

We also compared the proposed method with state-of-the-art approaches (Fan, Lam, & Li, 2018; Florea, Florea, Badea, Vertan, & Racoviteanu, 2019; Kuo, Lai, & Sarkis, 2018; S. Li & Deng, 2019; S. Li et al., 2017; Y. Li et al., 2018, 2019; Z. Li et al., 2019; Lian, Li, Tao, Huang, & Niu, 2019; C. Liu, Tang, Lv, & Wang, 2018; Z. Liu, Li, & Deng, 2017; Simonyan & Zisserman, 2014; J. Yu et al., 2018; L. Zhao et al., 2017; S. Zhao, Cai, Liu, Zhang, & Chen, 2018) on the RAF-DB database (see Table 4.4). Since few state-of-the-art approaches used the confusion matrix diagonal average value as the evaluation metric, to achieve a fair comparison, we converted them to average accuracy like in most existing works. Li et al. (S. Li & Deng, 2019) proposed a DLP-CNN network for facial expressions recognition which achieved an accuracy of 84.13% on average. The higher
accuracies of FER were achieved with gACNN (Y. Li et al., 2019) and APM-VGG (Z. Li et al., 2019), which are 85.07% and 85.17%, respectively. Compared with above approaches for facial expression recognition, our method has a superior recognition accuracy which is 87.06% on average.

4.4.5 Experiment on Compound Facial Expressions

To further investigate the proposed ReCNN, we evaluated its performance of recognizing compound facial expression on the RAF-DB database. The RAF-DB database also includes twelve compound facial expressions: Happily Surprised (HS), Happily Disgusted (HD), Sadly Fearful (SF), Sadly Angry (SA), Sadly Surprised (SS), Sadly Disgusted (SD), Fearfully Angry (FA), Fearfully Surprised (FS), Angrily Surprised (AS), Angrily Disgusted (AD), Disgustedly Surprised (DS) and Fearfully Disgusted (FD). Figure 4.9 shows the examples of compound facial expressions from RAF-DB database.

RAF-DB database totally contains 792 test samples of compound facial expressions. For compound facial expression recognition, we discarded the facial expression of fearfully disgusted during the training process since it only has 8 images which are too few compared with others. We used eleven categories of compound facial expressions in the training stage, 3,162 images in total. To the best of our knowledge, compound facial expression recognition that we implement is a rare work on the latest RAF-DB database. It is because that the compound
Table 4.5: Comparison of accuracies (%) using different methods on compound facial expressions from RAF-DB. The highest results are highlighted in bold.

<table>
<thead>
<tr>
<th>Method</th>
<th>Average</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM+HOG</td>
<td>33.65</td>
<td>51.89</td>
</tr>
<tr>
<td>SVM+Gabor</td>
<td>35.76</td>
<td>53.54</td>
</tr>
<tr>
<td>baseDCNN</td>
<td>40.17</td>
<td>56.69</td>
</tr>
<tr>
<td>DLP-CNN</td>
<td>44.55</td>
<td>57.95</td>
</tr>
<tr>
<td>ReCNN (Ours)</td>
<td>46.08</td>
<td>61.62</td>
</tr>
</tbody>
</table>

Table 4.6: Comparison of accuracies (%) using different methods on eleven compound facial expressions from RAF-DB. The highest results are highlighted in bold.

<table>
<thead>
<tr>
<th></th>
<th>HS</th>
<th>HD</th>
<th>SF</th>
<th>SA</th>
<th>SS</th>
<th>SD</th>
<th>FA</th>
<th>FS</th>
<th>AS</th>
<th>AD</th>
<th>DS</th>
<th>Avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLP-CNN</td>
<td>79.3</td>
<td>38.3</td>
<td>31.8</td>
<td>30.3</td>
<td>22.2</td>
<td>64.5</td>
<td>63.6</td>
<td>62.1</td>
<td>28.9</td>
<td>60.3</td>
<td>8.8</td>
<td>44.55</td>
</tr>
<tr>
<td>ReCNN (Ours)</td>
<td>77.78</td>
<td>38.3</td>
<td>18.8</td>
<td>27.27</td>
<td>16.67</td>
<td>69.5</td>
<td>66.67</td>
<td>70.69</td>
<td>23.68</td>
<td>74.71</td>
<td>22.86</td>
<td>46.08</td>
</tr>
</tbody>
</table>

Facial expressions are more complex and more difficult to be detected compared with the basic facial expressions. Moreover, it still remains a great challenge due to the lack of training data. Therefore, we only compared with four baseline methods from RAF-DB database reported in (S. Li & Deng, 2019) including conventional hand-crafted feature methods and deep learning methods. We used both confusion matrix diagonal average value of 11 compound facial expressions and regular accuracy for a fair comparison.

Table 4.5 shows the comparison results on the RAF-DB database between the proposed ReCNN and other approaches. From the Table 4.5, we can conclude observations as follows. First, among the baseline methods, the best classification accuracy is 57.95% for DLP-CNN. And the accuracy decreases to 44.55% when the evaluation metric is the confusion matrix diagonal average value. Second, the proposed ReCNN achieves the best performance among these methods with a classification accuracy of 61.62% and an average accuracy of 46.08%, which are 3.67% and 1.53% higher than DLP-CNN, respectively. We also provided the
accuracies of eleven compound facial expressions and their comparison with DLP-CNN in Table 4.6. Compared with DLP-CNN, the proposed ReCNN outperforms DLP-CNN in average accuracy and five categories of facial expressions. The results indicate that our proposed ReCNN is also applicable for compound facial expression recognition.

4.4.6 Ablation Experiment

To investigate the relationship between crucial facial regions and facial expressions and better understand our proposed ReCNN, we conducted a quantitative analysis of our method for facial expression recognition through the ablation experiment. The ablation experiment was validated on the AffectNet database.

Effect of different crucial regions in ReCNN. We first evaluated the effect of each crucial region in our proposed ReCNN. Table 4.7 shows that: (1) The lowest accuracy is about 62.21% when recognizing facial expressions only focusing on the whole face. (2) From Table 4.7, it is obvious that when applying eye region or mouth region strategy, the performance is better than that of only using the whole face. It demonstrates that there are certain relationships between crucial regions and facial expressions that can reflect the representativeness among individual facial expressions. Moreover, we can see that they have a similar performance on the recognition rates which is 62.82% and 62.91%, respectively. One possible reason is that although the crucial region can further improve the performance, according to FACS, some facial expressions share the same AUs in the eye region and mouth region which is not easy to distinguish them by using AUs from only one region. (3) The combination of three parts including the whole face, eye and mouth region gives further performance boost with an accuracy of 64.06%. The experimental results are consistent with FACS since using a combination of AUs from the eye and mouth region can express the specific facial expressions. It shows that the proposed ReCNN is capable of discovering and utilizing the relationships
Chapter 4: Relation-Aware Facial Expression Recognition

Table 4.7: Evaluation of effect of different crucial regions in RECNN on AffectNet database. The highest results are highlighted in bold.

<table>
<thead>
<tr>
<th>Accuracy</th>
<th>Face</th>
<th>Eye region</th>
<th>Mouth region</th>
</tr>
</thead>
<tbody>
<tr>
<td>62.21</td>
<td>√</td>
<td></td>
<td></td>
</tr>
<tr>
<td>62.82</td>
<td>√</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>62.91</td>
<td>√</td>
<td></td>
<td>√</td>
</tr>
<tr>
<td><strong>64.06</strong></td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
</tbody>
</table>

Table 4.8: Evaluation of effect of attention modules and weighted loss function in RECNN on AffectNet database. The highest results are highlighted in bold.

<table>
<thead>
<tr>
<th>Accuracy</th>
<th>1st-level relation</th>
<th>2nd-level relation</th>
<th>Weighted Loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>62.21</td>
<td>√</td>
<td></td>
<td>√</td>
</tr>
<tr>
<td>63.60</td>
<td>√</td>
<td></td>
<td>√</td>
</tr>
<tr>
<td>58.50</td>
<td>√</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td><strong>64.06</strong></td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
</tbody>
</table>

between the crucial regions and facial expressions to further improve the facial expression recognition performance.

Effect of relation modules and weighted loss function in ReCNN. We further evaluated the effect of relation modules and weighted loss function in our proposed ReCNN. Table 4.8 shows the results of ablation experiment. We first study the effect of relation modules in ReCNN using a weighted loss function. Table 4.8 shows three kinds of network structure: (1) ReCNN without relation modules; (2) ReCNN with the 1st-level relation module; (3) ReCNN with the combination of 1st-level and 2nd-level relation module. From Table 4.8, it shows that the network without relation modules achieves the lowest accuracy of 62.21% among the three kinds of network structure. Combining with the 1st-level relation module, the network has an improvement of 1.39% in the recognition rate. After adding the 2nd-level relation module, the performance of the network can be further boosted.
with an accuracy of 64.06%. This indicates that the 2nd-level relation module can further obtain a refined final representation using the results from 1st-level relation module. We have also evaluated the effect of the weighted loss function in ReCNN. From Table 4.8, we can see that the weighted loss function can lead to a significant improvement in facial expression recognition performance. And the performance of the network without the weighted loss function significantly declines to 58.50% on the AffectNet database. Since there is data imbalance in terms of the emotional category in the AffectNet database, we conclude that the weighted loss function can effectively overcome the influence of data imbalance.

4.5 Summary

In this chapter, we propose a novel relation-aware method for facial expression recognition called Relation Convolutional Neural Network (ReCNN). The proposed ReCNN is an end-to-end architecture which can adaptively capture the relationships between crucial regions and facial expressions and focus on the most discriminative regions to generate the discriminative representation of facial expressions based on the relationship. ReCNN has superior recognition performance compared with state-of-the-art approaches on the in-the-wild databases including the basic facial expressions and compound facial expressions. Ablation experiments show the relationship between crucial regions and facial expressions can significantly improve the recognition performance.
Chapter 5

Local and Global Perception Generative Adversarial Network for Facial Expression Synthesis

5.1 Introduction

Facial expression recognition has a wide range of applications in various domains such as healthcare, virtual reality (VR), augmented reality (AR), driver assistant systems and entertainment (Guo et al., 2020; Lian et al., 2020; Lou et al., 2019; X. Sun & Lv, 2019; Yiming Wang et al., 2021; S. Zhang, Yu, Wang, Dong, & Pham, 2021). The performance of most existing methods, especially deep learning-based methods, heavily relies on the quantity and quality of training data. However, collecting a large annotated facial expression database is often limited by the resources we can find. And it also usually requires professional expertise as well as time consuming and expensive. Therefore, how to effectively generate the required facial data with minimum costs has been an interesting research topic in recent years.

Facial expression synthesis is one of the most successful solutions to the problem of insufficient data. The most notable method for facial expression synthesis is Generative Adversarial Networks (GANs), which can effectively generate the new facial expression of an input facial image according to the target expression label. Whilst GANs have achieved impressive results on the task of facial expression synthesis, existing methods still suffer from some limitations. Most existing GAN-based methods are designed for general face synthesis tasks without considering the characteristics of facial expressions, which are not
appropriate to synthesize facial expressions. In fact, according to psychology research such as Facial Action Coding System (FACS) (Ekman et al., 2002; Friesen & Ekman, 1978), it is obvious that the differences between facial expressions usually locate in some certain crucial regions such as eye and mouth. Moreover, studies have shown that the attention naturally focuses on specific facial regions when humans recognize and distinguish different facial expression (Wegrzyn et al., 2017). For example, eyes play an important role for fear analysis while the mouth is vital for recognizing happiness. However, previous GAN-based methods mainly focus on the face as a whole but these local facial parts have been significantly overlooked for facial expression synthesis, which leads to overlapping and blur in the local facial regions of generated results. With this observation, we investigate and utilize features of local facial regions in the task of facial expression synthesis.

In this chapter, we propose a novel end-to-end facial expression synthesis method called Local and Global Perception Generative Adversarial Network (LGP-GAN) by integrating local and global facial information to synthesize facial expression. LGP-GAN is a two-stage cascaded architecture that decomposes the process of facial expression synthesis. Stage I utilizes local networks to generate local facial regions. Stage II uses a global network to generate final facial expressions using the output of the Stage I as input. The local network pays special attention to the most discriminative regions such as eye and mouth while the global network mainly focuses on the whole facial structure. In summary, the contributions of this chapter are summarized as follows:

- Inspired by the observation that the differences between facial expressions mainly occur in some crucial facial regions, we propose a novel end-to-end facial expression method LGP-GAN with the local network capturing texture details of these crucial facial regions while the global network learning the general structure and profile of the face.
- The proposed LGP-GAN has a two-stage cascaded architecture that divides the facial expression synthesis process into local facial region generation
and global facial image generation. It can fully utilize both the local and global facial information in the process of facial expression synthesis, which can synthesize facial expressions step by step.

- We have explored the role of crucial facial regions in facial expression synthesis. The qualitative and quantitative evaluation on the public database of facial expression shows that the proposed LGP-GAN has superiority over the state-of-the-art methods. It also shows the importance of crucial facial regions in facial expression synthesis, which can further improve the performance.

5.2 Related Work

In recent years, the technologies of machine learning and deep learning has achieved impressive performance in various fields such as computer vision and pattern recognition (X. Bi, Zhao, Huang, Chen, & Ma, 2020; L. Chen et al., 2018; Ha et al., 2018; B. Hu & Wang, 2020; F.-Q. Liu & Wang, 2020; Q. Liu, Li, Hu, & Gu, 2020; W. Liu et al., 2021; X. Liu et al., 2020; Ming, Meng, Fan, & Yu, 2021; Xia et al., 2019; N. Zeng et al., 2020). One of the deep learning methods, Generative Adversarial Networks (GANs) (Goodfellow et al., 2014; K. Wang et al., 2017) designed according to the game theory have attracted increasing attention recently. The typical GAN consists of a generator and a discriminator. The generator generates fake images making the generated images as realistic as possible. On the other hand, the discriminator learns to distinguish the authenticity of the generated images. Both generator and discriminator are simultaneously trained in the same framework, which is so-called adversarial learning. After interactive confrontations, the generator can finally generate indistinguishable images resembling the real images. Based on classical GANs, many variants of GANs have been developed to further improve performance such as DCGAN (Radford, Metz, & Chintala, 2015), CGAN (Mirza & Osindero, 2014) and WGAN
Chapter 5: Local and Global Perception Generative Adversarial Network for Facial Expression Synthesis

(Arjovsky, Chintala, & Bottou, 2017; Gulrajani, Ahmed, Arjovsky, Dumoulin, & Courville, 2017). So far, GANs have become one of the most notable generative models have been applied to various computer vision tasks such as image-to-image translation (Isola, Zhu, Zhou, & Efros, 2017; K. Liao, Lin, Zhao, & Gabbouj, 2019; Pang, Xie, & Li, 2018; H. Zhang, Sindagi, & Patel, 2019; J.-Y. Zhu, Park, Isola, & Efros, 2017), image synthesis (Y. Chen et al., 2020; Ruffino, Hérault, Laloy, & Gasso, 2020; Z. Wang, Healy, Smeaton, & Ward, 2020; Yuan & Peng, 2019) and image inpainting (Minyu Chen, Liu, Ye, & Wang, 2020; Xu, Zeng, Li, & Huang, 2020). In particular, recent advances in GANs have achieved remarkable results in the facial attribute editing task (Dogan & Keles, 2020; M. Liu et al., 2019; Wu, Lin, Chang, Chang, & Liao, 2019; K. Zhang, Su, Guo, Qi, & Zhao, 2020) which is one of the most successful applications of GANs. StarGAN (Y. Choi et al., 2018) and AttGAN (Z. He, Zuo, Kan, Shan, & Chen, 2019) are two representative methods in facial attribute editing. StarGAN takes the facial image and the target facial attribute as input to complete the facial attribute editing task only using one generator and one discriminator. AttGAN is an encoder-decoder architecture, which is very similar to StarGAN in terms of facial attribute editing but uses the latent representation to represent the facial attribute.

Facial expression synthesis can be treated as a subproblem of facial attribute editing which has been a hot research topic in various research fields, especially for facial analysis. Previous works just treat this task as a general image-to-image translation in which the facial expressions are considered as special facial attributes. Some existing methods (Y. Choi et al., 2018; Z. He et al., 2019; M. Liu et al., 2019; Wu et al., 2019) handle the task of facial expression synthesis by modifying relevant facial attributes such as smiling, mouth open and mouth closed. For instance, StarGAN (Y. Choi et al., 2018) uses a single generator to synthesize new facial expression of the input facial image according to given desired labels of facial expressions. However, these methods are not appropriate to generalize the facial expression synthesis task due to various facial deformations of facial expressions. In fact, facial expression synthesis is a more complex task compared
with facial attribute editing, which usually has large transformations in facial regions. Recent advances in GANs specially designed for facial expression synthesis have shown impressive results and drawn prevalent attention. Zhou et al. (Y. Zhou & Shi, 2017) proposed a conditional difference adversarial autoencoder (CDAAE) to generate facial images with desired emotion states. Chen et al. (M. Chen et al., 2018) proposed Double Encoder Conditional GAN (DECGAN) for facial expression synthesis. There are also some works that utilize geometry information to guide the facial expression generation such as GC-GAN (Qiao et al., 2018) and G2GAN (Song et al., 2018). Ding et al. (Ding et al., 2018) proposed ExprGAN which edited the facial expressions according to the controllable expression intensity. Pumarola et al. (Pumarola, Agudo, Martinez, Sanfeliu, & Moreno-Noguer, 2018) designed a GAN network structure based on StarGAN generating new facial expressions according to the given Action Units (AU) labels, which achieved impressive results on facial expression synthesis.

However, existing GAN-based methods mainly consider the face as a whole without paying special attention to local facial regions which leads to overlapping and blur in the local facial regions of generated results. In fact, research in physiology and psychology (Ekman et al., 2002; Friesen & Ekman, 1978; Wegrzyn et al., 2017) have shown that the differences between facial expressions often appear in crucial regions such as eye and mouth instead of evenly appearing in the whole face. Moreover, the importance of local facial regions has been attracted increasing attention from various research fields especially facial analysis in healthcare. For example, Liu et al. (X. Liu et al., 2020) proposed a hierarchy convolutional neural network for facial paralysis evaluation which can extract facial paralysis features from local facial regions and reduce the impact of redundant information. Inspired by this, the main aim of this work is to consider and utilize the crucial facial regions in the task of facial expression synthesis. In this chapter, we propose a Local and Global Perception Generative Adversarial Network (LGP-GAN) with a two-stage cascaded architecture. This kind of two-stage scheme has been proposed in previous works such as image segmentation,
for instance, Zhao et al. (N. Zhao, Tong, Ruan, & Sheng, 2019) proposed a two-stage network for pancreas segmentation, which first determines the candidate regions and then refine the segmentation on these regions. Different from (N. Zhao et al., 2019), we utilize two-stage scheme to fully utilize local and global facial information for facial expression synthesis in this chapter. The proposed LGP-GAN divides the facial expression synthesis process into two parts: local facial region generation and global facial image generation, which can synthesize facial expressions step by step.

5.3 Methodology

Figure 5.1: The overview of the LGP-GAN for facial expression synthesis. Our method is an end-to-end model with a two-stage cascaded structure consisting of two local networks and one global network: In Stage I, the two local generators generate the crucial regions of eyes and mouth, which learn the transformation of different local facial regions between different facial expressions; In Stage II, the global network is used to the global network is used to perceive and supplement the global facial information outside of the crucial regions, which can further refine the generated results.

In this section, we describe the details of the proposed LGP-GAN for facial expression synthesis. The proposed LGP-GAN is a two-stage cascaded architecture integrating local and global facial information to synthesize facial
expression step by step. It divides the facial expression synthesis process into two parts: local facial region generation and global facial image generation. The proposed LGP-GAN aims at generating a target expression of a given facial image while retaining identity properties, which can be used to solve the problem of insufficient training data in facial expression recognition and applied in various fields such as healthcare and entertainment. Specifically, we denote the input RGB facial image as $I_x$ with arbitrary facial expression. The facial expression image is characterized by an $n$-dimensional vector $f = [f^{(1)}, f^{(2)}, \ldots, f^{(n)}]^T$, where each attribute $f^{(i)}$ is a binary value (0 or 1) indicating the category of facial expressions, such as happy, sad, or surprise. Our objective is to transform the input image $I_x$ with facial expression $f_x$ to an output image $I_y$ which has the target facial expression $f_y$.

### 5.3.1 Network Architecture

As illustrated in Figure 5.1, the proposed LGP-GAN has a cascaded structure which includes two local networks and one global network. The input includes a facial image $I_x$ and a target facial expression label $f_y$. It outputs an image $I_y$ that displays the face from the input image with the emotion represented by $f_x$. The local networks capture the local texture details corresponding to the local facial regions of the eye and mouth while the global network is used to learn the whole facial information. The rationale of this design is that the differences different facial expressions often occur in crucial regions. Existing methods simply take the whole facial image as input to synthesize facial expressions without paying special attention to local facial regions, which leads to overlapping and blur around local facial regions such as eyes and mouths. One main reason is that local facial regions have fewer pixels than other regions and thus the network is prone to fit the regions outside of local facial regions during network training. In this chapter, the design of cascaded structure including two local networks and one global network can
help the network better learn features in different local regions of a facial image. The proposed cascaded structure can divide the facial expression synthesis into local facial region generation and global facial image generation and thus synthesizes facial expressions step by step.

**a) Preprocessing**

The Facial Action Coding System (FACS) (Ekman et al., 2002; Friesen & Ekman, 1978) is a well-known taxonomy of human facial expressions. FACS defines 44 Action Units (AUs) which represent the basic motions of single or groups of muscles. According to FACS, each specific facial expression can be expressed by a set of muscle contractions of AUs. And most AUs appear in the eye and mouth region of six basic facial expressions (Happy, Sad, Surprise, Fear, Disgust and Anger). For example, happy can be expressed with a combination of cheek raise (AU6) and lip corner puller (AU12). Therefore, the eye and mouth are the crucial regions of facial expressions, which have a major contribution to facial expressions. Inspired by this, we design two local networks to capture the local texture details of the crucial facial regions.
Facial expression synthesis can be treated as a subproblem of facial attribute editing which has been a hot research topic in various research fields, especially for facial analysis. Previous works just treat this task as a general image-to-image translation in which the facial expressions are considered as special facial attributes. To utilize these crucial facial regions in our proposed LGP-GAN, the first step is to locate appropriate eye and mouth regions since too small or large regions can lead to less discriminative information or more redundant information from facial regions. The simple and effective method is based on the facial landmarks since the facial images can be easily detected and aligned by the recently advanced face detection and alignment methods. We first detect the facial landmarks, and then use these landmarks to obtain two local facial regions (eyes and mouth) as shown in Figure 5.2. The advantage of this step is that it can focus on the most important and salient regions and thus reduces the interferences from other unrepresentative facial regions. Therefore, the whole face is divided into two parts based on the local facial regions. For a given facial image $I_x$, we can obtain the eye region $I_{eye}$, mouth region $I_{mouth}$ and the background region $I_{bg}$ outside of the crucial regions.

**b) Generator**

As shown in Figure 5.1, our generator $G$ has a two-stage cascaded architecture which consists of three basic generators: two local generators $G_{local} = \{G_{eye}, G_{mouth}\}$ and one global generator $G_{global}$. All three generators have similar architectures, but they are assigned with different learning tasks. We leverage two local generators to capture the deformation of crucial facial regions and one global generator to lean the whole facial texture. We borrow the idea of the basic architecture of the generators from (Y. Choi et al., 2018) which has proven to be successful for the task of the image-to-image translation, facial attribute editing and facial expression synthesis. The architecture of this kind of generators consists of a set of convolutional layers for downsampling, residual blocks and deconvolution layers for upsampling. The parameters of layers in each
generator such as filter size, kernel size and stride are set referring to (Y. Choi et al., 2018). Table 5.1 illustrates the details of the generator network architecture including the local generator and global generator. The input channel of each generator is $3 + n$ defined by the channel of the input RGB image and the dimension of labels such as the number of categories of facial expressions. In order to focus on the facial areas that need to accommodate the changes in the process of facial expression synthesis, we also utilize attention mechanisms in each generator as in (Pumarola et al., 2018). As shown in Table 5.1, we split the output layer of each generator into two parallel parts, one to generate the colour mask $C$ and the other to generate the attention mask $M$. The attention mask $M$ is a weight matrix calculated adaptively by the network when generating the facial expression. To utilize the attention mask $M$, the final output of the generator $I_{output}$ can be represented as:

$$I_{output} = (1 - M) \times C + M \times I_{input} \quad (5-1)$$

where $\times$ represents element-wise multiplication; $I_{input}$ is the input image.

In Stage I, the crucial regions of eyes $I_{eye}$ and mouth $I_{mouth}$ are concatenated with the target facial expression label $f_y$ as input to two local networks $G_{eye}$ and $G_{mouth}$. The local generators $G_{local}$ generate local facial regions $I_{local} = \{I_{eye}, I_{mouth}\}$, which learns the transformation of different local facial regions between different facial expressions; e.g., transforming mouth open to mouth closed. As shown in Table 5.1, each local generator consists of two downsampling convolutional layers, three bottleneck residual blocks, and two Deconvolutional layers. All layers are followed by Instance Normalisation (IN) (Ulyanov, Vedaldi, & Lempitsky, 2016) and ReLU activation except the output layer, where the Tanh function and Sigmoid function are used instead in the output layer of the colour image and mask respectively.

In Stage II, the global network is used to perceive and supplement the global facial information outside of the crucial regions, which can further refine the
Chapter 5: Local and Global Perception Generative Adversarial Network for Facial Expression Synthesis

Table 5.1: Generator network architecture (N: the number of output channels, K: kernel size, S: stride size, P: padding size).

<table>
<thead>
<tr>
<th>Layers</th>
<th>Layer Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global generator</td>
<td></td>
</tr>
<tr>
<td>Conv+IN+ReLU</td>
<td>N64, K7×7, S1, P3</td>
</tr>
<tr>
<td>Conv+IN+ReLU</td>
<td>N128, K4×4, S2, P1</td>
</tr>
<tr>
<td>Conv+IN+ReLU</td>
<td>N256, K4×4, S2, P1</td>
</tr>
<tr>
<td>6×Residual Block</td>
<td>N256, K3×3, S1, P1</td>
</tr>
<tr>
<td>DeConv</td>
<td>N128, K4×4, S2, P1</td>
</tr>
<tr>
<td>DeConv</td>
<td>N64, K4×4, S2, P1</td>
</tr>
<tr>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>Color image: Conv+Tanh</td>
<td>N3, K7×7, S1, P3</td>
</tr>
<tr>
<td>Attention mask: Conv+Sigmoid</td>
<td>N1, K7×7, S1, P3</td>
</tr>
<tr>
<td>Local generator</td>
<td></td>
</tr>
<tr>
<td>Conv+IN+ReLU</td>
<td>N64, K7×7, S1, P3</td>
</tr>
<tr>
<td>Conv+IN+ReLU</td>
<td>N128, K4×4, S2, P1</td>
</tr>
<tr>
<td>Conv+IN+ReLU</td>
<td>N256, K4×4, S2, P1</td>
</tr>
<tr>
<td>3×Residual Block</td>
<td>N256, K3×3, S1, P1</td>
</tr>
<tr>
<td>DeConv</td>
<td>N128, K4×4, S2, P1</td>
</tr>
<tr>
<td>DeConv</td>
<td>N64, K4×4, S2, P1</td>
</tr>
<tr>
<td>Output</td>
<td></td>
</tr>
<tr>
<td>Color image: Conv+Tanh</td>
<td>N3, K7×7, S1, P3</td>
</tr>
<tr>
<td>Attention mask: Conv+Sigmoid</td>
<td>N1, K7×7, S1, P3</td>
</tr>
</tbody>
</table>

generated results. For the input of the global network, after padding the generated local facial regions \( I_{local} = \{ I_{\text{eye}}, I_{\text{mouth}} \} \), we first blend generated local facial regions \( I_{local} \) and background region \( I_{bg} \) outside of the crucial regions to an aggregated result and then concatenate the target facial expression label \( f_y \) to it. As shown in Table 5.1, the global generator has a similar structure with the local generators but utilizing six bottleneck residual blocks. The output of the global network \( G_{local} \) is the final high-quality facial image \( I_y \) with target facial expression \( f_y \).

c) Discriminator

As shown in Figure 5.1, the discriminator \( D \) for the proposed method contains two local discriminators \( D_{local} = \{ D_{\text{eye}}, D_{\text{mouth}} \} \) and one global discriminator \( D_{global} \). The local discriminator examines different local regions to evaluate the quality of local details, while the global discriminator examines the final output of the whole facial information to judge the holistic facial features. All discriminators
Table 5.2: Discriminator network architecture (N: the number of output channels, K: kernel size, S: stride size, P: padding size, h and w: the height and width of the input image, $n_d$: the number of categories of facial expression).

<table>
<thead>
<tr>
<th>Layers</th>
<th>Layer Information</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Global discriminator</strong></td>
<td></td>
</tr>
<tr>
<td>Conv+Leaky ReLU</td>
<td>N64, K4×4, S2, P1</td>
</tr>
<tr>
<td>Conv+Leaky ReLU</td>
<td>N128, K4×4, S2, P1</td>
</tr>
<tr>
<td>Conv+Leaky ReLU</td>
<td>N256, K4×4, S2, P1</td>
</tr>
<tr>
<td>Conv+Leaky ReLU</td>
<td>N512, K4×4, S2, P1</td>
</tr>
<tr>
<td>Conv+Leaky ReLU</td>
<td>N1024, K4×4, S2, P1</td>
</tr>
<tr>
<td>Conv+Leaky ReLU</td>
<td>N2048, K4×4, S2, P1</td>
</tr>
<tr>
<td><strong>Local discriminator</strong></td>
<td></td>
</tr>
<tr>
<td>Conv+Leaky ReLU</td>
<td>N64, K4×4, S2, P1</td>
</tr>
<tr>
<td>Conv+Leaky ReLU</td>
<td>N128, K4×4, S2, P1</td>
</tr>
<tr>
<td>Conv+Leaky ReLU</td>
<td>N256, K4×4, S2, P1</td>
</tr>
<tr>
<td>Conv+Leaky ReLU</td>
<td>N512, K4×4, S2, P1</td>
</tr>
</tbody>
</table>

| Output                        |                         |
| Real or fake: Conv            | N1, K3×3, S1, P1        |
| Classifier: Conv              | $N(n_d), K \frac{h}{64} \times \frac{w}{64}, S1, P0$ |

have similar structures which are adapted from the PatchGAN architecture of (Isola et al., 2017). Table 5.2 shows the details of the discriminator network architecture including the local discriminator and global discriminator. As shown in Table 5.2, the basic architecture of these three discriminators has no normalization and contains several convolutional layers for downsampling followed by Leaky ReLU activations with a slope of 0.01. The number of downsampling convolutional layers of the local discriminator $D_{local}$ is set to 4 based on the size of the local regions while that of the global discriminator $D_{global}$ is set to 6. We also select the parameters of layers in each discriminator such as filter size, kernel size and stride according to (Isola et al., 2017). We not only utilize each discriminator to distinguish whether the synthesized facial expression is a real or not but also use the auxiliary classifier of each discriminator to predict its category of facial expression.
5.3.2 Loss Function

A weighted sum of four losses is imposed to supervise the proposed LGP-GAN in an end-to-end manner, including adversarial loss, facial expression classification loss, reconstruction loss and attention loss. Given a facial image $I_x$ with the facial expression $f_x$, the input of the network is $I_{input} = \{I_x, I_{eye}, I_{mouth}\}$ and the target facial expression is $f_y$. We introduce a two-stage cascaded generator $G = \{G_{global}, G_{local}\}$ in the proposed LGP-GAN. The output of two local generator networks $G_{local} = \{G_{eye}, G_{mouth}\}$ and one global generator network $G_{global}$ are $I_{local} = \{I_{eye}, I_{mouth}\}$ and $I_y$ respectively. We define $I_{output} = \{I_{eye}, I_{mouth}, I_y\}$ to represent the output of three subnetworks in the following section. And the output attention mask of three subnetworks is defined as $M = \{M_{eye}, M_{mouth}, M_y\}$.

In order to generate the realistic facial image, we use a composite discriminator $D = \{D_{global}, D_{local}\}$ in LGP-GAN to classify the output as real/fake and its category. $D_{local} = \{D_{eye}, D_{mouth}\}$ and $D_{global}$ are used to examine the outputs of the local generator $I_{local} = \{I_{eye}, I_{mouth}\}$ and global generator $I_y$.

**Adversarial Loss.** In the proposed method, we utilize an adversarial loss from WGAN-GP (Gulrajani et al., 2017) to make the generated facial images indistinguishable from the real images. The adversarial loss can be written as:

$$
\mathcal{L}_{adv} = \sum_{D_{i} \in D, G_{i} \in G} \left\{ E_{i} \left[ \log D_{i}(I_{i}) \right] - E_{i, f_y} \left[ \log D_{i}\left(G_{i}(I_{i}, f_y)\right)\right]\right. \\
+ \left. \lambda_{gp} E_{I} \left[ \left\| \nabla_{I} D_{i}(\tilde{I}) \right\|_{2} - 1 \right]^{2} \right\} 
$$

(5-2)

where $I_i \in I_{input}$, $\tilde{I}$ is the random interpolation distribution between the input image and generated image. And $\lambda_{gp}$ is a penalty coefficient which was set as 10 in the experiment.

**Facial Expression Classification Loss.** The goal of facial expression synthesis is to translate an input image $I_x$ with arbitrary facial expression $f_x$ into an output image $I_y$ with the target facial expression $f_y$. To achieve this aim, we add an
auxiliary classifier on top of the discriminator and impose the Categorical Cross Entropy (CCE) as classification loss function during the training process. The objective can be divided into two parts: a classification loss of real images used to optimize the parameters of the discriminator and a classification loss of generated images used to optimize the parameters of the generator. The facial expression classification loss can be formulated as:

\[
\mathcal{L}_{\text{cls}}^G = \sum_{G_i \in G} E_{I_i \in \text{output}} \left[ -\log D_i(f_y | I_i) \right] \tag{5-3}
\]

\[
\mathcal{L}_{\text{cls}}^D = \sum_{D_i \in D} E_{I_i \in \text{input}} \left[ -\log D_i(f_x | I_i) \right] \tag{5-4}
\]

**Reconstruction Loss.** To maintain the identity information that the faces in both input and output images are from the same person, we utilize reconstruction loss including a cycle reconstruction loss and a self-reconstruction loss to add extra constraints. Inspired by (Isola et al., 2017), we use L1 loss for the reconstruction since it produces less blur compared with L2 loss. During the process of facial expression synthesis, we minimize the L1 difference between the original image and its final reconstruction which is the final output image of the whole network. The cycle reconstruction loss and self-reconstruction loss can be formulated as:

\[
\mathcal{L}_{\text{cytc}} = E_{I_x} \left[ \left\| G(G(I_x, f_y), f_x) - I_x \right\|_1 \right] \tag{5-5}
\]

\[
\mathcal{L}_{\text{self}} = E_{I_x} \left[ \left\| G(I_x, f_x) - I_x \right\|_1 \right] \tag{5-6}
\]

**Attention Loss.** The purpose of the attention mask is to make the network focus on the facial areas where large changes occur in the process of facial expression synthesis. However, the attention mask is very easy to saturate to 1 during training which significantly reduces the effect of the generator. Therefore, we borrow the idea of attention loss from (Pumarola et al., 2018) to avoid this situation using L2 loss. The attention loss can be represented as:

\[
\mathcal{L}_{\text{att}} = E_{I_x} \left[ \left\| M_{\text{eye}} \right\|_2 + \left\| M_{\text{mouth}} \right\|_2 + \left\| M_y \right\|_2 \right] \tag{5-7}
\]

**Full Loss.** The full loss function for G and D are expressed, respectively, as:
Chapter 5: Local and Global Perception Generative Adversarial Network for Facial Expression Synthesis

\[
\mathcal{L}_G = -\mathcal{L}_{adv} + \lambda_1 \mathcal{L}_{cls}^G + \lambda_2 \mathcal{L}_{cycle} + \lambda_3 \mathcal{L}_{self} + \lambda_4 \mathcal{L}_{att} \quad (5-8)
\]

\[
\mathcal{L}_D = \mathcal{L}_{adv} + \lambda_1 \mathcal{L}_{cls}^D \quad (5-9)
\]

where \(\lambda_1, \lambda_2, \lambda_3, \lambda_4\) are the hyper-parameters that represent the weight of each loss function. During network training, referring to (Y. Choi et al., 2018; Z. He et al., 2019; Pumarola et al., 2018) and the network performance, we set the \(\lambda_1 = 1, \lambda_2 = 10, \lambda_3 = 10, \lambda_4 = 0.1\) in our experiment.

### 5.4 Experiment

This section first introduces the database and evaluation metrics used in our experiment and then describes the implementation details of our experiment. Finally, this section presents the experimental results of our proposed method and compares with existing methods.

#### 5.4.1 Database and Evaluation Metrics

The Radboud Faces Dataset (RaFD) (Langner et al., 2010) is used as our training data. The RaFD consists of 4,824 images with a size of 681×1024 collected from 67 participants under the laboratory settings. The facial images of each participant are captured by cameras from three different angles. We only use the frontal facial images which are about 1,608 in total in order to ensure the eye and mouth region are visible. And we randomly select 90% facial images for the training set and the remaining 10% facial images for the test set. The facial images in this database are labelled by eight discrete categories of facial expressions including angry, contemptuous, disgusted, fearful, happy, neutral, sad and surprised.

We use two different evaluation metrics in our experiment for quantitative evaluation of the facial expression synthesis task. Inception Score (IS) (Salimans et al., 2016) and Fréchet Inception Distance (FID) (Heusel, Ramsauer, Unterthiner, Nessler, & Hochreiter, 2017) are widely used to evaluate the quality of the synthesized images. The IS scores measure the image quality through the
probability outputs of the pre-trained Inception network which calculates the KL divergence between the conditional distribution and marginal distribution. The FID scores utilize a pre-trained Inception network to extract the final average pooling features of the real images and the synthesized images and measure their similarity.

5.4.2 Implementation Details

During the stage of preprocessing, all the input facial images were first aligned and cropped to the size of 128×128 according to the facial landmarks detected by Dlib (King, 2009) which is an open-source library capable of detecting 68 landmarks of the face. And then we obtained the corresponding eye and mouth region of each facial image according to the detected facial landmarks. In our experiments, the sizes of the eye region and mouth region are 48×96 and 48×64, respectively. The proposed LGP-GAN was trained in the PyTorch framework through an NVIDIA GeForce GTX 1080 GPU with 8 GB memory on a desktop PC. We used Adam optimizer (Kingma & Ba, 2014) with $\beta_1 = 0.5, \beta_2 = 0.999$ to train the model. The batch size was set as 16. For the learning rate, we initially set the learning rate as 0.0001 for the first 100k iterations and decayed linearly to 0 for the next 100k iterations. We trained the network updating the discriminator five times for each generator update. The training took about 1 day to complete.

5.4.3 Qualitative Evaluations

For qualitative evaluations, eight categories of facial expressions (Angry, Contemptuous, Disgusted, Fearful, Happy, Neutral, Sad and Surprised) in the RaFD database were used for facial expression synthesis to evaluate our proposed LGP-GAN. Figure 5.3 illustrates the generated results of each category of facial expressions using our proposed LGP-GAN. As Figure 5.3 shows, our proposed LGP-GAN can generate realistic facial images with eight categories of facial expressions and their local regions such as eyes and mouths are clear and sharp.
Chapter 5: Local and Global Perception Generative Adversarial Network for Facial Expression Synthesis

The further evaluation is to compare the performance of the proposed LGP-GAN with the current state-of-the-art approaches including StarGAN (Y. Choi et al., 2018), AttGAN (Z. He et al., 2019) and GANimation (Pumarola et al., 2018). StarGAN and AttGAN are two representative methods treating the task of facial expression synthesis as a subproblem of facial attribute editing. GANimation is another representative method that designs a special GAN network architecture and achieves impressive results on facial expression synthesis recently. Figure 5.4 shows the qualitative comparison of the facial expression synthesis using different methods on the RaFD database. In Figure 5.4, the input facial images of each column are from the RaFD database. Each column represents the process of synthesizing a target facial expression using different approaches including

Figure 5.3: Qualitative results of facial expression synthesis on RaFD database (Input, Angry, Contemptuous, Disgusted, Fearful, Happy, Neutral, Sad, Surprised).
Chapter 5: Local and Global Perception Generative Adversarial Network for Facial Expression Synthesis

![Figure 5.4: Qualitative comparison of facial expression synthesis on RaFD database (target facial expression from top to bottom: contemptuous, disgusted and happy).](image)

AttGAN, StarGAN, GANimation and our proposed LGP-GAN. As illustrated in Figure 5.4, we can conclude observations as follows. Firstly, among these state-of-the-art methods, GANimation has better performance than AttGAN and StarGAN in terms of facial expression synthesis. For instance, AttGAN and StarGAN fail to transform happy into contemptuous in the first row of Figure 5.4, which retains some details of happy especially in the local region such as the mouth. The experimental results demonstrate that the methods for facial attribute editing are not generalizable enough for generating the facial expression and facial expression synthesis is a more complex task because of large transformations and deformations in facial regions. Secondly, as shown in Figure 5.4, all state-of-the-art methods are prone to generating blur and overlapping around local regions when large facial deformation occurs such as transforming from mouth open to mouth closed. The reason is that these methods only consider the general facial information without paying special attention to local facial regions. Finally, it is obvious that the proposed LGP-GAN method has shown clear superior overall quality over the existing state-of-the-art methods in terms of quality, clarity, and coherence of target facial expressions. In particular, the proposed LGP-GAN has
fewer blurs and overlapping in generated facial expressions especially in local facial regions such as the eye and mouth region. The main reason is that the proposed LGP-GAN can capture texture details of crucial facial regions and learn the whole facial information simultaneously. In addition, the proposed LGP-GAN has a two-stage cascaded architecture that divides the facial expression synthesis into local facial region generation and global facial image generation, which enables to learn the facial details step by step.

5.4.4 Quantitative Evaluations

Table 5.3: Comparison of our method with other methods on RaFD database using IS and FID.

<table>
<thead>
<tr>
<th>Method</th>
<th>$IS \uparrow$</th>
<th>$FID \downarrow$</th>
</tr>
</thead>
<tbody>
<tr>
<td>StarGAN</td>
<td>1.29</td>
<td>14.73</td>
</tr>
<tr>
<td>AttnGAN</td>
<td>1.26</td>
<td>18.18</td>
</tr>
<tr>
<td>GANimation</td>
<td>1.30</td>
<td>12.20</td>
</tr>
<tr>
<td>LGP-GAN</td>
<td><strong>1.31</strong></td>
<td><strong>11.88</strong></td>
</tr>
</tbody>
</table>

For quantitative evaluations, we evaluate the quality of the generated facial expressions using IS (Salimans et al., 2016) and FID (Heusel et al., 2017) metrics. The better quality of the synthesized image is, the smaller of the FID and the larger of IS are. We also compared the proposed method with state-of-the-art methods on the RaFD database. Table 5.3 illustrates the comparison results using IS and FID metrics on the RaFD database. For IS metric, our proposed LGP-GAN slightly outperforms other state-of-the-art methods within a max margin of 0.05 as shown in Table 5.3. It is because that IS just evaluated the synthetic results using the output of pre-trained Inception network on ImageNet instead of comparing the synthetic distribution with the real distribution, which is not well appropriate to facial image database such as RaFD. But IS is still one of the most widely used metrics for evaluating the performance of GANs. For FID metric, as shown in Table 5.3, our proposed LGP-GAN has the lowest FID, which indicates a distinct
Chapter 5: Local and Global Perception Generative Adversarial Network for Facial Expression Synthesis

lead over other state-of-the-art methods. Compared with IS, FID considers both real samples and synthetic samples which is one of the most robust metrics for evaluating the performance of GANs. In terms of IS and FID metric, GANimation and the proposed LGP-GAN outperform StarGAN and AttGAN. It shows that the approaches for facial attribute editing is not well suitable for the facial expression synthesis task. In addition, the proposed LGP-GAN achieves better performance than GANimation, which shows the importance of crucial facial regions in facial expression synthesis and can further improve the performance.

5.5 Summary

In this chapter, we propose a novel end-to-end facial expression synthesis method called Local and Global Perception Generative Adversarial Network (LGP-GAN). The proposed LGP-GAN has a two-stage cascaded architecture including two local networks and a global network. The proposed LGPGAN can fully utilize local facial information and global facial information in the process of facial expression synthesis. The quantitative and qualitative evaluations on the public database show that the proposed LGP-GAN has superior performance compared with state-of-the-art approaches on the task of facial expression synthesis.
Chapter 6

Real-Time Facial Affective Computing on Mobile Devices

6.1 Introduction

Facial affect plays a crucial role in our daily lives such as psychological analysis, medical diagnosis, education, decision-making, customer marketing, and advertising. Driven by the vast application demands, facial affective computing has become an active research field and has attracted a lot of research attention from various research areas such as human-computer interaction, computer vision and artificial intelligence. In particular, facial affective computing is one of the most important components of human-computer interaction, because it provides a new dimension to human-machine interactions. For instance, if robots can analyse human facial affect, they can have appropriate responses and behaviours according to the analysis results.

In the field of psychology, affect is a term for the external exhibition of internal emotions and feelings. The aim of facial affective computing is to develop algorithms or systems to interpret and estimate human affects from human facial images or videos (Tao & Tan, 2005). Specifically, facial affect is usually described based on two types of models: one is the categorical model, namely facial expressions, such as the six basic facial expressions (Happiness, Sadness, Fear, Anger, Surprise and Disgust) defined by Ekman et al. (Ekman & Friesen, 1971); another is the dimensional model which uses valence and arousal to represent the facial expression intensity on a continuous scale. Valence distinguishes the degree
of positive or negative of a facial expression, and arousal indicates the degree of intriguing/agitating or calming/soothing of an event (Russell, 1980).

Most research about facial affective computing has mainly focused on constrained laboratory environments instead of real-world scenarios and is not suitable for large-scale practical applications. Due to recent advances in deep learning technologies and the advent of databases which are large-scale and in-the-wild, convolutional neural networks (CNNs) have obtained remarkable performance in facial affective computing and outperformed many conventional methods with a large margin. However, one of the drawbacks of the CNNs is that the computational complexity increases significantly as the performance improves. Therefore, CNNs are usually performed on high-performance devices. But for ordinary users, these devices are expensive and not portable enough. Moreover, many users usually have no opportunity to access these devices.

Recently, mobile devices with embedded cameras have become inseparable parts of people’s lives, which play an important role in many personal and business applications such as video chat and social networks. Moreover, a wide variety of emerging mobile applications have been actively studied in various areas such as human-computer interaction, education and entertainment. The popularity and portability of mobile devices with high-quality cameras motivate us to develop real-time facial affective computing on mobile devices using CNNs for ordinary users. However, conventional CNNs are not easy to implement and generalizable enough for real-time applications on mobile devices where the storage, memory and computational power are relatively limited. Therefore, it is of central importance to design a novel CNN architecture for facial affective computing on mobile devices.

The aim of this chapter is to investigate the possibility of CNNs embedded in mobile devices for real-time facial affective computing under real-world scenarios. The main challenge in this task is the processing of the users’ facial images on the device itself, without uploading the images to the external cloud server for processing. There are two advantages to do this: one is that it is beneficial for the
security of user information and privacy and the other is that compared with uploading to the cloud server, the processing speed is faster on the mobile devices since no data is uploaded through the network. Major contributions of our work are as follows:

- To overcome the limited processing power of mobile devices, we propose a light-weight but effective CNN architecture for real-time facial affective computing using both categorical model and dimensional model on mobile devices, which well balances the performance and computational complexity.
- We have explored facial affective computing using both categorical model (e.g., happy, neutral) and dimensional model (valence and arousal) on mobile devices.
- The proposed network has been implemented on mobile devices with an embedded mobile camera and requires only a low consumption of memory and storage. The implemented application can analyse users’ facial affect in real-time, which can be a good functional component for other emerging mobile applications.

6.2 Related Work

Many studies in facial affective computing have been published over the past few years (Y. Wang et al., 2017; Yiming Wang et al., 2016; H. Yu & Liu, 2014) since facial affect plays an important role in human-computer interaction (Cid, Moreno, Bustos, & Núñez, 2014; Goulart et al., 2019; Leo et al., 2018). For example, facial affect computing is a functional supplement for intelligent surveillance (Varghese & Thampi, 2018) when combining with multiple moving targets tracking technologies (X. Zhou, Yu, Liu, & Li, 2015) to detect the facial affect of the crowd in video surveillance to avoid potential dangers and disasters. Previous works mainly focused on the categorical model based on six basic facial expressions.
(Happiness, Sadness, Fear, Anger, Surprise and Disgust) defined by Ekman et al. (Ekman & Friesen, 1971). The extracted features are applied to the classifier such as support vector machines (SVM) (Shan et al., 2009; H. Yu & Liu, 2015), AdaBoost (Yubo Wang et al., 2004) and hidden Markov models (HMMs) (Uddin et al., 2009) to achieve facial expression recognition. Traditional methods mainly depend on hand-crafted features based on facial information such as geometry, appearance or texture information. Therefore, according to the different facial information, the features for facial expression recognition can be roughly divided into two categories: geometric features and appearance features.

Geometric features describe the locations and shapes of facial components extracted from facial images such as measurements among coordinates of landmarks on the face. Active Appearance Model (AAM) (Edwards et al., 1998) and the Active Shape Model (ASM) (Cootes et al., 1995) are used in most geometric feature-based methods for detecting facial components, for instance, Choi et al. (H.-C. Choi & Oh, 2006) proposed a real-time facial expression recognition method used AAM with second order minimization and a neural network. Appearance features use the texture information of the face, including a histogram of oriented gradients (HoG) (J. Chen, Chen, Chi, & Fu, 2014; Orrite, Gañán, & Rogez, 2009), local binary pattern (LBP) (Luo, Wu, & Zhang, 2013; Shan et al., 2009), scale invariant feature transform (SIFT) (Barroso, Santos, & Proença, 2013; Carcagni et al., 2015), and Gabor features (Gu et al., 2012; H. Yang et al., 2018). However, these hand-crafted features can be implemented on facial expression recognition under laboratory-controlled environment successfully but are not generalizable enough for the variation from the wild environment such as lighting, camera view, head pose and ethnicity.

In recent years, deep learning, especially convolutional neural networks (CNNs), has achieved impressive performance for various computer vision and pattern recognition tasks (B. Hu & Wang, 2020; Xia et al., 2019; Xing et al., 2017; N. ZHENG, 2019). There are also enormous methods using CNNs in facial expression recognition in recent years (Lian et al., 2020; Sharma, Balouchian, &
Foroosh, 2018; J. Zeng, Shan, & Chen, 2018). In addition, there also have been some research on the dimensional model since the development of deep learning (Mollahosseini et al., 2017; Siqueira, 2018). Within the dimensional model, a particular facial affect can be mapped on an arousal/valence value, which describes the range of arousal (high to low) and valence (pleasure to displeasure). In these studies, deep learning-based architectures are especially efficient for facial affective computing and can achieve high performance. The main reason for the success of CNNs is that CNNs are able to overcome the limitations of the handcrafted features in the wild settings by directly extracting highly discriminative features of the raw data.

However, existing CNNs are not generalizable enough for mobile devices, since they were not originally designed for mobile devices and didn’t consider the storage, memory and computational power of mobile devices as important attributes in the architecture design. Real-time computing on mobile platforms is challenging because of hardware performance constraints. To the best of our knowledge, for mobile affective computing, existing approaches are mainly limited to use one categorical model for facial expression recognition (Suk & Prabhakaran, 2014). FaceReader (Noldus Information Technology, Wageningen, The Netherlands) is a successful and commercially available software program that can automatically analyse facial expressions using the categorical model. But for the dimensional model, FaceReader can only perform the valence prediction on mobile platforms and the arousal prediction is still unavailable currently. Therefore, real-time facial affective computing using both categorical model (e.g., happy and neutral) and dimensional model (valence and arousal) are barely used for mobile platforms. To overcome these limitations, we design a light-weight CNN architecture for facial affective computing on mobile devices. The proposed architecture well balances performance and computational complexity. We have implemented the proposed network architecture on actual mobile devices to demonstrate its feasible and high efficiency. The implemented mobile application
can analyse users’ facial affect and output both users’ facial expression category and values of valence and arousal in real-time.

### 6.3 Methodology

We propose a light-weight CNN architecture for facial affective computing. The network architecture is shown in Figure 6.1. In this section, we firstly provide instruction of the proposed network structure and then discuss the loss functions used in the training stage.

![Figure 6.1: The architecture of our light-weight deep model for facial affective computing (the facial image is from the AffectNet (Mollahosseini et al., 2017)). The network consists of four convolutional blocks, where the raw image pixel is treated as the input.](image)

<table>
<thead>
<tr>
<th>B1</th>
<th>B2</th>
<th>B3</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 × conv layer</td>
<td>3 × conv layer</td>
<td>3 × conv layer</td>
</tr>
<tr>
<td>(32, 3 × 3, 1 × 1)</td>
<td>(64, 3 × 3, 1 × 1)</td>
<td>(128, 3 × 3, 1 × 1)</td>
</tr>
<tr>
<td>1 × Maxpooling</td>
<td>1 × Maxpooling</td>
<td>1 × Maxpooling</td>
</tr>
<tr>
<td>(2 × 2, 2 × 2)</td>
<td>(2 × 2, 2 × 2)</td>
<td>(2 × 2, 2 × 2)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>B4</th>
<th>B5</th>
<th>B6</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 × conv layer</td>
<td>1 × Global average pooling</td>
<td>Softmax (classification) or linear (regression)</td>
</tr>
<tr>
<td>(256, 3 × 3, 1 × 1)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 6.3.1 Network Architecture

This work mainly focuses on designing a light-weight CNN for facial affective computing on mobile devices, which uses facial images as input and outputs the
categories of facial expression and values of valance and arousal. We start with VGG network (Simonyan & Zisserman, 2014) for facial affective computing to create our network. Unlike other conventional CNNs, VGG network consists of several convolutional blocks, each of which contains several convolutional layers stacked on top of each other. These convolutional layers have a very small kernel size of $3 \times 3$ with the stride of 1. And there is a Maxpooling layer which often has a pool size of $2 \times 2$ with a stride of 2 at the end of each convolutional block. The convolutional layers in the same block often have the same number of filters.

The proposed light-weight CNN is inspired by the VGG network (Simonyan & Zisserman, 2014), which can be regarded as a simplified VGG network with the attributes of few parameters and low computation complexity but high performance. These attributes of the proposed network architecture meet the requirements of mobile development. The main design principle of the proposed network architecture is to reduce the computation complexity and parameters of the original VGG network. In fact, we strongly reduce the number of parameters in all layers compared with the original VGG network. The architecture of the proposed network is shown in Figure 6.1 and Table 6.1. The proposed network architecture contains four convolutional blocks. For each convolutional block except the first and last one, there are three convolutional layers and one Maxpooling layer. The first convolutional block has two convolutional layers and one Maxpooling layer. And the last convolutional block only contains three convolutional layers. Batch normalization (Ioffe & Szegedy, 2015) is also used after each convolutional layer to improve the training process. The numbers of filters of each convolutional layer of four convolutional blocks are set to 32, 64, 128 and 256 respectively.

After four convolutional blocks, we use a Global Average Pooling (GAP) layer instead of fully connected layers to extract 256-dimensional feature vectors. The purpose of using GAP is to reduce parameters and computation complexity. GAP was proposed by Lin et al. (Lin, Chen, & Yan, 2013). In (Lin et al., 2013), which was used to takes the average of each feature map replacing the traditional fully
Chapter 6: Real-Time Facial Affective Computing on Mobile Devices

connected layers on the CNN. As shown in Figure 6.2, the main difference between those two layers is that the output of a flatten layer has to be fed into a fully connected layer to get features we need, but the output of the global average pooling layer has already the desired dimension. The fully connected layer can add a large number of parameters and then increase the complexity of the model and the degree of overfitting. Therefore, the main advantage of GAP over the fully connected layers is that there is no need for parameter optimization in the global average pooling, which greatly reduces parameters and computation complexity.

![Figure 6.2: Difference between fully connected layer and global average pooling layer.](image)

Finally, the facial expression recognition and prediction of valence and arousal in this chapter are regarded as a classification and regression task respectively. And the proposed light-weight CNN architecture is used for both classification and regression tasks. Therefore, after GAP, the output layer of the network contains 8 nodes with softmax activation or 2 nodes with linear activation depending on the tasks of classification or regression.

### 6.3.2 Loss Function

In this work, we use both the categorical model (e.g., happy and neutral) and dimensional model (valence and arousal) to describe the facial affect. For facial expression recognition, we regard it as a multi-class classification problem. And we formulate the task of valence and arousal estimation as a regression problem. Therefore, the loss function for the task of facial affective computing can be represented as the formula:
\[ L_{\text{Affect}} = \{ \text{loss}_c, \text{loss}_r \} \]  

(6-1)

where \( \text{loss}_c \) and \( \text{loss}_r \) represent the loss function for the task of classification and regression respectively, and next we will introduce them in detail.

**Facial expression recognition:** For multi-class classification problems, Categorical Cross Entropy (CCE) is a commonly used loss function as follows:

\[ \text{loss}_{\text{CCE}} = - \sum_{i=1}^{N} g_i \log(p_i) \]  

(6-2)

where \( N \) is the number of images in the training data; \( p_i \) and \( g_i \) represent the prediction and ground truth. However, CCE can lead to over-fitting problem and make the model become too confident about its predictions. Therefore, we modify the CCE loss function using the Label Smoothing Regularization (Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna, 2016), the variant of CCE loss function is thus given by:

\[ \text{loss}_c = -(1 - \varepsilon)\text{loss}_{\text{CCE}} - \varepsilon H_{u,p} \]  

(6-3)

where \( \varepsilon \) is a hyper parameter, it is set to 0.1; \( H_{u,p} \) represents the dissimilarity between the predicted distribution \( p \) and its uniform distribution, defined as:

\[ H_{u,p} = \sum_{i=1}^{N} \frac{1}{C} p_i \log(p_i) \]  

(6-4)

where \( C \) is the number of categories.

**Valence and arousal estimation:** Mean Squared Error (MSE) loss function is usually used for regression problem which shown as follows:

\[ \text{loss}_{\text{MSE}} = \frac{1}{N} \sum_{i=1}^{N} (p_i - g_i)^2 \]  

(6-5)

where \( N \) is the number of images in the training data; \( p_i \) and \( g_i \) represent the prediction and ground truth of valence and arousal of each facial image. However, since range of the value of valence and arousal are from −1 to 1, it can lead to inaccurate estimation if we only use MSE, for instance, the prediction values of valence are −0.2 and 0.6 respectively, they have the same RMSE if the ground truth is 0.2, but prediction of 0.6 is better than prediction of −0.2 since prediction
of 0.6 expresses a positive emotion similar to the ground truth. Therefore, we modify the MSE loss function giving more punishment to the samples with different signs, the variant of MSE loss function which shown as follows:

\[
 f = \begin{cases} 
 \text{loss}_{\text{MSE}}, & S(p_i, g_i) = 0 \\
 \text{loss}_{\text{MSE}} + \alpha \frac{1}{N} \sum_{i=1}^{N} \text{sign}(S(p_i, g_i)), & S(p_i, g_i) \neq 0 
\end{cases} 
\] (6-6)

where \( \alpha \) is hyper parameter, it is set to 0.1; \( S(p_i, g_i) \) is defined as:

\[
 S(p_i, g_i) = \left( \text{sign}(p_i) - \text{sign}(g_i) \right)^2 
\] (6-7)

where \( \text{sign}(\cdot) \) is the sign function.

### 6.4 Experiment

#### 6.4.1 Experimental Setup

One of the key issues for implementing facial affective computing using CNNs is the choice of database. Most existing databases for facial affective computing in the wild are so small and only contain annotated image data of the categorical model (facial expressions). There are few databases for facial affective computing providing annotated image data of the dimensional model (valence and arousal). For the proposed light-weight CNN training, we use by far the largest database in the field of facial affective computing, AffectNet (Mollahosseini et al., 2017), which provides annotated the categorical model and dimensional model image data. This database includes more than 1M facial images, which its creators collected them from three major search engines by using 1,250 related keywords in six different languages. About 420,299 manually annotated facial images have labels of categories of facial expressions and the values of valence and arousal. Figure 6.3 shows examples of annotated facial images from the AffectNet database.

For facial expression recognition, we regard it as a multi-class classification problem. The numbers of each category of facial expressions for training is listed in Table 6.2. During the process of training, we use eight categories of facial
expressions for training including Neutral, Happy, Sad, Surprise, Fear, Disgust, Anger and Contempt and invalid facial expressions (none, uncertain and no-face) in AffectNet database are discarded. The total number of facial images for facial expression classification in the training set is 287,651. We formulate the task of valence and arousal estimation as a regression problem in which the network learns to predict the values of valence and arousal from a face image. In the AffectNet database, the facial images are manually annotated for the values of valence and

![Examples of the eight categories of facial expressions](image)

Figure 6.3: Examples of the eight categories of facial expressions (Neutral, Happy, Sad, Surprise, Disgust, Fear, Anger and Contempt) and their values of valence and arousal from the AffectNet (Mollahosseini et al., 2017).

<table>
<thead>
<tr>
<th>Facial Expression</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neutral</td>
<td>74,874</td>
</tr>
<tr>
<td>Happy</td>
<td>134,415</td>
</tr>
<tr>
<td>Sad</td>
<td>25,459</td>
</tr>
<tr>
<td>Neutral</td>
<td>14,090</td>
</tr>
<tr>
<td>Surprise</td>
<td>6,378</td>
</tr>
<tr>
<td>Fear</td>
<td>3,803</td>
</tr>
<tr>
<td>Disgust</td>
<td>24,882</td>
</tr>
<tr>
<td>Contempt</td>
<td>3,750</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>287,651</strong></td>
</tr>
</tbody>
</table>

Table 6.2: The numbers of each category of facial expressions in AffectNet for training.
arousal from −1 to 1. The facial images labelled with −2 are invalid data in the database which we also discard. The total number of training data is 320,739.

During the stage of preprocessing, we only use manually annotated images that are divided by its creators into training and validation sets and contains about 420,299 images. We evaluate the proposed light-weight CNN on the AffectNet validation set since currently its test set has not been released. The total number of facial images for facial expression classification and valence and arousal estimation in the validation set are 4000 and 4500 respectively. The images of the AffectNet database are cropped to the size of a face bounding box provided by the database. And then we resize all the cropped face images to be an equal size of 96×96 px. In our experiment, we train two separate light-weight CNNs for facial expression recognition and valence and arousal estimation. The proposed light-weight CNN is trained on a desktop PC with a specification of Intel (Santa Clara, CA, USA) Core i7, 4.20 GHz processor, 16 GB of RAM memory and 8 GB NVIDIA (Santa Clara, CA, USA) GeForce GTX 1080 GPU. All tasks are trained end-to-end using TensorFlow. For all tasks, we use the stochastic gradient descent (SGD) method to optimize the model and set momentum as 0.9. The initial learning rate is set as 0.01 and divided by 10 after 15 epochs. We stop training at the 30th epoch which takes about 24 hours. During the training stage, we set the batch size as 128.

6.4.2 Evaluation Metrics

We evaluate the proposed light-weight CNN on the AffectNet validation set since currently its test set has not been released. The total number of facial images for facial expression classification and valence and arousal estimation in the validation set are 4,000 and 4,500 respectively. For facial expression classification, we use classification accuracy as the main evaluation metric since it is well-defined widely used metrics for evaluation of the classification task. And for the valence and arousal estimation, we use and calculate 4 different evaluation metrics for evaluation of valence and arousal estimation task, since it outputs the values of
valence and arousal in a continuous domain. In the following, we briefly review these metrics.

One of the most common evaluation metrics in a continuous domain is Root Mean Square Error (RMSE) which is defined as:

\[ RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (p_i - g_i)^2} \]  

(6-8)

where \( n \) is the number of images in the evaluation set; \( p_i \) and \( g_i \) are the prediction and ground truth of \( i \)-th image.

Pearson’s Correlation Coefficient (CC) (Nicolaou, Gunes, & Pantic, 2011) is another evaluation metric which can consider the covariance of prediction and ground-truth compared with RMSE:

\[ CC = \frac{cov(p, g)}{\sigma_p \sigma_g} \]  

(6-9)

where \( COV \) is covariance function; \( \sigma_p \) and \( \sigma_g \) are the standard deviation of each time series (e.g., prediction and ground-truth).

Based on CC, Concordance Correlation Coefficient (CCC) (Valstar et al., 2016) computes the square difference between the means of two compared time series:

\[ CCC = \frac{2 \rho \sigma_p \sigma_g}{\sigma_p^2 + \sigma_g^2 + (\mu_p - \mu_g)^2} \]  

(6-10)

where \( \rho \) is CC; \( \sigma_p^2 \) and \( \sigma_g^2 \) are the variance of each time series; \( \mu_p \) and \( \mu_g \) are the means for the two variables. Unlike CC, the predictions that are well correlated with the ground-truth but shifted in value are penalized in proportion to the deviation in the CCC.

Sign Agreement (SAGR) is a very important evaluation metric proposed in (Nicolaou et al., 2011) to evaluate the performance of a valence and arousal estimation with respect to the sign agreement. Therefore, SAGR is defined as:

\[ SAGR = \frac{1}{n} \sum_{i=1}^{n} \delta(sign(p_i), sign(g_i)) \]  

(6-11)

where \( \delta(\cdot) \) is the Kronecker delta function, defined as:
\[ \delta(a, b) = \begin{cases} 1, & a = b \\ 0, & a \neq b \end{cases} \] (6-12)

6.4.3 Experimental Results

Facial expression recognition: To evaluate our proposed light-weight CNN, eight facial expressions in AffectNet validation set are used: Neutral, Happy, Sad, Surprise, Fear, Disgust, Anger and Contempt. Each category of facial expressions contains 500 samples. Figure 6.4 shows the facial expression classification confusion matrix of the proposed light-weight CNN on AffectNet validation set. Among the eight facial expressions, the highest accuracy is Happy with an accuracy of 78.0%. The accuracies of other facial expressions can also be obtained from the confusion matrix: Neutral (54.4%), Sad (57.8%), Surprise (59.6%), Fear (59.6%), Disgust (53.0%), Anger (53.0%) and Contempt (52.6%). The average accuracy of all eight facial expressions is about 58.50%. We have also evaluated
Table 6.3: Classification accuracy (%) of facial expressions on AffectNet validation set (the bold value indicates the best classification accuracy).

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Support Vector Machine (Mollahosseini et al., 2017)</td>
<td>30.00%</td>
</tr>
<tr>
<td>Microsoft Cognitive Services (Mollahosseini et al., 2017)</td>
<td>37.00%</td>
</tr>
<tr>
<td>AlexNet (Mollahosseini et al., 2017)</td>
<td>58.00%</td>
</tr>
<tr>
<td>(Siqueira, 2018)</td>
<td>50.32%</td>
</tr>
<tr>
<td>(Sharma et al., 2018)</td>
<td>56.38%</td>
</tr>
<tr>
<td>(Zeng et al., 2018)</td>
<td>57.31%</td>
</tr>
<tr>
<td>Ours</td>
<td><strong>58.50%</strong></td>
</tr>
</tbody>
</table>

the proposed light-weight CNN by comparing its performance with the state-of-the-art methods including traditional methods and deep learning-based methods. Table 6.3 shows the results of the comparison on the AffectNet. From Table 6.3, we can find that our proposed method outperforms other existing methods in terms of classification accuracy on the AffectNet database.

**Valence and arousal estimation:** We have used and calculated 4 different evaluation metrics for evaluation of the valence and arousal estimation task including Root Mean Square Error (RMSE), Pearson’s Correlation Coefficient (CC), Concordance Correlation Coefficient (CCC) and Sign Agreement (SAGR). In order to evaluate the computation complexity, we have also added the number of trainable parameters as the evaluation metric. Table 6.4 shows the results of our experiments in the valence and arousal estimation on the validation set of the AffectNet databases. We have compared our method with other three state of the art methods which were proposed in (Mollahosseini et al., 2017) and (Siqueira, 2018). Mollahosseini et al. (Mollahosseini et al., 2017) utilized both Support Vector Regression (SVR) and AlexNet for valence and arousal estimation. And Siqueira (Siqueira, 2018) proposed a multi-task learning (MTL) network for this task. From Table 6.4, it is obvious that the performance of the proposed method outperforms SVR and MTL. Moreover, compared with AlexNet, the proposed method also achieves better performance in arousal estimation on all evaluation
metrics. For valence estimation, the performance of AlexNet slightly outperforms our method in valence prediction on all evaluation metrics except for SAGR. But the AlexNet has a high computational cost since the network has about 60 million trainable parameters. In terms of the computation complexity, the proposed light-weight CNN only has about 2 million trainable parameters which are far less than AlexNet.

In summary, the experimental results show that the proposed light-weight CNN can well balance the performance and computation complexity. It achieves superior performance while retaining the low computation complexity compared with state-of-the-art methods. It is thus more suitable for mobile development. Compared with (Mollahosseini et al., 2017), although the proposed approach shows a slightly inferior performance for valence prediction, it has better performance on facial expression recognition and arousal prediction. More importantly, the number of parameters of the proposed method is only about 2 million which is very important for mobile devices, since the storage, memory and computational power of mobile devices are very limited.

### 6.5 Design for Mobile Platforms

#### 6.5.1 Design Principles and Purpose

To achieve mobile device intelligence using deep learning methods is an emerging trend over recent years. For many mobile applications, the common solutions for
achieving mobile device intelligence can be roughly divided into two categories: cloud-based solution and local-based solution.

The strategy of the local-based solution allows to train a fixed deep learning model such as CNN model offline through a high performance device and then deploy it on mobile devices or the cloud to implement its function. The cloud-based solution allows to upload data to the cloud and then receive results, which may be the best solution for many mobile applications. This is because we don’t need to consider the constrained resources of mobile devices. We can save the storage space and use a high-performance CPU or GPU cluster from the cloud to improve the speed of algorithm execution. However, for facial affective computing, the cloud-based solution is not an ideal solution. There are two weaknesses: (1) privacy issue. For facial affective computing, it is indispensable to capture the facial images of users. Sending users’ facial images to the cloud to analyse facial affect will have potential problems of privacy security; (2) network latency. Since cloud-based solution needs to upload and receive the data from the cloud, the network environment will be essential for smooth performance. Any data dropout and latency will cause inconvenience when exchanging data.

Therefore, the local-based solution is an alternative solution and more suitable for the facial affective computing task, which could avoid the above issues. The local-based solution which processes the algorithm using local hardware of mobile devices. However, because of the limited storage and computational resources of mobile devices, it is a challenging task to design a CNN architecture and deploy it on mobile devices. In the mobile setting, there is a trade-off between hardware constraints, such as computational resources or storage space, and performance of the CNN models. In this chapter, according to the requirements of mobile development, we design a CNN architecture with the goal of balancing this trade-off. We have designed a light-weight CNN architecture for facial affective computing, which maximizes resource utilization and performance on mobile devices. Based on the proposed light-weight CNN, we have developed a facial affective computing application to detect the user’s facial affect in real-time and
the application with low hardware requirements can robustly run on the mobile platform. It can benefit many other useful mobile applications such as education, health care, driver monitoring system, and entertainment. In particular, facial affective computing is one of the most important parts of the driver monitoring system, because it can be used to assist safe driving. The drivers require a healthy emotional state during driving for the right judgment of the traffic. The results of facial affective computing can help drivers to recognize their emotions and make drivers aware of them. Therefore, the implemented mobile application for facial affective computing can be applied in the driver monitoring system, which is a good functional component.

6.5.2 Mobile Implementation

We implement a real-time mobile application of facial affective computing on the iOS platform with Swift. The proposed light-weight CNN architecture is deployed on the mobile application by using Apple’s Core ML framework which can integrate trained deep learning models into mobile applications. By using the CPU, GPU, and Neural Engine, Core ML can optimize on-device performance and minimize its memory and power consumption. Core ML is the foundation for mobile development which supports the vision for analysing images, natural language processing, speech and sound analysis. If the models are created and trained using a supported third-party deep learning framework such as TensorFlow, Core ML framework could provide a conversion tool to convert the trained model to the Core ML model format.

In our experiment, we first use the human face detection method proposed by Viola and Jones (Viola & Jones, 2004) with default parameters for face detection in the mobile application, and then the input facial image is cropped based on the face bounding box and resized to be an equal size of 96×96 px. Finally, we analyse the facial affective of the input facial image using the proposed light-weight CNN architecture. We train two separate models for facial expression recognition and valence and arousal estimation tasks. The interface of the implemented mobile
application for facial affective computing is shown in Figure 6.5. The user only needs to grant permission for the application to use the front or back camera of mobile devices. The application could automatically detect the users’ faces and then feed the facial image into the pretrained CNN model to get the results of facial expression, valence and arousal in real-time which are shown on the bottom of the interface.

6.5.3 Evaluation of Storage Consumption and Processing Time

The storage consumption is an important metric for mobile development. For deep learning approaches, we need to train a fixed model and then deploy it on mobile devices to implement its functions. Compared with desktop devices, the storage space of mobile devices is small and limited. However, existing deep learning approaches were mainly designed for desktop devices, which didn’t consider storage consumption of the obtained training model. The size of the pretrained model is so large for mobile devices which can be up to hundreds of megabytes. Moreover, for mobile deployment, different development platforms such as iOS and Android also have different rules about the size of mobile applications. And for many mobile applications, the facial affective computing maybe only one of
the functions instead of primary functions, so its storage consumption should be small and reasonable. In order to evaluate the storage consumption of our proposed method, we compare with two classic CNNs in our experiments, AlexNet (Krizhevsky et al., 2012) and VGG16 (Simonyan & Zisserman, 2014), which are the classic networks for image classification tasks. As shown in Table 6.5, the model size of them is 233 MB and 528 MB respectively. Compared with two classic CNNs, our proposed light-weight CNN architecture has fewer parameters, and its model size is 15 MB. After transforming to Core ML model format, its size is only 8 MB, approximately two orders of magnitude smaller than two classic CNNs, which meets the requirements of mobile development.

Another problem is to balance performance and computation complexity. For deep learning methods, especially CNNs, the performance of networks depends on the depth of the network structure to some extent. A deeper network structure and higher computation complexity could lead to higher performance. However, high computation complexity means that we need a high-performance device. The mobile devices with hardware constraints cannot guarantee the processing speed while retaining high performance especially for real-time applications. We test our implemented mobile application on two iOS mobile devices, iPhone 5s and iPhone X.
X, which used to represent low-performance and high-performance devices respectively. The results are displayed in Table 6.6. Although the codes of application are non-optimized, the average running speed on iPhone 5s and iPhone X are 31 fps and 60 fps respectively. For iPhone 5s, the memory consumption is about 100 MB which accounts for 10% of its total memory (1 GB). Since iPhone X has a larger memory than iPhone 5s, the system allocates more memory (150 MB) to the mobile application which accounts for 5% of its total memory (3 GB). This means that our proposed light-weight CNN architecture is well suitable for real-time applications on mobile devices.

6.6 Discussion

In this chapter, the proposed light-weight CNN architecture achieves high performance when evaluated on the AffectNet database. It can well balance the performance and computational complexity on mobile devices for real-time facial affective computing tasks. The AffectNet database used for training and testing by the proposed method is collected from the wild environment, which provide data variations of many real-life mobile applications. However, we notice that some issues still need to be discussed. First, the proposed method works well when the facial images are visible and frontal, and no occlusion from other objects. However, in some special cases, the method cannot obtain the right results of facial affective. Therefore, designing more effective networks could potentially improve the performance to handle more varied situations from the wild environment. But at the same time, the complexity of the network will also increase. We thus need to find a balance between performance and efficiency on mobile devices. Second, another factor that affects performance is the choice of the face detection method. In our experiment, we used the Viola-Jones face detection algorithm which is a common and popular method but it cannot work well in some cases. One solution is to use the more advanced and effective face detection methods which have been
implemented in recent years to lead to good performance. For example, for iOS platforms, Apple provides the Vision framework including a deep learning-based face detection method which can be used in mobile applications in future.

### 6.7 Summary

We have proposed a light-weight CNN architecture for real-time facial affective computing on mobile devices. The key design principle of proposed network architecture is to minimize the number of parameters and computational complexity. This network uses facial images as input and outputs the categories of facial expression and values of valance and arousal. Compared to conventional CNNs, the proposed method well balances the high performance and low computation complexity. Moreover, the performance of the proposed method outperforms a series of existing methods in our experiment. We have also implemented a real-time facial affective computing mobile application that has a low consumption of memory and storage on actual mobile devices to demonstrate the feasibility of the proposed method for mobile development.
Chapter 7

Conclusion and Future Work

7.1 Contributions

This thesis focused on two fundamental subtasks in facial behaviour analysis for affective computing in the wild which are eye analysis and facial expression analysis. The aim of this thesis is to enable facial behaviour analysis for affective computing closer to reality. After extensively reviewing the principal studies in related areas, it identified the problems and challenges and resorted to the technologies of deep learning for robust solutions. Its main contributions are summarised as follows:

In Chapter 3, this thesis first focuses on the subtask of eye analysis in facial behaviour analysis. The thesis proposes a fully convolutional network (FCN) for eye centre localization. The proposed network treats eye centre localization as a special subproblem of the task of semantic segmentation, which can accurately locate the eye centre. Experiments on two challenging databases show that the proposed method has higher accuracy of eye centre localization compared with the state-of-the-art approaches. More importantly, it opens up a novel and promising future direction for this area of research.

From Chapter 4, this thesis turns to address another important subtask of facial behaviour analysis, namely facial expression analysis. It proposes to use a novel relation-aware facial expression recognition method called Relation Convolutional Neural Network (ReCNN) to adaptively capture the relationships between crucial regions and facial expressions and focus on the most discriminative regions for recognition. ReCNN utilizes the relation module to compute the relation weight which reflects the importance of crucial regions to facial expressions. It further
uses the relation weight to generate weighted features as a final representation for facial expression. After being trained on a large-scale and in-the-wild facial expression database, ReCNN can accurately recognize the categories of facial expressions, which has superior recognition accuracy compared with state-of-the-art approaches. Evaluations also indicated that the relationship between crucial regions and facial expressions is beneficial for performance improvement of facial expression recognition. This provides an efficient alternative to existing methods that normally depend on the whole face for recognition.

Encouraged by the outcomes of Chapter 4, this thesis continues to explore the role of crucial facial regions in facial expression synthesis in Chapter 5. For addressing the problem of insufficient training data for facial expression recognition, it proposes a novel end-to-end method called Local and Global Perception Generative Adversarial Network (LGP-GAN) for synthesizing facial expressions. The core of the proposed method is the two-stage cascaded architecture that divides the facial expression synthesis process into local facial region generation and global facial image generation. Specifically, the local network is first used to capture texture details of the crucial facial regions and then the global network learns the general structure and profile of the face. The purpose of this design is to fully utilize both the local and global facial information to synthesize facial expressions step by step. The proposed network has been showed state-of-the-art performance on the public database in qualitative and quantitative experiments on the task of facial expression synthesis.

Finally, a case study towards exploring mobile facial affective computing is presented in Chapter 6. It proposes a light-weight CNN architecture for mobile development, which well balances the performance and computational complexity. Evaluations demonstrate that this network structure offers higher performance than state-of-the-art approaches on the public database. To validate its feasibility for mobile development, it is implemented and ported to an actual mobile device. The evaluation on the mobile device demonstrates that the proposed network is
well-suited for mobile which can high maintain running speed without taking up too much memory or storage space.

In conclusion, this thesis developed novel and robust algorithmic solutions based on deep learning to facial behaviour analysis for affective computing in the wild. It mainly focuses on addressing the problems and challenges of two subtasks of facial behaviour analysis which are eye analysis and facial expression analysis. We believe that this thesis will provide valuable experience on adapting the technologies of deep learning to facial behaviours analysis task and bring more interested investigators into this study. Moreover, we hope that relying on the recent progress in deep learning will be beneficial to the further development of facial behaviour analysis for affective computing in the wild.

### 7.2 Future Work

In this section, we first discuss the limitations of the proposed method in each chapter of this thesis and then provide brief analysis for future directions.

1) It can be found that the proposed method for eye centre localization in Chapter 3 is a shallow architecture because annotated training data for eye centre localization is limited. A deeper network structure and more training data can significantly improve the network performance.

2) Although the proposed ReCNN in Chapter 4 has superior recognition performance on the in-the-wild databases which outperforms state-of-the-art methods, it still has limited performance on nonfrontal facial images. In addition, the proposed ReCNN is a deep and complex network architecture, which is not well-suited for mobile development.

3) For facial expression synthesis, the proposed LGP-GAN in Chapter 5 can work well on the lab-collected datasets, but it cannot perform perfectly sometimes for the facial expressions from natural and un-controlled conditions. Moreover, the facial images used in this chapter are labelled by eight discrete categories of facial
expressions. Thus, the proposed method can only generate discrete facial expressions. However, not all facial expressions can be represented by discrete categories which ignore the intensity levels of facial expressions.

4) In Chapter 6, the designed light-weight CNN is well-suited for mobile devices, which has high performance and low computational complexity. However, there are still some gaps in recognition performance compared to some state-of-the-art approaches. Considering there are some relations between the category and intensity of facial expressions, these relations could further improve the performance. Moreover, the proposed method can only work well when the facial images are visible and frontal, and no occlusion from other objects. However, in some special cases, the method cannot obtain the right results.

Targeting at addressing the aforementioned limitations, there are some promising directions for future work to extend this thesis:

1) For eye centre localization, we will explore how to use synthetic methods to generate data of eye centre for training in the future. And design a deeper network structure to further improve the performance. Since we only focus on the eye centre localization in this thesis, another future work is to implement eye gaze estimation using the accurate location of the eye centre.

2) In order to address the issue of nonfrontal facial images, we will explore the use of face frontalization method (Yiming Wang et al., 2021; Y. Wang et al., 2017; Yiming Wang et al., 2016) to improve its recognition performance for nonfrontal faces. Moreover, we plan to implement ReCNN on mobile devices. In the future, we will optimize the network structure and design a simplified version. The simplified version will be easy to port to mobile devices, which maintains high performance and low consumption of computing resources.

3) The future work for facial expression synthesis is to explore how to adapt LGP-GAN to unconstrained facial expression datasets for expression synthesis considering different intensity levels of facial expressions.

4) For affective computing on mobile devices, future work will further improve system functions of facial behaviour analysis. We will use multi-task learning
which implements the target task in one network without two separated networks by using the relations between the category and intensity of facial expressions. For some special cases, we will use the more advanced and effective face detection methods or face frontalization methods to process the facial images. In addition to recognizing facial expressions, the function of real-time eye gaze estimation will be implemented and integrated into this system.
References


135


141


Appendix

FORM UPR16
Research Ethics Review Checklist
Please include this completed form as an appendix to your thesis (see the Research Degrees Operational Handbook for more information)

<table>
<thead>
<tr>
<th>Postgraduate Research Student (PGRS) Information</th>
<th>Student ID: 880590</th>
</tr>
</thead>
<tbody>
<tr>
<td>PGRS Name: Yifan Xia</td>
<td></td>
</tr>
<tr>
<td>Department: CTS</td>
<td>First Supervisor: Prof. Hui Yu</td>
</tr>
<tr>
<td>Start Date: 01/02/2018 (or progression date for Prof Doc students)</td>
<td></td>
</tr>
<tr>
<td>Study Mode and Route: Part-time ☐</td>
<td>MPni ☐</td>
</tr>
<tr>
<td>Full-time ☒</td>
<td>PhD ☒</td>
</tr>
<tr>
<td>Title of Thesis: Machine Analysis of Facial Behaviour for Affective Computing</td>
<td></td>
</tr>
<tr>
<td>Thesis Word Count: 44732 (excluding ancillary data)</td>
<td></td>
</tr>
</tbody>
</table>

If you are unsure about any of the following, please contact the local representative on your Faculty Ethics Committee for advice. Please note that it is your responsibility to follow the University’s Ethics Policy and any relevant University, academic or professional guidelines in the conduct of your study.

Although the Ethics Committee may have given your study a favourable opinion, the final responsibility for the ethical conduct of this work lies with the researcher(s).

UKRIO Finished Research Checklist:
(If you would like to know more about the checklist, please see your Faculty or Departmental Ethics Committee reps or see the online version of the full checklist at: http://www.ukrio.org/what-we-do/code-of-practice-for-research)

a) Have all of your research and findings been reported accurately, honestly and within a reasonable time frame? YES ☒ NO ☒

b) Have all contributions to knowledge been acknowledged? YES ☒ NO ☒

c) Have you complied with all agreements relating to intellectual property, publication and authorship? YES ☒ NO ☒

d) Has your research data been retained in a secure and accessible form and will it remain so for the required duration? YES ☒ NO ☒

e) Does your research comply with all legal, ethical, and contractual requirements? YES ☒ NO ☒

Candidate Statement:
I have considered the ethical dimensions of the above named research project, and have successfully obtained the necessary ethical approval(s).

Ethical review number(s) from Faculty Ethics Committee (or from NRES/SCREC): ETHIC-2019-445

If you have not submitted your work for ethical review, and/or you have answered ‘No’ to one or more of questions a) to e), please explain below why this is so:

Signed (PGRS): [Signature] Date: 21/04/2021

UPR16 – April 2018
Certificate of Ethics Review

Project Title: Real-time Facial and Eye Gaze Animation Using a Single RGB Camera
Name: Yilun Xia  User ID: 880950  Application Date: 16-Apr-2019 12:00  IR Number: ETHC-2019-445

You must download your certificate, print a copy and keep it as a record of this review.

It is your responsibility to adhere to the University Ethics Policy and any Department, School or professional guidelines in the conduct of your study including relevant guidelines regarding health and safety of researchers and University Health and Safety Policy.

It is also your responsibility to follow University guidance on Data Protection Policy:
- General guidance for all data protection issues
- University Data Protection Policy

You are reminded that as a University of Portsmouth Researcher you are bound by the UK BBSRC Code of Practice for Research; any breach of this code could lead to action being taken following the University’s Procedure for the Investigation of Allegations of Misconduct in Research.

Any changes in the answers to the questions reflecting the design, management or conduct of the research over the course of the project must be notified to the Faculty Ethics Committee. Any changes that affect the answers given in the questionnaire, not reported to the Faculty Ethics Committee, will invalidate this certificate.

This ethical review should not be used to infer any comment on the academic merits or methodology of the project. If you have not already done so, you are advised to develop a clear protocol/proposal and ensure that it is independently reviewed by peers or others of appropriate standing. A favourable ethical opinion should not be perceived as permission to proceed with the research, there might be other matters of governance which require further consideration including the agreement of any organisation hosting the research.

(A1) Please briefly describe your project. The aim of my project is to develop a novel real-time facial animation system, which is able to capture 3D facial and eye gaze performances only using a single RGB camera. The system will be validated on some publicly available datasets which have ground truth data annotated by human subjects and comprehensively compared with state-of-the-art approaches.

(A2) What field do you belong to? CCI

(A3) I am sure that my project requires ethical review by my Faculty Ethics Committee because it includes at least one material ethical issue.

(A5) Has your project already been externally reviewed? No

(B1) Is the study likely to involve human participants? No

(B2) Are you certain that your project will not involve human subjects or participants? Yes

(C6) Is there any risk to the health & safety of the researcher or members of the research team beyond those that have already been risk assessed? No

(D2) Are there risks of damage to physical and/or ecological environmental features? No

(D4) Are there risks of damage to features of historical or cultural heritage (e.g. impacts of study techniques, taking of samples)? No

(E1) Will the study involve the investigator and/or any participants in activities that could be considered contentious, unacceptable, or illegal, or in any other way harmful to the reputation of the University of Portsmouth? No

(F2) Could the research outputs potentially be harmful to third parties? No

(G1) Please confirm that you have read the University Ethics Policy and have considered the implications for your project: Confirmed

(G2) Please confirm that you have read the UK BBSRC Code of Practice for Research and will conduct your project in accordance with it: Confirmed

(G3) The University is committed to the Concordat to Support Research Integrity: Confirmed

(G4) Submitting false or incorrect information is a breach of the University Ethics Policy and may be considered as misconduct and be subject to disciplinary action. Please confirm you understand this and agree that the information you have entered is correct: Confirmed