

國立臺灣大學電機資訊學院電機工程學研究所



碩士論文

Graduate Institute of Electrical Engineering
College of Electrical Engineering and Computer Science
National Taiwan University
Master Thesis

使用深度時域對比網絡之人臉情緒辨識

Deep Temporal-Contrastive Network for
Facial Expression Recognition

黎子駿

Zi-Jun Li

指導教授：傅立成 博士

Advisor: Li-Chen Fu, Ph.D.

中華民國 107 年 8 月

August, 2018

誌謝



赴台學習的這兩年，在不同文化背景里進行學術研究，得到的不僅是知識量的提升、知識領域的拓展和解決問題能力的提升，更多的是學會兼容並包、求同存異。

首先要感謝的是指導老師，傅立成教授，老師對學術研究的嚴謹和科學的懷疑精神讓我印象深刻；即便很忙也會參加每週的會議，與我們分享和討論自己的所見所聞，同時提醒我們要保持追求卓越的態度。除了學術上的指導，老師在為人處世、人生觀等方面也給予我很多的啟發。

感謝 ACL 和資工系的各位，承蒙各位的厚愛，讓我在這兩年時光快樂、充實地度過。感謝安陞學長的協助，跟你討論後總是能讓我找到有新的靈感，並且不辭勞苦幫忙檢查我的投稿。感謝俊緯、大偉、光耀三位 AFM 的學長，在我初來台灣、剛進實驗室時關心我、幫助我適應新的環境。感謝柏文提高我桌球的能力，在我趕投稿時借電腦給我做實驗。特別要感謝孟皓、佐新和侑寰，這三位戰友陪伴了我兩年，熬夜過，討論過，也經常一起出去玩，和你們共度的喜怒哀樂是我在台灣最棒的回憶。感謝宇閎學弟在學術和生活上給我的幫助，三個學長的研究成果都要靠你傳承下去，能力越大，責任越大，希望你能勇往直前，青出於藍。

感謝台北同鄉會肯定我在學業上的表現頒發給我獎學金，讓我在碩二下學期衣食無虞，可以更專注在學術研究上。

最後感謝我的父母在這二十多年來的辛苦付出，感謝你們當初堅持讓我讀研，也要感謝在對岸默默等我兩年、支持我的女朋友，你們的期待是我奮鬥的動力；還要感謝各位支持我的親戚，希望我的表現沒有讓你們失望，未來我會繼續努力！

子駿 May 12, 2018

摘要



臉部情緒反映了人類心理活動，因此情緒識別是人機互動的關鍵要素。臉部情緒識別，甚至對於人類來說，也是一個具有挑戰性的任務。這主要是因為每個人都

有自己表達情緒的強度和方式。為了從不同的個體裡提取出各種表情的共性，個體的個性造成對情緒判別的影響要盡可能地縮小。

在本論文中，我們提出使用時域對比的深度網絡來實現一個基於視頻的臉部情緒辨識系統。該深度網絡利用時域上的特徵來減少個體個性造成的影響。外表特徵和幾何特徵分別從人臉照片和人臉關鍵點的坐標通過卷積神經網絡（CNN）和深度神經網絡（DNN）提取出來。為了使模型從相鄰幀（情緒類別、強度相似）提取出來的特徵是相似的，我們使用了額外的損失函數。緊接著，我們通過比較視頻幀在高維空間的距離來挑選出一段視頻中最有代表性的兩幀。我們利用那兩幀在高維空間中的對比表達來做情緒分類。

我們使用聯合微調來結合以人臉照片和人臉關鍵點作為輸入的兩個模型。兩個模型相輔相成，使得整個系統得到更好的識別率。

我們在兩個廣泛使用在情緒識別的數據集（CK+和 Oulu-CASIA）進行實驗。實驗結果體現出我們提出的方法能夠有效地提取出關鍵幀，而且在情緒識別準確率上優於現今較好的方法。

關鍵字：臉部情緒辨識、卷積神經網絡、對比表達

ABSTRACT



Facial expression reflects psychological activities of human and it is key factor in interaction between human and machines. Facial expression recognition is a challenging task even for human since individuals have their own way to express their feelings with different intensity. In order to extract commonality of facial expressions from different individuals, personality effect of individual needs to be minimized as much as possible.

In this thesis, we construct a video-based facial expression recognition system by using a deep temporal-contrastive network(DTCN) that utilizes the temporal feature to remove the personality effect. Appearance and geometry feature are extracted by CNN and DNN from face image and coordinate of facial landmark, respectively. In order to let our CNN framework be able to extract similar features from adjacent frames, special loss function is introduced. Then, the two most representative frames of a video/image sequence are picked out through comparison of distances among frames. Facial expressions can be classified by the so-called contrastive representation between expressions of those two key frames in high dimension space.

We utilize joint fine-tuning to combine two models which take face image and facial landmark as input, respectively. Those two models are complementary and the recognition accuracy is improved by this combination.

We conducted our experiment in the most widely used databases (CK+ and Oulu-CASIA) for facial expression recognition. The experiment results show that the proposed method outperforms those from the state-of-the-art methods.

Keywords: Facial Expression Recognition, Convolution Neural Network, Contrastive Representation.

TABLE OF CONTENTS

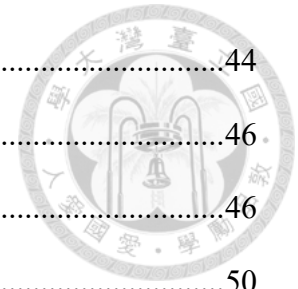


口試委員會審定書	#
誌謝	I
摘要	II
ABSTRACT	III
TABLE OF CONTENTS	IV
LIST OF FIGURES	VII
LIST OF TABLES	IX
Chapter 1 Introduction	1
1.1 Motivation	1
1.2 Literature Review	3
1.2.1 Facial Analysis	3
1.2.2 Facial Expression Recognition and Detection	4
1.2.3 Image-Based and Video-Based Methods for FER	5
1.3 Contribution	9
1.4 Thesis Organization	10
Chapter 2 Preliminaries	11
2.1 Face Detection	11
2.1.1 Introduction to Face Detection	11
2.1.2 Face Detection Based on Hand-Crafted Feature	11
2.1.3 Face Detection Based on Deep Learning	13
2.2 Face Alignment	14
2.2.1 Introduction to Face Alignment	14



2.2.2	Facial Landmark Localization.....	15
2.2.3	Face Alignment and Warping.....	16
2.3	Convolution Neural Network	17
2.3.1	Introduction to Convolution Neural Network.....	17
2.3.2	CNN for Facial Analysis	20
Chapter 3	Facial Expression Recognition.....	22
3.1	Preprocessing.....	22
3.2	Structure of DTCN	24
3.3	Temporal-Contrastive Appearance Network	25
3.3.1	Transfer Learning for TCAN	25
3.3.2	Contrastive Representation of TCAN	27
3.3.3	Training Process of TCAN.....	29
3.3.4	Loss Function.....	30
3.4	Temporal-Contrastive Geometry Network	34
3.4.1	Architecture of TCGN.....	34
3.4.2	Contrastive Representation of TCGN	35
3.4.3	Training Process of TCGN.....	37
3.5	Deep Temporal-Contrastive Network.....	38
3.5.1	DTCN: Combination of TCAN and TCGN	38
3.5.2	Attributes of DTCN.....	40
Chapter 4	Experiment	41
4.1	Configuration.....	41
4.2	Description of Dataset and Evaluation	42
4.2.1	The Extended Cohn-Kanade (CK+).....	42
4.2.2	Oulu-CASIA	43

4.2.3	Evaluation	44
4.3	Quantity Results.....	46
4.3.1	Results on CK+	46
4.3.2	Results on Oulu-CASIA.....	50
4.4	Quality Analysis.....	54
Chapter 5	Conclusion and Future Work.....	57
REFERENCE	58



LIST OF FIGURES



Figure 1-1 Sample of user interface in the Android App.....	2
Figure 1-2 Structure of Generated Contrastive Network	6
Figure 1-3 Pipeline of trivial CNN-LSTM method.....	8
Figure 2-1 Haar-like feature used for Viola-Jones face detector.....	12
Figure 2-2 Architecture of Single Shot Scale-invariant Face Detector	13
Figure 2-3 Facial landmark localization.....	15
Figure 2-4 Face alignment and warping.....	16
Figure 2-5 Convolution in convolutional layer	18
Figure 2-6 Enhancing and Cropping network	20
Figure 3-1 Pipeline of preprocessing.....	23
Figure 3-2 Structure of DTCN	24
Figure 3-3 Architecture of TCAN	27
Figure 3-4 End to end training of TCAN	29
Figure 3-5 Triplet loss	32
Figure 3-6 Architecture of TCGN	35
Figure 3-7 Visualization of contrastive representation.....	37
Figure 3-8 Joint fine-tuning of DTCN.....	39
Figure 4-1 Faces of one image sequence in FER datasets (CK+).....	43
Figure 4-2 Images in NI and VL systems under three illumination conditions.....	43
Figure 4-3 Individual independent 10 fold-cross validation (CK+).....	45
Figure 4-4 The comparison of our TCAN/TCGN/DTCN in CK+	47
Figure 4-5 Confusion matrix of our DTCN in CK+.....	50
Figure 4-6 The comparison of our TCAN/TCGN/DTCN in Oulu-CASIA.....	51

Figure 4-7 Confusion matrix of our DTCN in Oulu-CASIA54

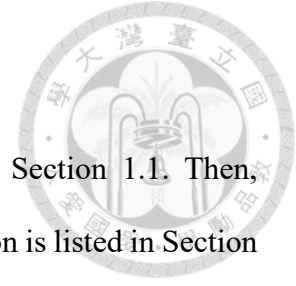


LIST OF TABLES



Table 1-1 Basic facial expression and the corresponding action units	5
Table 3-1 Three extra loss functions applied in facial expression recognition.....	33
Table 3-2 Comparison of three kinds of method for facial expression recognition	40
Table 4-1 The Specification of the computer for our experiments for FER.....	41
Table 4-2 The distribution of seven emotions in CK+	42
Table 4-3 The distribution of six emotions in Oulu-CASIA	44
Table 4-4 Accuracy of our TCAN with different loss in the CK+ database.....	47
Table 4-5 The accuracy of each expression in CK+ by our models	48
Table 4-6 Overall accuracy in the CK+ database	49
Table 4-7 Accuracy of our TCAN with different loss in the Oulu-CASIA database	51
Table 4-8 The accuracy of each expression in Oulu-CASIA by our models.....	52
Table 4-9 Overall accuracy in the Oulu-CASIA database	53
Table 4-10 The key frames picked by TCAN and its predictions in CK+	55
Table 4-11 The key frames picked by TCAN and its predictions in Oulu-CASIA	56

Chapter 1 Introduction



In this chapter, we describe the motivation of this work in Section 1.1. Then, literature review of this thesis is presented in Section 1.2. Contribution is listed in Section 1.3, and in Section 1.4, we give the organization of this thesis.

1.1 Motivation

High speed development of the world provides us with unprecedented convenience and enhances our life quality. Advanced technologies usually save our effort and make our work/life more efficient. However, modern life with advanced technologies may bring some problems at the same time. Declining birthrate, aging society and mental diseases caused by overwork preoccupy many highly developed countries and regions. Overwork people cannot outlet their pressure and do not have enough time and energy to take care of their children and parents. As a result, the incidence of mental diseases is increasing, children and elders are lacking of love and care. In order to solve or alleviate those problems, researchers started to focus on technology for both physical and psychological health of human.

To serve human better, technology with users' psychology as auxiliary information are more friendly to human. Facial expression reflects psychological activities of human, and therefore, it is key factor in interaction between human and machines. Recognized facial expressions are significant feedback for Artificial Intelligence (AI) system such as a robot, HRI system, chatbot to understand and reason out the thinking and next action of human users. [1] captures the emotional reaction of children to a behavior performed by the robot and those affective states are used by the Interaction Reinforcement Learning(IRL). The user's interface of [1] is shown in Figure 1-1 and RoBoHon is deployed to perform the experiment with the child participants.

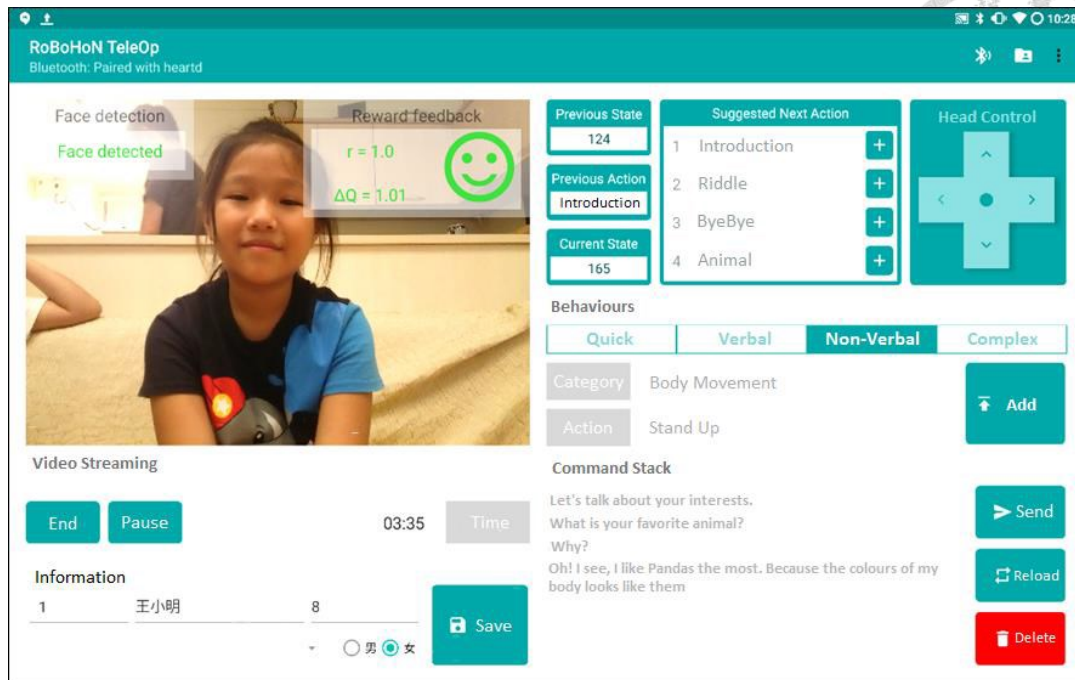
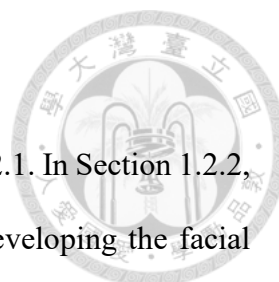


Figure 1-1 Sample of user interface in the Android App

For the purpose of recording users' mental states and better interaction between human and AI system, we design a video-based facial expression recognition system. In our system, facial expressions are divided into seven classes which are angry, contemptuous, disgusted, fearful, happy, sad and surprised, though the capability of our system is not limited to these seven emotions.

Facial expression recognition (FER) has been developed for a few past decades, and in Section 1.2.3, image-based and video-based methods for FER are compared and discussed. Those two kinds of method have their own advantages and drawbacks, so in this thesis, we propose Deep Temporal-Contrastive Network(DTCN) that combines the advantage of those two kinds of methods and try our best to avoid their drawbacks.

To sum up, in order to record users' mental states and obtain better interaction between human and AI system, we propose a video-based facial expression recognition system by using deep temporal-contrastive network.



1.2 Literature Review

We first give an introduction to facial analysis task in Section 1.2.1. In Section 1.2.2, we focus on facial expression and briefly introduce the trend of developing the facial expression recognition and detection techniques. Related works of facial expression recognition are compared and discussed in Section 1.2.3

1.2.1 Facial Analysis

From the history of human, all technologies should be developed to serve human and thus most focus on human. With the development of computer science, machines are able to capture feature of human by the data recorded by different sensors, *e.g.*, RGB camera, depth camera, *etc.* Facial analysis is significant to human oriented applications since face contains rich information, such as identity, gender, expression, age, health status, and some personal attributes. Efficient use of these face information is the key to era of AI.

Facial analysis is a classic topic for researchers major in computer science. Face verification or identification [2], facial expression recognition [3-5], face alignment [6], face action unit detection [7] are all main topics of facial analysis. Some of them are robust and mature enough to be applied to real applications, *e.g.*, identity identification of customs, searching for criminals, facial special effect, *etc.*

In the early stage of computer vision, researchers try to design some fixed feature, such as SIFT, HOG, LBP, *etc.*, and classify items by the extracted feature. Those methods are not robust because they extract the same feature for all kinds of tasks but fail to be specified to each kind of task, which makes the methods only suboptimal. Then, the fast development of convolution neural networks (CNN) overcomes the shortcomings of hand-crafted feature and CNN has widely used in image-based tasks, including facial

analysis. We introduce CNN in details in Section 2.3.

1.2.2 Facial Expression Recognition and Detection

Facial expressions reveal a person's internal state, psychopathology, and social behavior. Basic facial expressions can be so many kinds, such as angry, contemptuous, disgusted, fearful, happy, sad, and surprised. Facial expressions are made up of subtle motion of facial muscles, so one way to classify facial expressions is to describe expressions in terms of configuration and intensity of the so-called facial action units (AUs) using the Facial Action Coding System (FACS) [8]. Table 1-1 shows the basic expressions composed of different combinations of AUs. Take "happy" as an example, this emotion is composed of AU 6 and AU 15, which represent raising of cheek and pulling of lip corner, respectively.

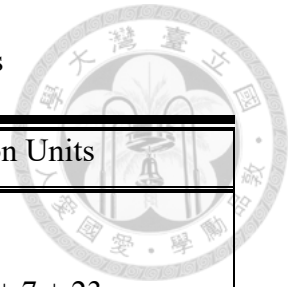
Another way to recognize facial expressions is similar to other image-based tasks: classify facial expressions by extracted feature from faces. There are some approaches which utilize hand-crafted features, such as HOG [9], SIFT [10], LBP [11, 12], and classify emotions by SVM with those feature. But compared with deep methods [5,6,7,8,9,10] in recent years, the performance of hand-crafted feature-based ones is barely satisfactory.







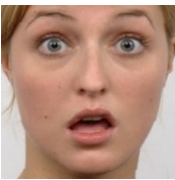
If we classify facial expression frame by frame, there is nothing different between facial expression recognition and detection. However, image-based recognition/detection discards all temporal information and is not stable compared with video-based methods.

In video-based facial expression recognition, a testing data with short length (usually 10~50 frames) should contain only one kind of expression. In video-based facial expression detection, a testing data with arbitrary length may contain no expression or several kinds of expressions and we need to predict the beginning and ending frames of each expression. Therefore, video-based detection is more challenging than video-based



Table 1-1 Basic facial expression and the corresponding action units



Face image	Basic emotion	Action Units
	Angry	4 + 5 + 7 + 23
	Contemptuous	R12A + R14A
	Disgusted	9 + 15 + 16
	Fearful	1 + 2 + 4 + 5 + 7 + 20 + 26
	Happy	6 + 12
	Sad	1 + 4 + 15
	Surprised	1 + 2 + 5B + 26

recognition.

1.2.3 Image-Based and Video-Based Methods for FER

Recent deep methods for facial expression recognition (FER) basically can be divided into image-based and video-based ones. Comparison between those methods is



discussed as follows:

1) *Image-based methods*

Image-based methods recognize emotions by a single image, and thus its training and testing are very fast. However, judgement of emotions through one image may not be stable. Image-based methods cannot capture the gradual changes of facial expression, which means they may not recognize mild expression. For example, Peak-Piloted Deep Network (PPDN) [5] is trained with peak expression face and non-peak expression face as input and drives the features of the non-peak expression towards those of the peak expression. PPDN achieves a great accuracy of peak expression recognition, but relatively poor accuracy of non-peak expression.

To overcome the instability of image-based methods, people try to utilize the comparison between different expressions and to capture the facial changes. Generated Contrastive Network (GCN) [4] imitates how human recognizes facial emotion: a reference face is generated by a generator net, and GCN utilizes contrastive representation between input face and reference face to classify the emotion, as shown in Figure 1-2.

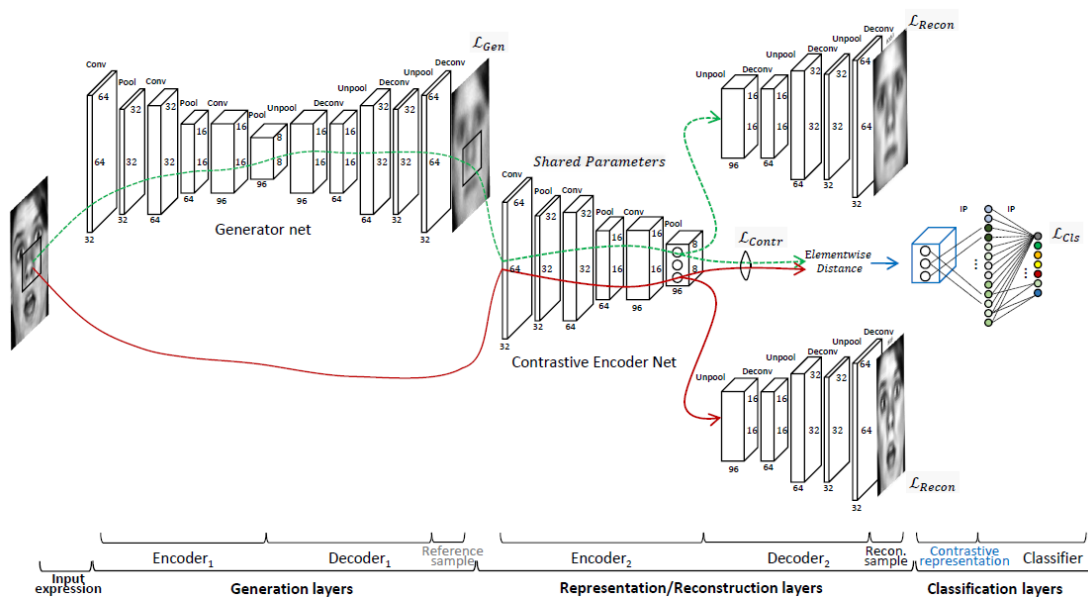


Figure 1-2 Structure of Generated Contrastive Network

Though the performance of GCN is outstanding, the generated reference image is more similar to the input image instead of the ground-truth, which means the classification results are not really obtained by comparing the current expression and the other expression. Therefore, it is paradoxical with its theory for recognizing emotion.

The image-based methods can be summarized as: use one single image as input to prevent interference in emotion recognition by repeated and redundant information; training and testing are very fast compared with video-based methods. However, these methods are not stable because they classify emotion with only one image; what's more, temporal features, which is useful in capturing small changes of the facial expression, are all discarded.

2) *Video-based methods*

Compared with image-based methods, the video-based methods focus on how to capture temporal information. The work in [13] combines two models, namely, 3D convolution and CNN-RNN to capture the temporal feature. For the others work, [3] is a two-stream network, where the first stream with face image taken as input utilizes 3D convolution to extract temporal feature, while the second stream with coordinates of facial landmarks taken as input just uses dense neural network to handle the temporal and spatial feature; [14] extracts spatial feature by Inception-ResNet and temporal feature by Conditional Random Field (CRF); [15] encodes video frames in two dimensions and estimates the pain intensity by recurrent convolution; [16] uses Inception-ResNet with facial landmarks to extract feature and capture temporal feature by long-short-term-memory (LSTM).

However, 3D convolution for spatial-temporal feature extraction is hard to train due to the huge computation and lack of pre-train model. LSTM takes all the video frames as input, which causes large computation and redundant information, and may interfere the

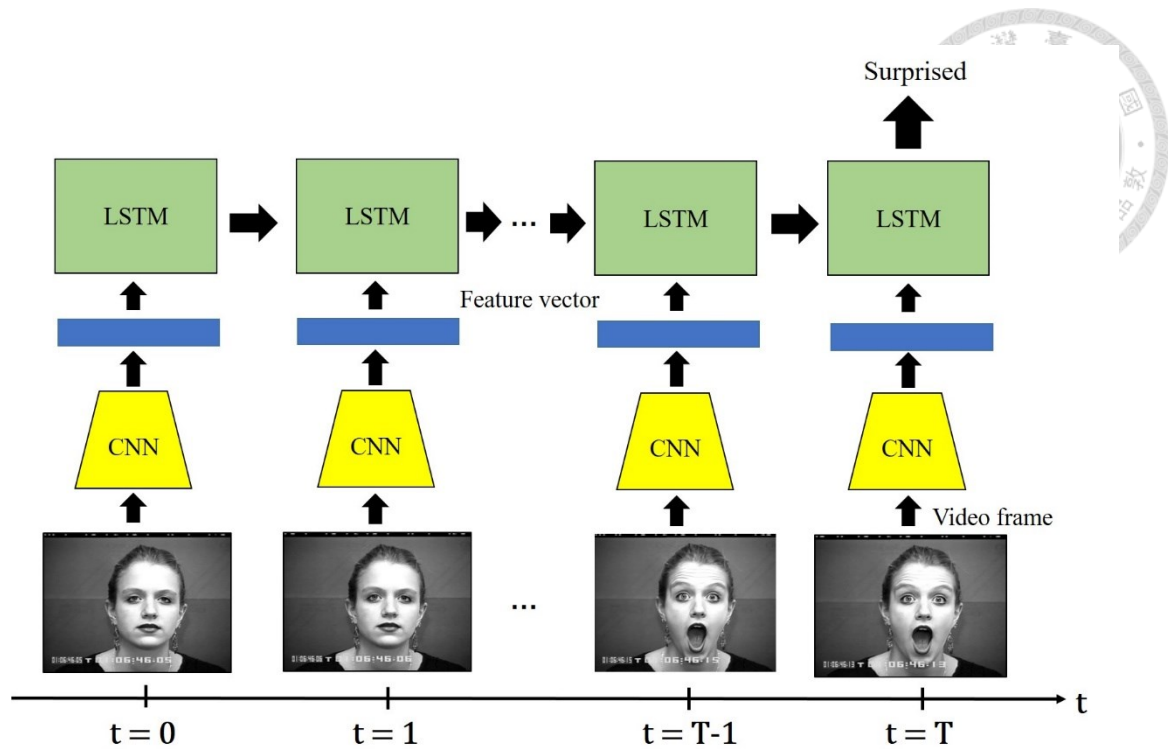


Figure 1-3 Pipeline of trivial CNN-LSTM method

final result.

Video-based methods are more stable since their results involve all the frames instead of one single image frame. However, methods mentioned above are taking all frames in the video as input which cost massive computation, and it turns out that the excessive information may disturb the final result.

Besides, repeated and redundant input causes huge computation cost in training and testing. For example, we show the CNN-Long Short Term Memory (LSTM) method for facial expression recognition in Figure 1-3. The model not only need to extract feature vector of each frame, but also need to utilize LSTM to integrate the temporal feature by all the input frames. The video-based methods take days or even weeks to train their models, which is hard to tune hyper parameters to obtain the optimal solution.

Therefore, the motivation of this thesis is to combine the advantages of these two kinds of method and to mitigate their drawbacks.

1.3 Contribution

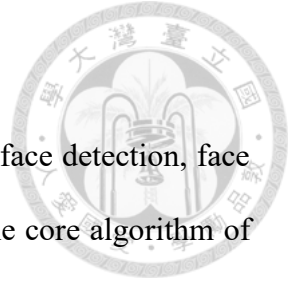
In this thesis, we propose a complete system for facial expression recognition. A novel network structure, deep temporal-contrastive network, is designed to extract the commonality of facial expression and to minimize personality effect as much as possible.

We reduce input frames to the classifier by using only two most representative frames. In this novel architecture, the redundant input information can be avoided. In other words, no longer all the frames in the input video will be needed to contribute to the result. Another benefit of this approach is that the bias of facial personality effect of individual can be removed by taking advantage of difference between two facial expressions, corresponding to two emotions of the same individual in the same video. The contributions of this work can be summarized in the followings:

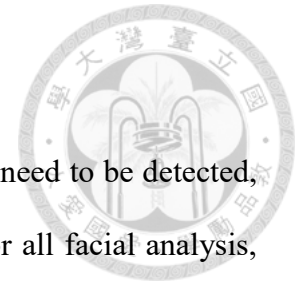
- I. We propose a novel and effective method to capture the temporal feature with moderate computation and save use of redundant information so as not to interfere the classification result.
- II. Our work takes input from a video, whose training process is relatively fast compared with traditional video-based methods, and is more stable than the image-based methods since its input contains temporal information.
- III. We apply contrastive loss and triplet loss in facial expression recognition (FER), which force our model to learn to extract the feature as our assumption during training process. The performances of extra loss functions are better validated in experiment.
- IV. We combine two streams with different inputs, sharing the same concept. The fused model achieves higher accuracy result compared with the state-of-the-art methods, validated in known datasets CK+ and Oulu-CASIA.

1.4 Thesis Organization

This thesis is organized as follows. In Chapter 2, we introduce face detection, face alignment and convolutional neural network(CNN). In Chapter 3, the core algorithm of this paper, deep temporal-contrastive network for facial expression recognition, is presented in details. In Chapter 4, we show the experiment results of facial expression recognition in several datasets. Conclusion and future work of this thesis is presented in Chapter 5.



Chapter 2 Preliminaries



Before recognizing facial expression, faces in the image/video need to be detected, cut and aligned. In this chapter, we first introduce the basic task for all facial analysis, face detection, and its solutions. Secondly, we briefly introduce facial alignment. At the last part of this chapter, we introduce the powerful and well-known feature extractor and classifier called convolution neural network (CNN).

2.1 Face Detection

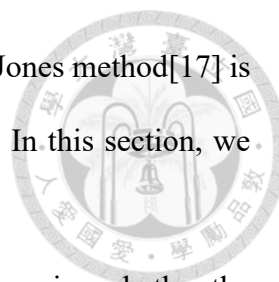
2.1.1 Introduction to Face Detection

The first step of all facial tasks should be tracing faces in an image or a video. Due to this reason, face detection has been studied over the past few decades and it has been well developed. In the early stage of the study in face detection, researchers tried to design robust handcraft features and trained an effective classifier. For example, Viola-Jones face detection[17], builds a simple and efficient classifier that selects a small number of important features(Haar-like) from a huge library of potential features using AdaBoost. We will give more details of hand-crafted feature-based method in Section 2.1.2.

In the recent years, the fast development and tremendous success of convolution neural network (CNN) also affect the research of face detection. CNN learns to extract useful feature to improve the accuracy or minimize the loss by end to end auto training. Thus, optimal solution of the training dataset can be obtained by using CNN. More details will be given in Section 2.3.

2.1.2 Face Detection Based on Hand-Crafted Feature

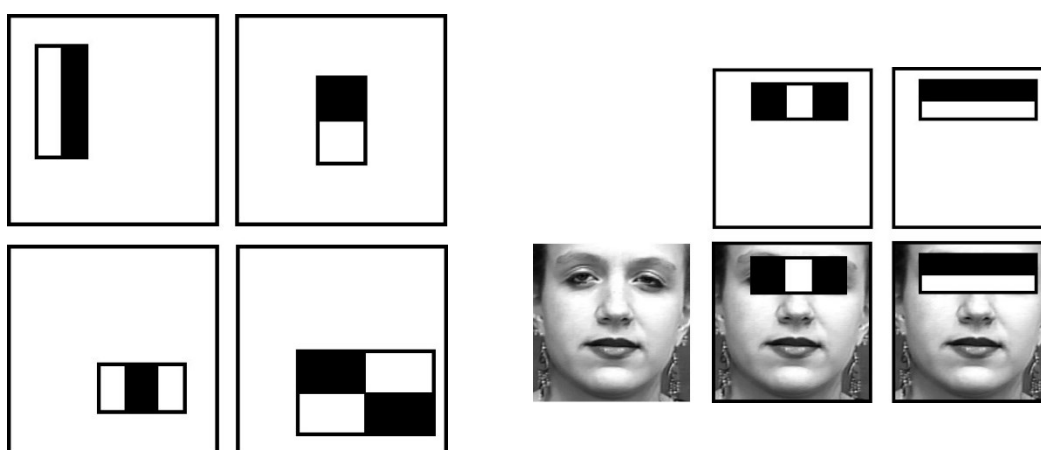
Face detection is the first and key step of many subsequent face related applications. In the early stage of image processing, hand-crafted features like SIFT, HOG, LBP are extracted to represent the image. Face detection has the similar development history.



Among those hand-crafted feature methods for face detection, Viola-Jones method[17] is the most famous and it is milestone in the history of face detection. In this section, we will focus on introduction of Viola-Jones face detection.

Viola-Jones method uses three kinds of Haar-like features to determine whether the region is a face. As shown in Figure 2-1(a), the features are composed by two/three/four rectangles, respectively. The regions have the same size and shape and are horizontally or vertically adjacent.

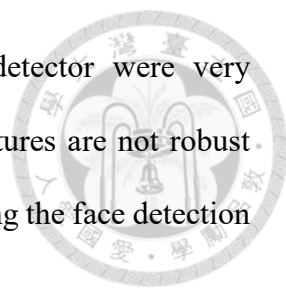
The reason why those filters can successfully classify face or not face is that it can capture useful and unique features of face. For example, in Figure 2-1(b), eyes are in the black regions of the first filter, whose sum of grayscale value is small, and the middle part between eyes is in the white region, whose sum of grayscale value is large. The value of the filter is the subtraction between the sum of the pixels in the black region and white region, thus the filter receives high response in that region. Due to similar reason, the second filter receives high response, too. A large number of filters are generated and some of them are selected by AdaBoost and cascaded as a strong classifier that achieves high detection accuracy with small computation.



(a) Example rectangle features

(b) Two features selected by AdaBoost

Figure 2-1 Haar-like feature used for Viola-Jones face detector



Hand-crafted feature-based methods like Viola-Jones face detector were very popular before the appearance of CNN. However, hand-crafted features are not robust and those methods optimize each component separately, which making the face detection system sub-optimal.

2.1.3 Face Detection Based on Deep Learning

CNN has a great capability in detection and recognition tasks. However, one of the drawbacks of traditional CNN is that it is time-consuming. Face detection is the basis of other facial analysis and it should not take too much time and computation. In order to accelerate face detection, small scale of CNN is designed. In [18], the proposed CNN cascade operates at multiple resolutions, quickly excludes the background regions in the low resolution stages, and carefully distinguish a small number of challenging candidates in the last high resolution stage.

Meanwhile, some researchers are inspired by some excellent works in object detection/recognition and have been successfully converted the architecture for object

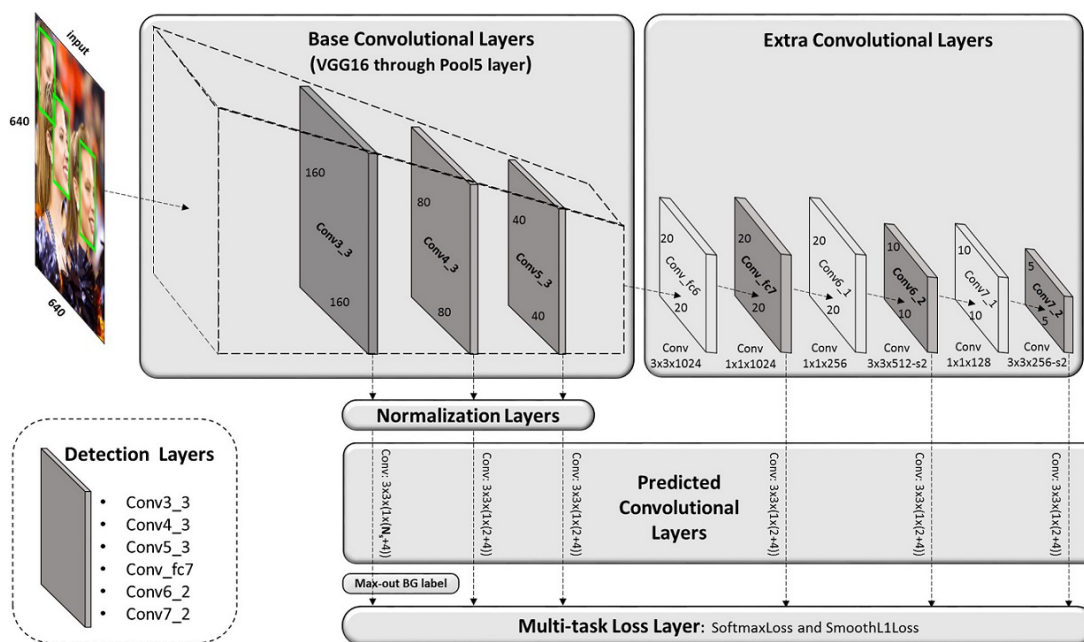


Figure 2-2 Architecture of Single Shot Scale-invariant Face Detector

detection, *e.g.*, You Only Look Once (YOLO) and Single Shot Detection(SSD), to solve face detection. Single shot scale-invariance detection[19] proposes an evolutionary architecture from single shot detection(SSD) for object detection. The architecture of [19] is shown in Figure 2-2. The proposed face detector has a wide range of anchor-associated layers as the figure shows, and a series of anchor scales in order to well handle different scales of faces.

With the development of deep learning, face detection with unconstrained environment becomes faster and better. More and more open sources for face detection are available for researchers focus on face analysis.

2.2 Face Alignment

2.2.1 Introduction to Face Alignment

Human head can row, yaw or pitch for a range of angle. In the real applications, the faces in images or videos can be arbitrary poses, which are not good input for facial analysis without face alignment. In order to make model robust to different scenarios, all facial image input should be warped into the same frontal pose.

Face alignment or facial landmarks localization on 2D images has received increasing attention owing to its comprehensive applications in automatic face analysis in the last two decades. However, it is extremely challenging in unconstrained environment because of occlusions and the variety of pose, lighting and expression. From an overall perspective, face alignment can be formulated as a task of searching over a face image for the pre-defined facial landmarks. In Section 2.2.2, we introduce facial landmarks localization and details of face alignment are shown in Section 2.2.3. Similar to face detection, face alignment is the basic of many facial analysis tasks since many facial tasks benefit from precise facial landmark localization.

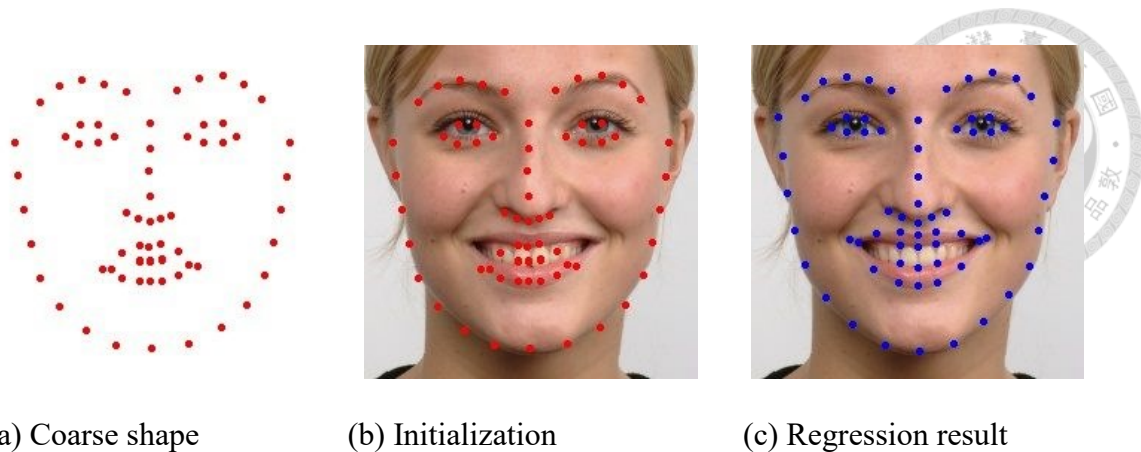


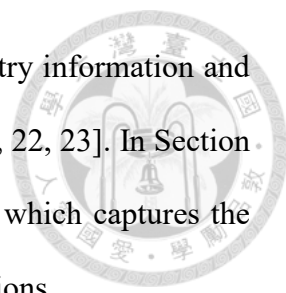
Figure 2-3 Facial landmark localization

2.2.2 Facial Landmark Localization

Facial landmarks are predefined points which are mainly located at the center or edge of the facial components, *e.g.*, eyes, mouth, eyebrows and nose, as shown in Figure 2-3(a) (The number of predefined facial landmarks in this thesis is 68).

Facial landmark localization typically starts from coarse initial shape, as shown in Figure 2-3(b) and we can see that those landmarks are not perfectly on their position, especially the points on the lip. Then the localization proceeds by refining the shape estimate step by step until convergence, as shown in Figure 2-3(c). Therefore, it actually is a regression problem that the initial points need to regress to the right position. [6] proposes a very efficient localization method, which extracts so-called local binary feature around the facial landmarks, iterates the process for 5 times and achieves 3000 FPS. Deep learning has been applied to facial landmarks localization [20, 21] as well and achieved great performance. We can utilize those deep learning methods if it is necessary. In facial expression recognition dataset, facial landmarks are not hard to be localized and normal methods are good enough for localization.

Facial landmark localization is the main step of face alignment and the quality of face alignment depends on the precision of the predicted facial landmarks. In next section, we introduce how to implement face alignment since we have the landmarks.



What's more, the structure of facial landmarks contains geometry information and those geometry feature can be used to recognize facial expression [3, 22, 23]. In Section 3.4, we propose a temporal-contrastive geometry network (TCGN) which captures the motion of each facial landmark and classifies emotions by those motions.

2.2.3 Face Alignment and Warping

In this thesis, in order to reserve more details of expression and avoid to interfere the structure of facial landmarks, we just rotate the faces and warp the face to fit the input size of our model. In the left face of Figure 2-4, the man's face is slightly twisting and the two red points on the face represent the eyes corners, which are detected at the facial landmarks localization. The red line is connecting the two eye corner and the angle between the red line and the horizontal line(blue) is denoted as θ . Let the coordinates of facial landmarks be $\{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$, where (x_i, y_i) denotes the coordinate of i^{th} facial landmark and N represents number of the predefined landmarks(N is 68 in this thesis). Then θ can be formulated as follow:

$$\theta = \tan^{-1}((x_r - x_l)/(y_r - y_l)) \tag{2-1}$$

where (x_l, y_l) and (x_r, y_r) are the coordinates of left and right eye corners, respectively.

Since the aligned faces should have the same pose, θ need to equal to zero (as shown in the right face of Figure 2-4). So the original image should be rotated $-\theta$ (the

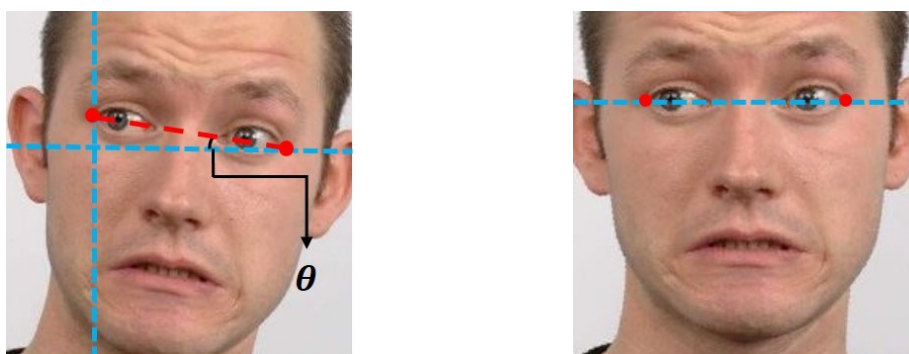


Figure 2-4 Face alignment and warping

opposite direction). The corresponding facial landmarks need to be rotated $-\theta$ with at the same time. Firstly the landmarks $\{(x_1, y_1), \dots, (x_N, y_N)\}$ are shifted such that the origin is at the center of the image, and those points become $\{(x'_1, y'_1), \dots, (x'_N, y'_N)\}$. Then, the new coordinates of facial landmarks should be formulated as:

$$\begin{bmatrix} x''_i \\ y''_i \end{bmatrix} = \begin{bmatrix} \cos(-\theta) & -\sin(-\theta) \\ \sin(-\theta) & \cos(-\theta) \end{bmatrix} \begin{bmatrix} x'_i \\ y'_i \end{bmatrix} \quad (2-2)$$

where (x''_i, y''_i) is the rotated coordinate of the i^{th} facial landmarks.

Face needs to be cropped from the image after alignment. x_{min}/y_{min} and x_{max}/y_{max} indicate the minimum and maximum of the coordinates of facial landmarks in x/y axis, respectively. We crop the image by the rectangle whose top-left and bottom-right points are (x_{min}, y_{min}) and (x_{max}, y_{max}) , respectively. The input size of our model is $224*224*3$ and we resize the face image by bi-cubic interpolation.

2.3 Convolution Neural Network

2.3.1 Introduction to Convolution Neural Network

Convolution neural network(CNN) is a deep learning method that is composed by a series of convolutional layers, pooling layers and fully-connected layers. It is general fact that, with the great capability of CNN, the learned models can obtain effective features in different tasks since the models are trained to be able to get a higher dimension representative feature from data instead of the hand-crafted features. The followings are introduction of some basic components of CNN.

1) Convolution, Pooling and Activation Function

Convolution/correlation is widely used in image processing, such as mean filter, edge detector, unsharp masking, *etc.* Convolution in image processing usually aims at extracting some kinds of attributes or features in an image by some hand-crafted kernels. However, those fixed kernels are not optimal for all tasks and the extracted feature is one-

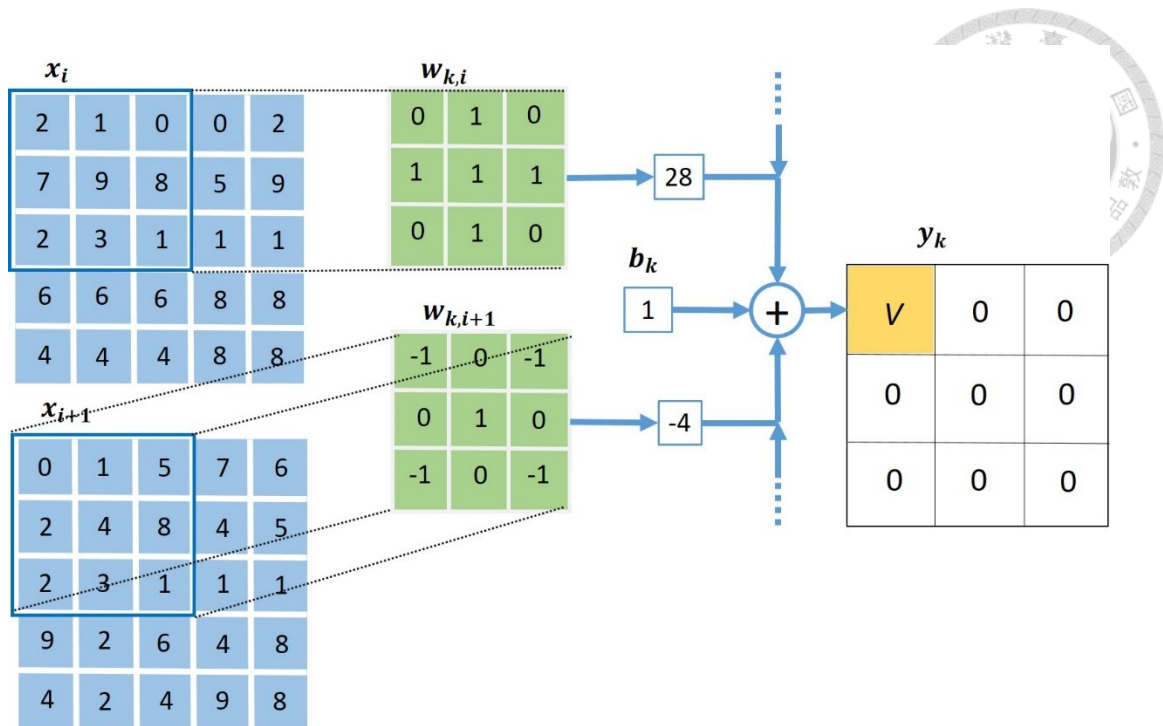


Figure 2-5 Convolution in convolutional layer

sided. Convolution layer essentially is a large set of kernels that learn from the dataset and extract various, efficient features. The process of convolution is shown in Figure 2-5, where x_i stands for the i^{th} channel of the feature map, $w_{k,i}$ and b_k represent the i^{th} kernel and bias of the k^{th} filter, respectively. y_k is the k^{th} channel of the output. The y_k can be expressed as:

$$y_k = \sum_i x_i * w_{k,i} + b_k \quad (2-3)$$

where $*$ denotes convolution.

In CNN construction, we usually stack a series of convolutional layers. Convolution is combination of multiplication and addition, so it is a linear operation. If we stack two convolution layers directly, in another words, if we combine two linear functions, then it actually equals to one linear function. Therefore, nonlinear function is utilized to connect two convolutions and this function is named activation function. Conventional activation functions include sigmoid, Rectified linear unit (ReLU), softmax, *etc.* Activation function

improves the capability of CNN and accelerates the convergence.

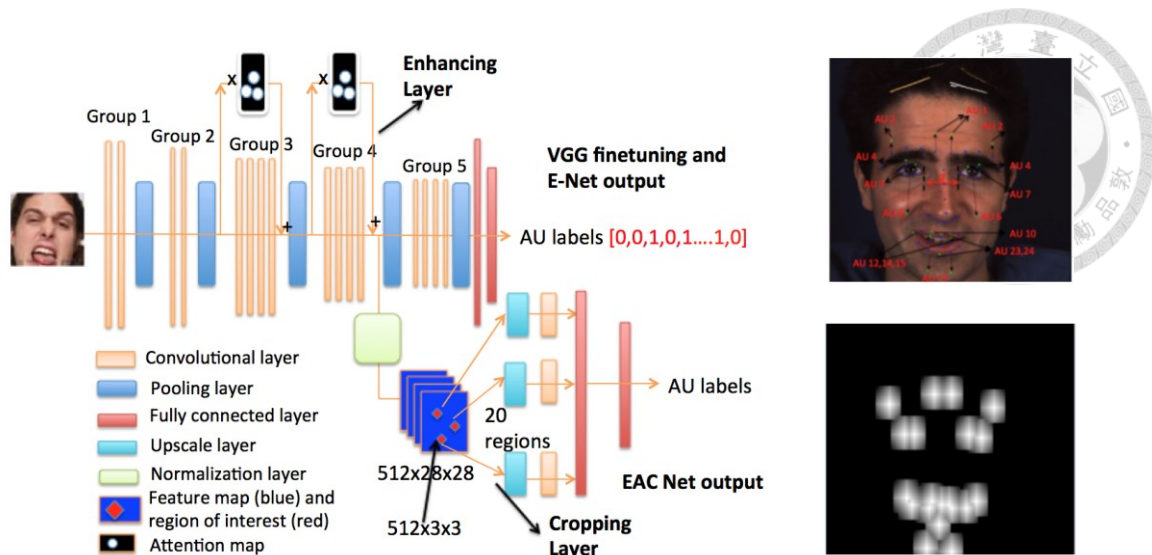
The extracted feature can be the same size to the input with padding, which costs a large amount of computation for the next layer and may contains redundant, unimportant information or noise. Pooling layer are proposed to solve this problem. Max pooling and mean pooling are the most widely used pooling in deep learning. For example, the max-pooling layer picks the largest value in a region, and sets that grid on the kernel to be 1 and others to be 0. Also, the stride is usually designed to be greater than 1, so the purpose of down-sampling can be achieved.

2) *Fully-Connected Layer*

One of the advantages of CNN is that feature extractor and classifier can be optimized at the same time, and the optimization is called end-to-end training. After extracting feature through multi convolution layers, stacked fully-connected layers accomplish classification by those features.

Fully-connected (FC) layers in CNN is essentially neural network. In FC layers, every neuron of an input feature map is connected to every neuron of the output feature map during the process. Therefore, every value of output feature map contains the relationship of all value in the input feature map.

FC layers usually occupy a great numbers of parameters since they connect all the neurons, which possibly cause overfitting. Regularization is adopted in FC layers to constrain the network specify to the training data. Another most commonly used operation to alleviate overfitting is drop-out. At every training epoch, every node is either drop-out off the net with probability $1-p$ or kept with probability p , then a reduced network is left. Incoming and outgoing connections to a drop-out node are also removed. Only the reduced network is trained in that epoch.



(a) Architecture of EAC net

(b) Attention map

Figure 2-6 Enhancing and Cropping network

2.3.2 CNN for Facial Analysis

Given the impressive performance of CNN, researchers started to utilize CNN to solve facial analysis tasks. It is a general belief that CNN is fully qualified to extract effective feature of faces and capture details in faces. Related work of facial expression recognition has been introduced in Section 1.2.3, so not tired in words here.

Face recognition has been extensively studied in recent years because of its practical application. DeepID[2] utilizes multi CNN models to capture appearance feature around the key points on face, and feature downscaled by PCA is verified by Joint Bayesian. VGG-Face[24] adopted the structure of VGG net [25] to solve face identification tasks. Both DeepID and VGG-face outperform the hand-crafted feature-based methods.

HyperFace[26] and all-in-one CNN for facial analysis[27] uses one shared CNN to accomplish multi-tasks for facial analysis. All-in-one CNN initializes their network with CNN model trained for face recognition and then extracts feature from different layers to solve different facial analysis problems, such as age estimation, facial landmark localization, gender estimation and so on.

As mentioned before, one way to recognize facial expression is to detect facial action units. Deep Region Multi-Label (DRML)[7] uniformly divides a face feature map into 8*8 patches and extracts low-level feature from each patch independently, such that attribute of each face patch can be reserved. In order to focus on small changes on the key region of face, Enhance and Cropped Net(EAC)[28] give more attention to key regions through masking a attention map generated by facial landmarks(as shown in Figure 2-6 (b)). In Figure 2-6 (a), similar to DRML, instead of doing batch normalization over the entire feature map, EAC cropped patches around the facial landmarks and do normalization independently to reserve attribute of each region.

Chapter 3 Facial Expression Recognition



In this chapter, we would like to introduce our network Deep Temporal-Contrastive Network (DTCN) that solves facial expression recognition and explain how the method is used to capture temporal information. Brief introduction of preprocessing is given in Section 3.1. In Section 3.2, we introduce the structure of DTCN, which is the combination of two model, Temporal-Contrastive Appearance Network (TCAN) and Temporal-Contrastive Geometry Network (TCGN), which will be described in details in Section 3.3 and Section 3.4, respectively. In Section 3.5, we introduce how to combine TCAN and TCGN.

To be clear, every video/image sequence in facial expression recognition datasets only contains one expression and the data (CK+ and Oulu-CASIA) we used in this paper is started with neutral expression and ended with peak expression. Therefore, we can use the two most representative faces, which are neutral face and peak expression face in recognition datasets, to describe the image sequence.

3.1 Preprocessing

The raw data of facial expression recognition (CK+ and Oulu-CASIA) is a set of image sequences. A frame of the image sequence contains a person and background. We need to capture the face area and do some preprocessing before the input data go through our network.

As presented in Chapter 2, face detection and facial landmark localization have been well developed. In order to focus on facial expression recognition, we decide to adopt some open source to capture faces and localize facial landmarks. The pipeline of preprocessing is shown in Figure 3-1. To speed up data processing, we transform each frame of the image sequence to grayscale (one channel) and downscale each frame for k times ($k=3$ in this thesis) at step 1. At step 2, we capture the face in the frame by the face detector from *Dlib* [29]. According to the face area detected in step 2, facial landmarks are localized by the detector from *Dlib* and the face is aligned as we mentioned in Section

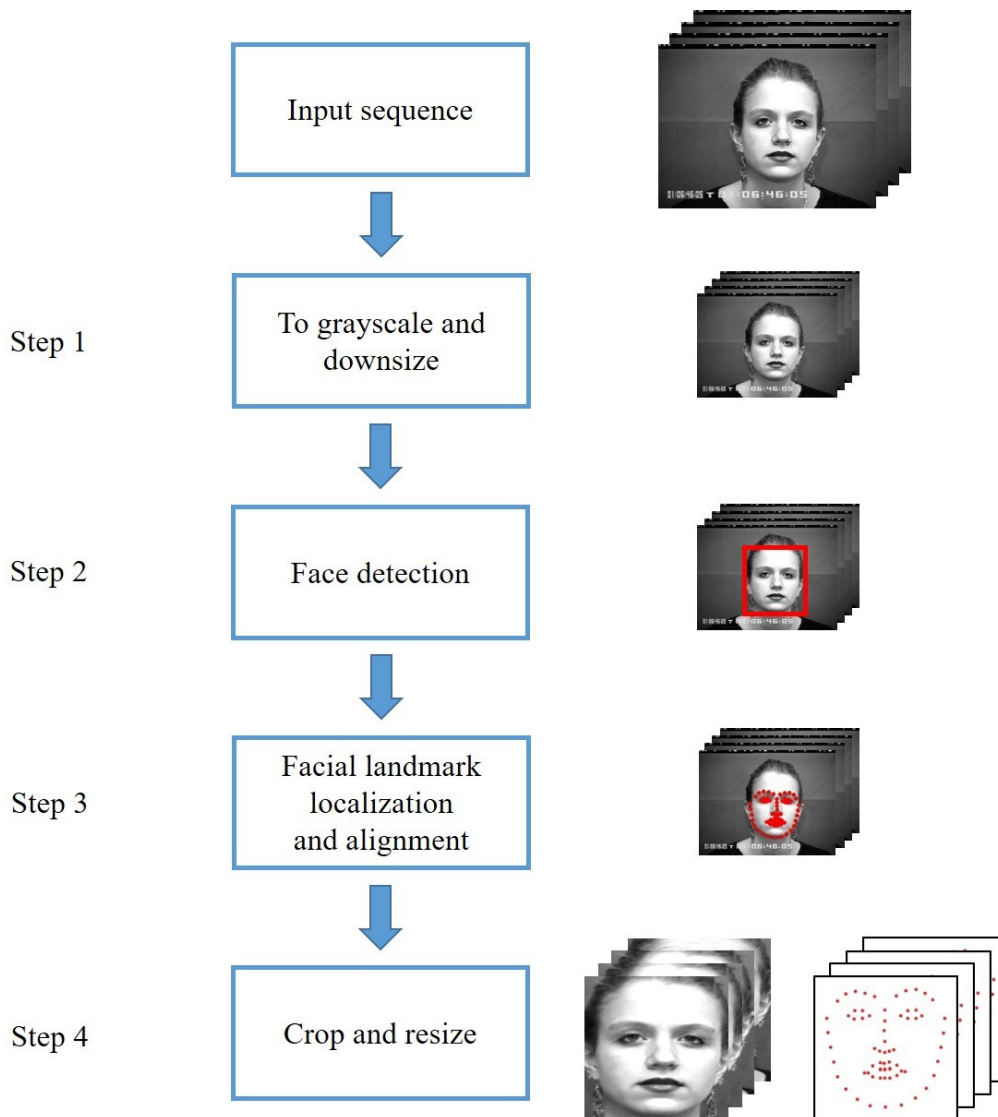
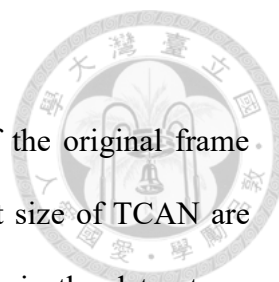


Figure 3-1 Pipeline of preprocessing



2.2.3.

At the last step, we map the coordinate of facial landmark of the original frame (higher resolution) and crop the face from the original frame. Input size of TCAN are $224 \times 224 \times 3$, thus we need to resize the face to 224×224 . If images in the dataset are grayscale, the input of TCAN is stacked by the grayscale image for three times. Due to the input size of TCAN, the coordinates of facial landmarks range from 0 to 224.

3.2 Structure of DTCN

Deep temporal-contrastive network (DTCN) is designed for facial expression recognition. DTCN is a combination of TCAN and TCGN, which share the same concept that captures the changes from temporal feature in feature map and facial landmark domain.

As shown in Figure 3-2, TCAN is bounded by the blue box. The input face sequence, which is cropped from the original video frames, goes through a pre-train face model VGG-Face [24] and generate the corresponding feature vectors. Then, TCAN

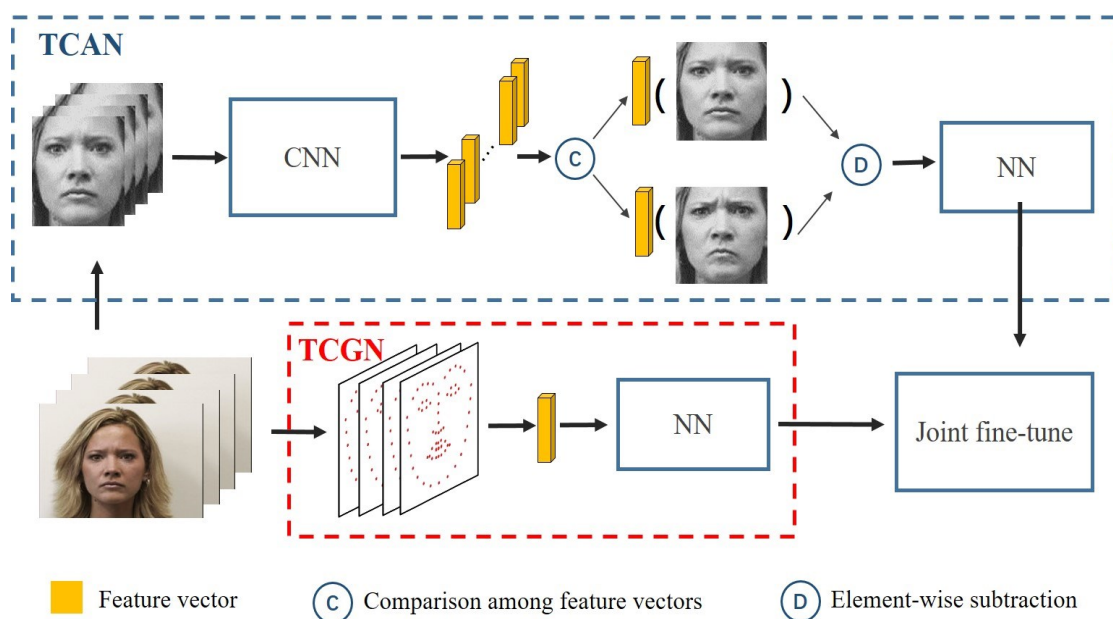
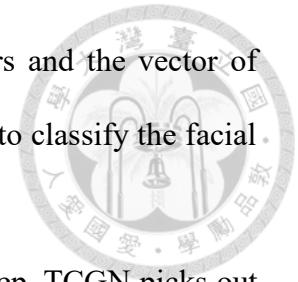


Figure 3-2 Structure of DTCN

automatically picks out the two most representative feature vectors and the vector of element-wise subtraction between those two feature vectors is used to classify the facial expression.



TCGN is bounded by the blue box in Figure 3-2. In the first step, TCGN picks out two groups of facial landmarks (one face/frame has one group of facial landmarks) whose coordinates have the largest Euclidean distance. Similar to TCAN, the vector of element-wise subtraction of those two most representative groups of facial landmarks is utilized to recognize facial expression.

After training TCAN and TCGN separately, we fuse these two models and fine-tune the integrated model (Section 3.5.1). Theory and details of TCAN and TCGN are given in the next two sections.

3.3 Temporal-Contrastive Appearance Network

3.3.1 Transfer Learning for TCAN

Performance of deep learning method depends on the amount and variety of training data in some ways. However, it's time-consuming to train a new CNN model and in many cases, the amount of available data for specific tasks is relatively small for CNN training. Therefore, it is reasonable for researchers to design a more effective and robust way to train the network, which thus calls for transfer learning or knowledge transfer in the field of deep learning and machine learning.

There are several commonly used transfer learning methods. The first basic one is taking a model trained on a large dataset as a fix feature extractor. This strategy is suitable for the tasks whose datasets are very small. One stream of [13] utilize the pre-train model from VGG-Face [24] as a fixed feature extractor and those extracted feature are used to classify facial emotion. However, this method obviously cannot be adopted when the

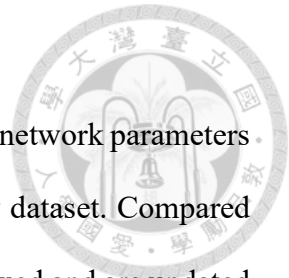
small dataset is very different from the large dataset.

Another strategy, which is called fine-tuning, is to initialize their network parameters by the pre-trained model and to train the initialized model with new dataset. Compared with the first method, the weights of the pre-trained network are not fixed and are updated by continuing the backpropagation training.

The work in [30] proposes partial transfer learning to initialize the network parameters with VGG16 even when the type of input (depth image) differs from VGG16 (RGB image). In order to avoid overfitting and representation specificity, the author “freeze” parameters in the first few layers and simplify the structure of VGG16. The simplified model has a better performance than the original size model in validation, which proves its theory of transfer learning.

Hyper-Face [26] and All-in-one CNN [27] initialize their models’ parameters by models trained for face detection and face identification, respectively, to facilitate subsequent solving of other facial analysis problems. Therefore, we decide to initialize our convolution layers with weights of VGG-Face [24] trained for face recognition.

The architecture of TCAN is shown in Figure 3-3. As transferred from VGG 16, TCAN contains 5 convolution groups and the groups are constructed by 2, 2, 3, 3, 3 convolutional layers, respectively. Max pooling is added between every two convolution groups. Our task is different from the pre-trained model and we reduce the total parameters to prevent overfitting, so parameters of the classification part (neural network) are initialized by following a normal distribution with specified mean and standard deviation instead of being initialized by the pre-trained model.



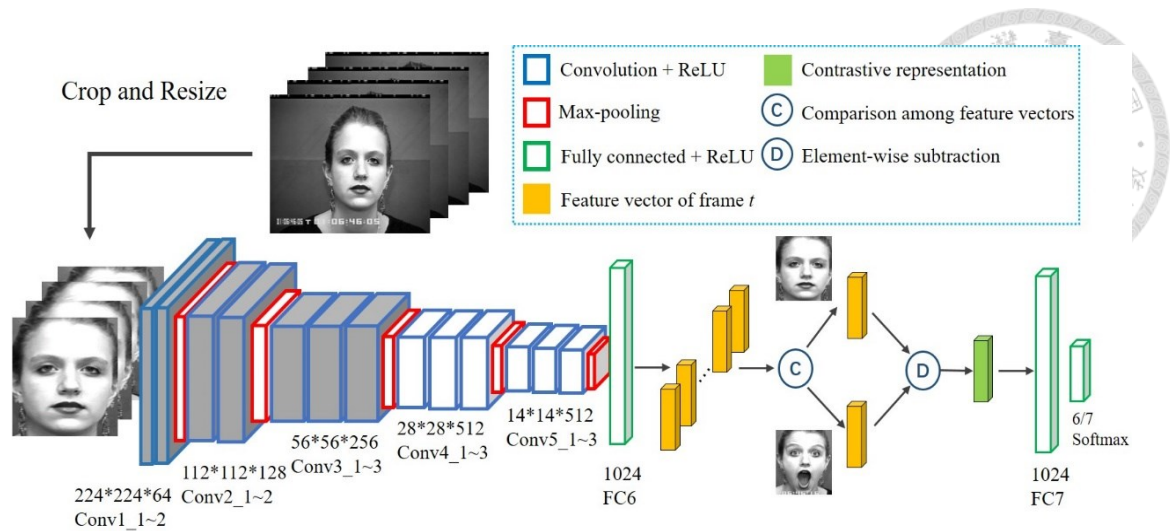


Figure 3-3 Architecture of TCAN

From the transfer learning theory of [30], using pre-trained model and fine-tuning it with small training set may lead to representation specificity, and thus causes overfitting. In our case, both CK+ and Oulu-CASIA are small datasets with hundreds of image sequences, while VGG-Face is trained with 13,000 faces. Inspired by [30], we “freeze” parameters of the first three groups of convolution layers without fine-tuning (as shown in Figure 3-3, the gray area indicates the frozen layers). It is proved in many works that the front convolution layers extract low level feature of input image, such as texture, corner, edge, *etc.* Therefore, the frozen layers can be seen as a fixed feature extractor and can prevent representation specificity.

What’s more, we add batch normalization [31] in convolution and fully-connected layers, which can reduce internal covariate shift in neural networks and allowing users to use higher learning rate for training.

3.3.2 Contrastive Representation of TCAN

The face of each image frame goes through the convolution layers in Figure 3-3 and outputs the corresponding feature vector. After we extract the feature vector of face in

each frame, what does the feature vector contain? From the theory of [32], the extracted face feature x is represented as the sum of two Gaussian variables:

$$x = \mu + \varepsilon \quad (3-1)$$

where μ indicates face identity and ε indicates face variation, e.g., facial expression, lighting.

Inspired by [32], we consider that the extracted feature vector $FV_{i,j,t}$, corresponding to the j^{th} emotion of the i^{th} individual at frame t , is composed of four terms:

$$FV_{i,j,t} = \mu_i + e_{j,t} + L_t + N_t \quad (3-2)$$

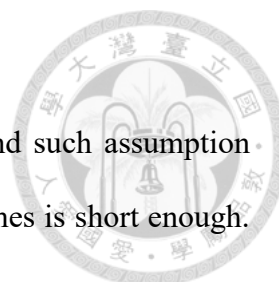
where μ_i represents the face identity information of i^{th} individual, $e_{j,t}$ is the variation of j^{th} emotion at frame t , L_t and N_t stands for lighting and noise at frame t .

Note that the facial expression in the real scenario is a gradual change from neutral expression to certain peak expression. Each video/image sequence in facial expression recognition datasets only contains one emotion, thus the two most representative frames in those videos are the neutral face and peak expression face. So the two most representative frames in videos for facial expression recognition are equivalent to the neutral and peak facial expression frames.

Inspired by [4], influence of μ_i and N can be avoided by using the so-called contrastive representation, which is a vector of the element-wise subtraction between feature vectors of neutral and peak facial expression. Contrastive representation CR can be denoted as follow:

$$\begin{aligned} CR &= d(FV_{i,j,t_{peak}}, FV_{i,j,t_{neutral}}) \\ &= (\mu_i + e_{j,t_{peak}} + L_{t_{peak}} + N_{t_{peak}}) \\ &\quad - (\mu_i + e_{j,t_{neutral}} + L_{t_{neutral}} + N_{t_{neutral}}) \end{aligned} \quad (3-3)$$

where $FV_{i,j,t_{peak}}$ and $FV_{i,j,t_{neutral}}$ represent the feature vectors of peak and neutral



expressions, respectively, and $d(\cdot)$ is element-wise subtraction.

Assume that L_t and N_t between two frames are identical, and such assumption can be more justified if the interval between every two adjacent frames is short enough.

Under this assumption, CR can be written as follows:

$$CR = e_{j,t_{peak}} - e_{j,t_{neutral}} \tag{3-4}$$

The emotion variation of neutral expression $e_{j,t_{neutral}}$ tends to zero, and then the contrastive representation CR actually is mapped from the emotion variation of peak expression $e_{j,t_{peak}}$. If the contrastive representation CR is incorporated as the input of the classifier, then the classification process is free of interference from face identity, lighting and noise.

So, a crucial problem is how to pick out the most representative frames out of a video. [33] proposes a method that captures several key frames of different actions by clustering the extracted features based on the distance between every pair of feature clusters. This idea considers that features of neighboring frames are similar, so in order to pick out the most representative frames, two feature vectors which are separated with the largest Euclidean distance will be chosen by the model. In Session IV, the neutral and peak frames picked by the system are shown and verifies whether this method works or not.

3.3.3 Training Process of TCAN

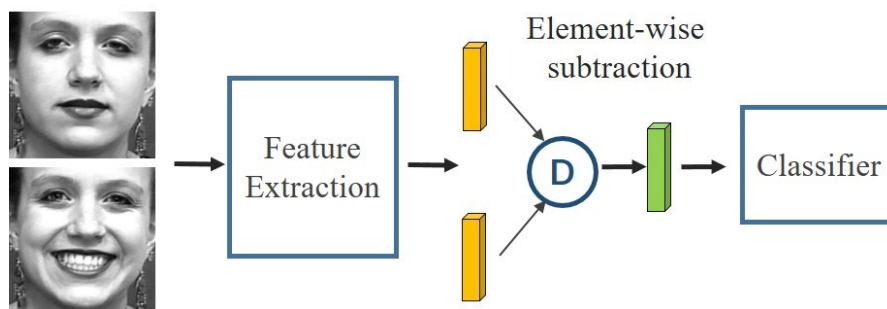


Figure 3-4 End to end training of TCAN

A main disadvantage of video-based methods is that training process is time-consuming even across multiple GPUs. Due to this drawback, it is hard to obtain optimal hyper parameters (such as numbers of layers/filters, thresholds, *etc.*) for FER. So we here propose a novel way to train our network more efficiently and time-saving.

Instead of input the entire image sequences during the training process, we pick out the neutral and pick expression image frames manually and feed it to TCAN (as shown in Figure 3-4). The databases we have used in this paper, namely, CK+ and Oulu-CASIA, are started with neutral face and ended with peak expression face. So the neutral expression is randomly picked from the first two images of an image sequence and the peak expression is randomly picked from the last 20% images.

It is worthwhile to clarify that the most representative frames (the neutral and peak expression in FER) are automatically picked out by the model during testing. The manual process that we designed above only aims at faster training.

Details of parameters in TCAN are shown in Figure 3-3. In the training process, the learning rate is set as 5×10^{-5} and the batch size is 8. In order to prevent overfitting, dropout is adopted in fully-connected layers (not including fc6) and data are augmented. That is, dropout rate is set to be 0.5, and an online data augmentation is implemented where the training data are randomly horizontally flipped, subjected to random rotation from -10° to 10° , and with random brightness and added Gaussian noise.

3.3.4 Loss Function

Loss function is a function that maps an event or values of one or more variables into a real number intuitively representing some “cost” caused by the corresponding event. The most widely used loss function for classification is cross entropy which describes the similarity between two probability distributions p and q . It equals zero when p and q have

the same distribution.

An optimization problem seeks to minimize a loss function and we can design different loss functions to meet specific optimization problems. Contrastive loss was first proposed by [34] which aims at reserving the distance attribute in high dimension space after dimensionality reduction. For example, in facial expression recognition, GCN [4] adopted contrastive loss function whose purpose is to enlarge the distance between two different emotions and diminish the distance if the two emotions are identical.

In our thesis, contrastive loss 1) forces the high level distance between neutral and peak expression frames to be as large as possible; 2) forces the high level distance between neutral (peak) expression frames to be as small as possible. The illustration of contrastive loss in our work is shown in the first row of Table 3-1. Contrastive loss \mathcal{L}_{con} is formulated as:

$$\begin{aligned} \mathcal{L}_{con} = & \max\left(0, m_{con1} - Eu\left(FV_{i,j,t_{peak}}, FV_{i,j,t_{neutral}}\right)\right) \\ & + \max\left(0, Eu\left(FV_{i,j,t_{neutral}}, FV_{i,j,t_{neutral}+1}\right) - m_{con2}\right) \\ & + \max\left(0, Eu\left(FV_{i,j,t_{peak}}, FV_{i,j,t_{peak}-1}\right) - m_{con2}\right) \end{aligned} \quad (3-5)$$

where m_{con1} and m_{con2} are margins, $Eu(\cdot)$ is a function that calculates the Euclidean distance between input vectors. If the distance between neutral and peak expressions is greater than m_{con1} , the first term will not contribute to \mathcal{L}_{con} . Similarly, If the distance between neutral (peak) expressions is smaller than m_{con2} , the second (third) term will not contribute to \mathcal{L}_{con} .

However, contrastive loss is a very strong limitation that it might mislead the learning process. To obtain a weaker constrain, triplet loss is utilized in face recognition [35]. As shown in Figure 3-5, a training sample is randomly chosen and it is called “anchor”. Then a sample of the same class with the anchor and a sample of different class

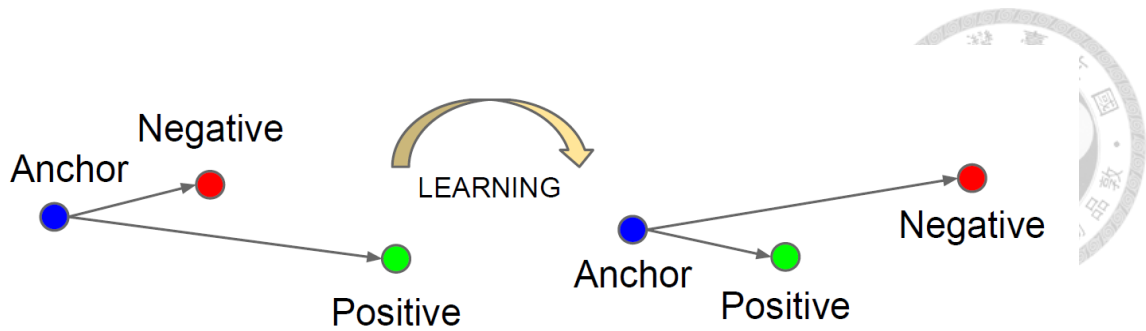


Figure 3-5 Triplet loss



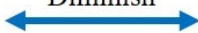

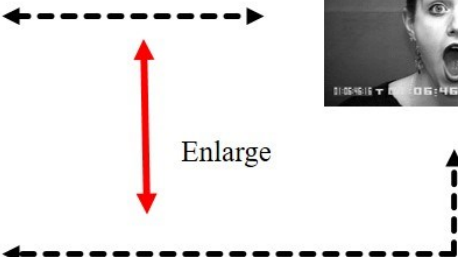

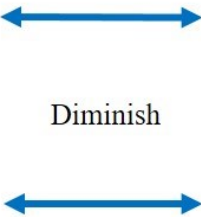
to the anchor are picked and are called “positive” and “negative”, respectively. The triplet loss \mathcal{L}_{tri} can be presented as following:

$$\mathcal{L}_{tri} = \max(0, Eu(FV_{pos}, FV_{anchor}) - Eu(FV_{neg}, FV_{anchor}) + \alpha) \quad (3-6)$$

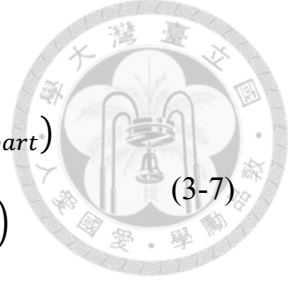
where FV_{anchor} , FV_{pos} and FV_{neg} are the feature of the anchor, the positive sample and negative sample, Eu is Euclidean distance between two vectors and α is a threshold. In the above formula, we can observe that \mathcal{L}_{tri} will be zero when the distance between $Eu(FV_{pos}, FV_{anchor})$ and $Eu(FV_{neg}, FV_{anchor})$ is greater than α .

Different from contrastive loss, triplet loss does not need to explore the exact margin (which might change in various situation) between two samples of same/different classes. As shown in Figure 3-5, triplet loss only aims to make the distance between different classes greater than the one between the same class.

Table 3-1 Three extra loss functions applied in facial expression recognition

Loss function	Diagram	
<p>Contrastive Loss</p>		<p>Enlarge</p>  <p>Diminish</p> 
<p>Triplet Loss</p>		<p>Enlarge</p> 
<p>Partial Contrastive Loss</p>		<p>Diminish</p> 

Though the intensity difference between neutral and peak expressions are quite significant, feature vectors of them should not be large since they both come from the same face. Center loss[36] defines and updates a center for every class and force each sample close to the center of its class. Inspired by center loss, we remove the first term of contrastive loss and define a new loss function called partial contrastive loss. Partial



contrastive loss is designed as follows:

$$\begin{aligned} \mathcal{L}_{partialCon} = & \max(0, Eu(FV_{i,j,t_{neutral}}, FV_{i,j,t_{neutral}+1}) - m_{part}) \\ & + \max(0, Eu(FV_{i,j,t_{peak}}, FV_{i,j,t_{peak}-1}) - m_{part}) \end{aligned} \quad (3-7)$$

where m_{part} is a margin decided by trial and error.

In the training process, four frames are picked out: neutral expression, peak expression and the adjacent frames of them ($t_{neutral} + 1$ and $t_{peak} - 1$). When the Euclidean distance between adjacent frames are less than m_{cen} , \mathcal{L}_{center} does not contribute to the total loss. The total loss function is defined as follows:

$$\mathcal{L}_{total} = \mathcal{L}_{crossEntropy} + \lambda \cdot \mathcal{L}_{con}(\text{or } \mathcal{L}_{tri}, \mathcal{L}_{partialCon}) \quad (3-8)$$

where λ is weight of \mathcal{L}_{con} (or $\mathcal{L}_{tri}, \mathcal{L}_{partialCon}$), and is set as 0.1 in our work. In Chapter 4, performance of loss function using \mathcal{L}_{con} , \mathcal{L}_{tri} and $\mathcal{L}_{partialCon}$ are compared.

3.4 Temporal-Contrastive Geometry Network

3.4.1 Architecture of TCGN

As mentioned in Section 2.2.2, the coordinates of facial landmark contains geometry information of facial expression. In TCAN, we can describe and classify the expression by appearance. If we observe the expression by geometry feature of the face, those two models with different kinds of input are complementary. Therefore, we propose a deep neural networks (DNN) which takes coordinates of facial landmarks as input and we call it temporal-contrastive geometry network (TCGN).

The architecture of TCGN is shown in Figure 3-6. Firstly, facial landmarks are located by open sources for face alignment since the face alignment is quite developed. Then, the normalization of coordinates of facial landmarks are implemented. After normalization, similar to TCAN, we pick out the two most representative groups of facial

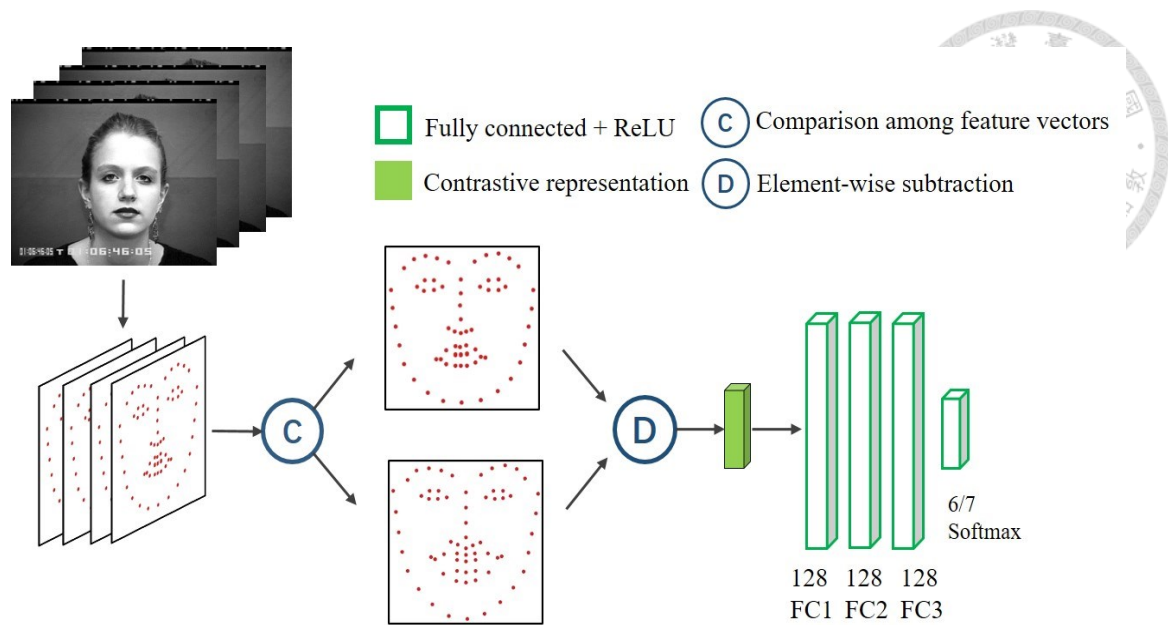


Figure 3-6 Architecture of TCGN

landmarks which have the largest Euclidean among the input groups of facial landmarks. The so-called contrastive representation is the element-wise subtraction between those two groups of facial landmarks picked by TCGN and it becomes the input of the next neural network classifier.

The architecture of TCGN is constructed by four fully-connected layers whose number of neuron are 128, 128, 128, 6/7 (depends on the number of emotion classes in the dataset). The first three fully-connected layers are followed by drop-out layers with drop-out rate which equals to 0.5. Batch normalization is applied to every fully-connected layer as well.

The architecture of TCGN is relatively small compared to TCAN and it cost less time in training and testing. In Section 3.4.2, we explain the reason why the neural network is named “temporal-contrastive” and discuss the commonality and difference between TCAN and TCGN.

3.4.2 Contrastive Representation of TCGN

After face alignment, we obtain the coordinates of facial landmarks of each frame.

Before the classification, the input coordinates of facial landmark should be normalized. As we mentioned before, the number of facial landmarks in this thesis are 68 which includes the points on jaw. Points on the jaw only describe the shape of face and contain a little information of facial expression. The shape of face is more likely to interfere the classification than benefit the result. So in TCGN, we remove those points whose serial numbers are from 1 to 17. Then the coordinates of facial landmark we used to classify expression can be denoted as $\{(x_{18}^{t_0}, y_{18}^{t_0}), (x_{19}^{t_0}, y_{19}^{t_0}) \dots (x_{68}^{t_0+T}, y_{68}^{t_0+T})\}$, where $(x_i^{t_0+j}, y_i^{t_0+j})$ represent the coordinate of the i^{th} facial landmark at time $t_0 + j$ (t_0 is the start time). To normalize the facial landmarks, coordinate of every point is subtracted by the coordinate of nose:

$$\begin{bmatrix} X_i^{t_0+j} \\ Y_i^{t_0+j} \end{bmatrix} = \begin{bmatrix} x_i^{t_0+j} \\ y_i^{t_0+j} \end{bmatrix} - \begin{bmatrix} x_{nose}^{t_0+j} \\ y_{nose}^{t_0+j} \end{bmatrix} \quad (3-9)$$

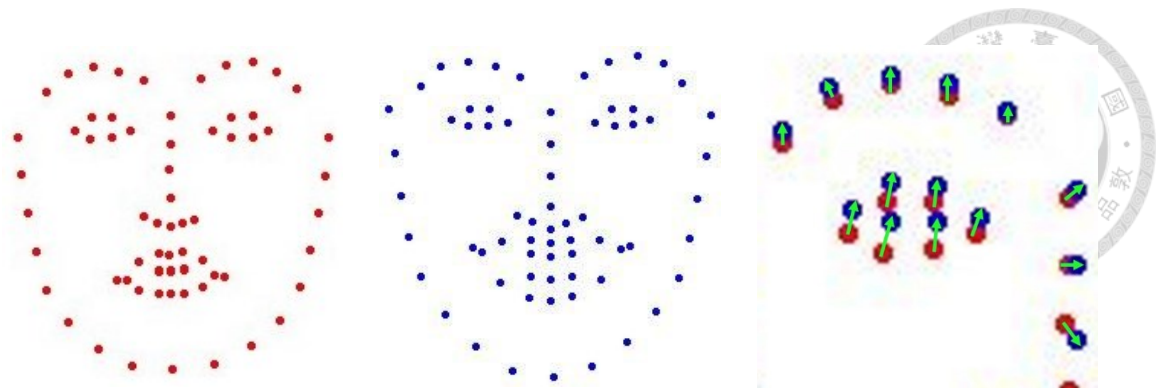
where $(X_i^{t_0+j}, Y_i^{t_0+j})$ represents the normalized coordinate of the i^{th} facial landmark at time $t_0 + j$. $(x_i^{t_0+j}, y_i^{t_0+j})$ are the coordinate of nose at time $t_0 + j$.

The Euclidean distance between two groups of facial landmark indicate the extent of changes between two faces. We picked out two groups of facial landmarks which have the largest Euclidean distance, which means we pick out the two most representative faces among the input video. Similar to Equation (3-3) The contrastive representation of TCGN can be formulated as follow:

$$CR = \{X_i^{t_{peak}} - X_i^{t_{neutral}}, Y_i^{t_{peak}} - Y_i^{t_{neutral}}\}, i = 18, 19, \dots, 68 \quad (3-10)$$

where $(X_i^{t_{peak}}, Y_i^{t_{peak}})$ and $(X_i^{t_{neutral}}, Y_i^{t_{neutral}})$ represent the i^{th} landmark's coordinate of peak and neutral expressions, respectively. Thus, CR is a vector with 102 dimensions (51 facial landmarks with x and y coordinates).

Figure 3-7 (a) and (b) are the shape of neutral face and happy face, and the



(a) Shape of neutral face (b) Shape of happy face (c) Motion around the eye

Figure 3-7 Visualization of contrastive representation

contrastive representation can be seen as the motion of every facial landmark. Figure 3-7 (c) illustrates the trajectory from neutral face (red points) to happy face (blue points) and the green arrows represent the motion vectors. As we can observe in Figure 3-7 (c), the eyebrow stretches out, the eye tends to squint and in Figure 3-7 (a)(b) we can see that there are great changes around lips. All of these observations are indicating that the person may be smiling. The information like we mentioned above is recorded in the contrastive representation. Therefore, it is quite convincing that the contrastive representation of TCGN should have good performance in facial expression recognition.

In 3.3.2, we introduce the contrastive representation for TCAN whose purpose is to remove the personality effect as much as possible and reserve the expression feature. However, in TCGN, the input is facial landmarks which does not contain any face identity information. The contrastive representation of TCGN does not aim to remove the personality but to record the trajectory of facial landmarks. Essentially, TCGN is a network that utilizes the motion of facial landmarks to classify facial expression.

3.4.3 Training Process of TCGN

Different from TCAN, TCGN does not need to go through multi convolutional layers and obtain the two representative frames by comparing. The speed of comparing distance

between facial landmarks is very fast, so we can directly pick out the peak and neutral frame.

In order to alleviate overfitting, drop-out layer is added after every fully-connected layer and the drop-out rate is 0.5. What's more, we augment the data by horizontally flipped and Gaussian noise addition.

3.5 Deep Temporal-Contrastive Network

3.5.1 DTCN: Combination of TCAN and TCGN

There are several methods to fuse multiple models. In machine learning, we usually adopt the method so-called “ensemble” to merge models. Ensemble is a voting method that basically can be divided as soft voting and hard voting. In soft voting, the result is decided by the maximum of the sums of the predicted probabilities, which is recommended for an ensemble of well-calibrated classifiers. In hard voting, the final result is decided by the majority predictions of all models.

Weighted-sum is one of the most widely used methods and it essentially belongs to soft voting. The final classification result of multiple models fused by weighted-sum can be express by the following:

$$Prediction = argmax(\sum_i \lambda_i \cdot Prob_i) \quad (3-11)$$

where $Prob_i$ and λ_i are the predicted probabilities and the weight of the i^{th} model. All of the methods we mentioned above are just combine the prediction of each model and do not tune the parameters inside each model.

Though weighted-sum is simple and practical, it still has some drawbacks. The weight λ_i of the i^{th} model is decided by trial and error, which means it may not obtains the optimal solution. Another issue is that the type of every model maybe different, *e.g.*, inputs of TCAN and TCGN are face image and facial landmarks, and action recognition

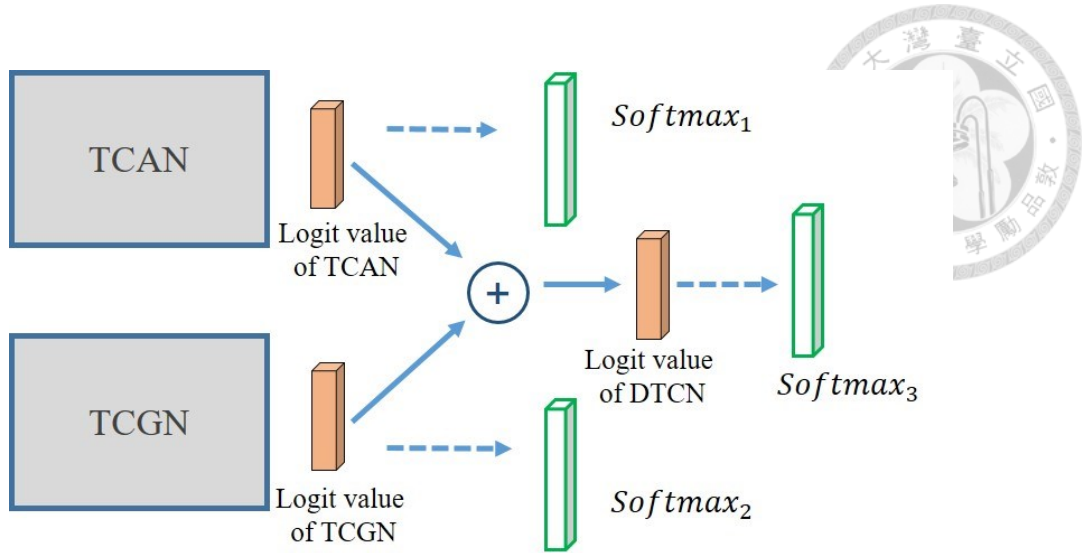


Figure 3-8 Joint fine-tuning of DTCN

models that take optical flow and video as input. Merging those models directly by weighted-sum may not appropriate.

Inspired by [3], we combine TCAN and TCGN by joint fine-tuning. We “freeze” the parameters of TCAN and TCGN except the last layer and fine-tune the last layer of TCAN and TCGN simultaneously. As shown in Figure 3-8, the layers filled with gray (including the CNN) indicate that the parameters of those layers are fixed.

After training TCAN and TCGN separately, the output of DTCN is defined as follow:

$$y_{DTCN,i} = \sigma(\text{logits}_{TCAN,i} + \text{logits}_{TCGN,i}) \quad (3-12)$$

where $\text{logits}_{TCAN,i}$ and $\text{logits}_{TCGN,i}$ are the i^{th} logit values of TCAN and TCGN, respectively. $\sigma(\cdot)$ represents softmax activation function.

The losses of TCAN, TCGN and DTCN are denoted as L_{TCAN} , L_{TCGN} and L_{DTCN} , respectively. All of those losses are cross-entropy:

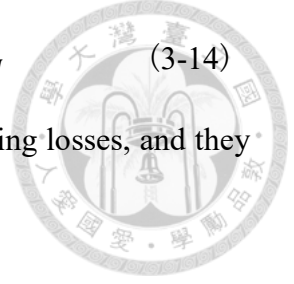
$$L = - \sum_{i=0}^c \hat{y}_i \log(y_i) \quad (3-13)$$

where \hat{y}_i is the i^{th} output value after softmax, and y_i is the i^{th} value of ground truth.

Then, the total loss for the joint fine-tuning process can be defined as:

$$L_{total} = \lambda_{TCAN} \cdot L_{TCAN} + \lambda_{TCGN} \cdot L_{TCGN} + \lambda_{DTCN} \cdot L_{DTCN} \quad (3-14)$$

where λ_{TCAN} , λ_{TCGN} and λ_{DTCN} are the weights of the corresponding losses, and they are set to 1, 1 and 0.1 in our thesis.



3.5.2 Attributes of DTCN

In Section 1.2.3, we introduce some image-based and video-based methods for facial expression recognition and both of them have some drawbacks. DTCN is proposed to overcoming those drawbacks and take advantage of them as much as possible. The comparison between image-based methods, video-based methods and DTCN is shown in Table 3-2.

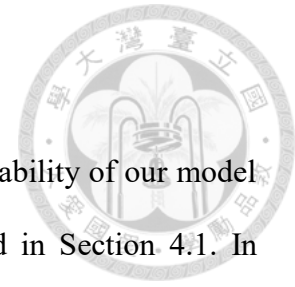
Thanks to the concise reserve information, computation of our method become much smaller in training and testing. In training process, we manually pick out the most representative frames so that not all the frames in the input video need to go through the network and do back propagation which are takes most of the time during training.

In testing, traditional video-based methods need to process 18 frames in CK+ and 22 frames in Oulu-CASIA in average to extract temporal feature. But our method only need 2 frames to capture expression changes, so we save computation in temporal feature extraction.

Table 3-2 Comparison of three kinds of method for facial expression recognition

Method	Temporal information	Redundant information	Computation Cost	Stability
Image-based	No	No	Low	No
Video-based	Yes	Yes	High	Yes
DTCN	Yes	No	Medium	Yes

Chapter 4 Experiment



In this chapter, we conduct some experiments to verify the capability of our model DTCN. Configuration of our hardware and software is introduced in Section 4.1. In Section 4.2, we describe the two most widely used facial expression recognition datasets, CK+ [37] and Oulu-CASIA [12], and how to evaluate our model. In Section 4.3, we compare our DTCN to the state-of-art methods and analysis our experiment results. In Section 4.4, we visualize the most representative frame chosen by DTCN and verify the correctness of our method.

4.1 Configuration

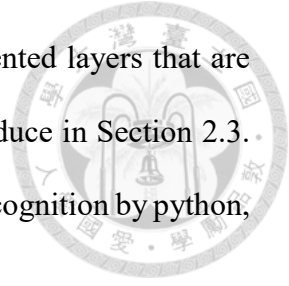
The experiments of our facial expression recognition system is executed on the computer consisted of an Intel(R) Xeon(R) central processing unit (CPU), and a NVIDIA GeForce GTX 1080 graphic processing unit (GPU) on Ubuntu. Table 4-1 demonstrates some details of the hardware specification of this computer.

We employ TensorFlow [38] library as our deep-learning toolbox for construction of our Deep Temporal-Contrastive Network (DTCN). TensorFlow provides plenty of functions for building deep-learning networks and frequently updates functions of the

Table 4-1 The Specification of the computer for our experiments for FER.

Equipment	Specification
Central Processing Unit (CPU)	Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz
Graphic Processing Unit (GPU)	NVIDIA GeForce GTX 1080
Operating System (OS)	Ubuntu 16.04 LTS
System Bit Type	64 bit

newest theories published in conferences. It contains some implemented layers that are common and basic for most of the network components as we introduce in Section 2.3. We implement face detection, face alignment and facial expression recognition by python, thus we install *Dlib*, *OpenCV*, TensorFlow of python version.



4.2 Description of Dataset and Evaluation

4.2.1 The Extended Cohn-Kanade (CK+)

Cohn-Kanade (CK+) [37] is the most widely used database for facial expression recognition. There are 593 image sequences (123 individuals) in CK+ and 327 of them (118 individuals) have seven emotion labels: angry, contemptuous, disgust, fear, happiness, sadness, and surprise. Some researchers regard the image sequences without labels as neutral expression, and in this thesis we only use 327 image sequences of seven labels (without neutral expression). CK+ contains grayscale images and RGB images, thus we convert RGB image sequences into grayscale. The number of each emotion is listed in Table 4-2. We can observe that the image sequences of seven emotions are unevenly distributed, *e.g.*, “Happiness” has 69 image sequences while “Contemptuous” has only 18, which might cause overfitting to the emotion which has greater number and bad performance on the emotion which has tiny number.

The size of images in CK+ is 640*490 for grayscale and 640*480 for RGB and the

Table 4-2 The distribution of seven emotions in CK+

Expression	Angry	Contemptuous	Disgusted	Fearful
Numbers	45	18	59	25
Expression	Happiness	Sad	Surprised	---
Numbers	69	28	83	---



Figure 4-1 Faces of one image sequence in FER datasets (CK+)

face occupies more than one third of the image. What's more, the coordinates of 68 facial landmarks are given in CK+. Each image sequence of CK+ is started with neutral expression and ended with peak expression, as we shown in Figure 4-1. The length of sequences in CK+ ranges from 6 to 71.

4.2.2 Oulu-CASIA

Oulu-CASIA [12] consists of 80 individuals and each individual have 6 image sequences that are labeled as angry, disgust, fear, happiness, sadness and surprise, respectively, with two imaging system, visible light(VL) and near infrared(NI), under three illumination conditions: weak illumination (only computer monitor is on), normal

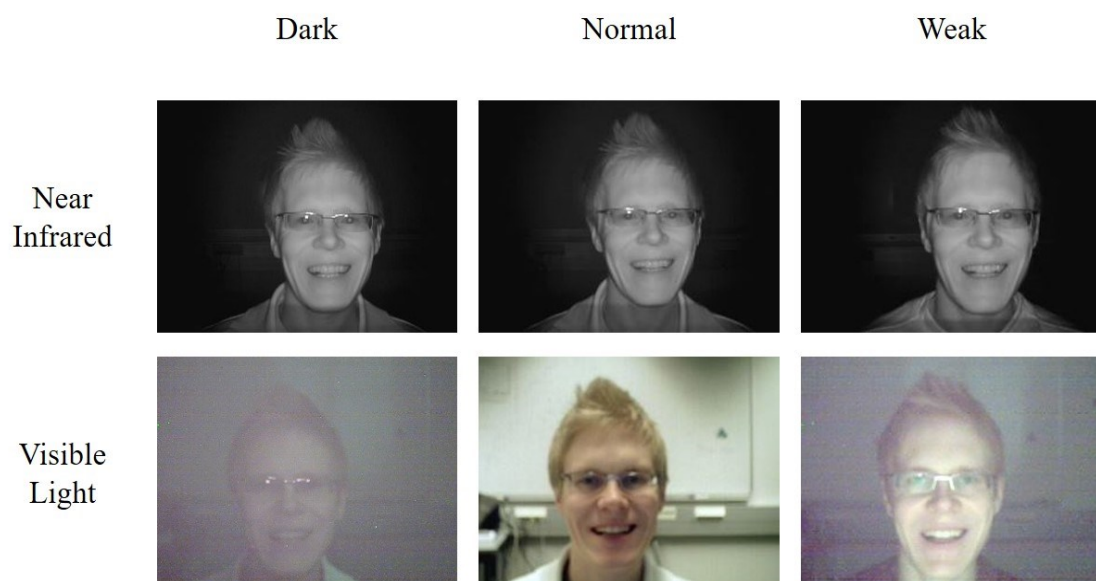


Figure 4-2 Images in NI and VL systems under three illumination conditions

Table 4-3 The distribution of six emotions in Oulu-CASIA

Expression	Angry	Disgusted	Fearful
Numbers	80	80	80
Expression	Happiness	Sad	Surprised
Numbers	80	80	80

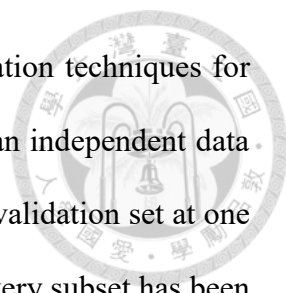
illumination and dark illumination (all lights are off). Figure 4-2 shows the six kinds of data included in the Oulu-CASIA. As we can observe, images generated by near infrared are not disturbed by different illumination conditions while images generated by visible light are seriously interfered by poor illumination.

According to the-state-of-the-art [3-5], these methods are all using the data under normal illumination in VL system. Therefore, to verify the capacity of our system fairly, we use the same data as the-state-of-the-art methods. Table 4-3 shows the emotion distribution of Oulu-CASIA. Different to CK+, the image sequences are evenly divided to six emotions.

The size of image in Oulu-CASIA is 320*240 and the face occupies more than one third of the image., coordinates of facial landmarks are not given in Oulu-CASIA thus we utilize *Dlib* to localize 68 facial landmarks. Compare to CK+, images in Oulu-CASIA has lower resolution, which might be one of the reasons why the average performance of CK+ is better than Oulu-CASIA. Similar to CK+, the sequence in this database is started with neutral expression and ended with peak expression. The length of sequences in Oulu-CASIA ranges from 9 to 45.

4.2.3 Evaluation

Appropriate and rigorous evaluation is significant for researchers to verify the



system/model they develop. K-fold cross validation is model validation techniques for assessing how the results of a statistical analysis will generalize to an independent data set. It evenly divides the data into K subset, and takes one subset as validation set at one time and other K-1 subsets are used for training. After K iteration, every subset has been validation set and we can obtain K results. Then the final score of the model is the average of those K results.

In facial expression recognition, we hope that the designed system can recognize expressions for anybody who is even not in the training data. Thus, we should test our system with the expression of individuals who are not in the training data.

Individual independent 10-fold cross validation is the most common used protocol to verify the capability of a facial expression recognition system. Take CK+ as an example, 10 subsets are composed by sampling in ID of ascending order with ten intervals (the last two subsets only have 11 individual). As shown in Figure 4-3, Subset 2~10 with blue bounding boxes in Iteration 1 are invited to train the model and Subset 1 with red bounding box in Iteration 1 becomes the validation. Because of the particularity of CK+, the number of image sequence in every subset is not the same (Subset 2 has 40 sequence

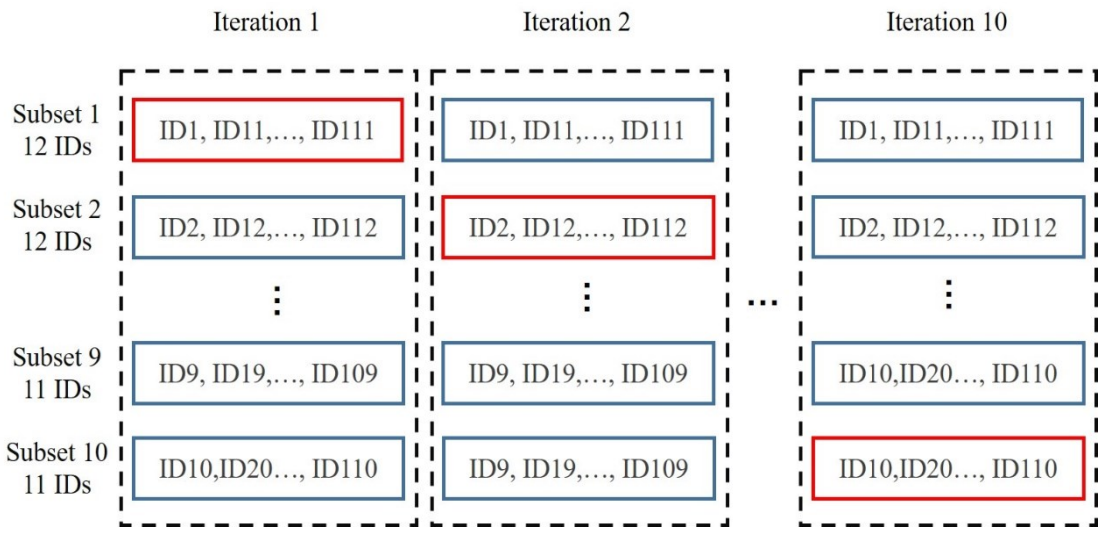


Figure 4-3 Individual independent 10 fold-cross validation (CK+)

while Subset 9 only has 28). After this procedure has run for ten times, the average accuracy will be seen as the performance of DTCN on CK+.



4.3 Quantity Results

DTCN is video-based (sequence-based) method. In testing, T frames are randomly picked out from the input image sequence (if the sequence is shorter than T frames, we pad it with the middle frame) and sorted as ascending order. Then, the selected frames go through DTCN, and outputs the emotion class.

Some image-based methods manually pick out the most expressive frame (last frame in CK+ and Oulu-CASIA) and use it for testing. It is not fair that those testing manually gives up the redundant frames and picks out the most valuable frames and whereas the image-based method should be evaluated with random frames from video. In the real scenario, the peak expression cannot be obtained by these methods.

4.3.1 Results on CK+

a) Comparison with different loss functions

We train our TCAN with different losses while TCGN with just using cross-entropy. Thus, in this part, we will only discuss the performance of TCAN with different loss functions.

The baseline of TCAN is using cross-entropy without others extra loss functions. Then we train TCAN by cross-entropy with triplet loss, contrastive loss and partial contrastive loss, respectively. The performance of each loss is shown in Table 4-4. From the table we can know that TCAN extra loss functions are better than the baseline in CK+ dataset. By comparing the performance between contrastive loss and partial contrastive loss, it indirectly proves that partial contrastive loss (proposed by us) with lighter limitation is better than contrastive loss.

Table 4-4 Accuracy of our TCAN with different loss in the CK+ database

Method	Accuracy
TCAN (baseline)	95.41%
TCAN + triplet	96.03%
TCAN + contrastive	96.03%
TCAN + partial contrastive	97.25%

b) Comparison with TCAN/TCGN/DTCN

In this part, the accuracy of TCAN, TCGN and their combination, DTCN, will be compared in CK+ dataset.

Figure 4-4 shows the comparison in CK+ among TCAN, TCGN and DTCN. The performance of TCAN is better than TCGN in recognizing most kinds of expression (except contemptuous). The reason why TCAN is better than TCGN is that the input of

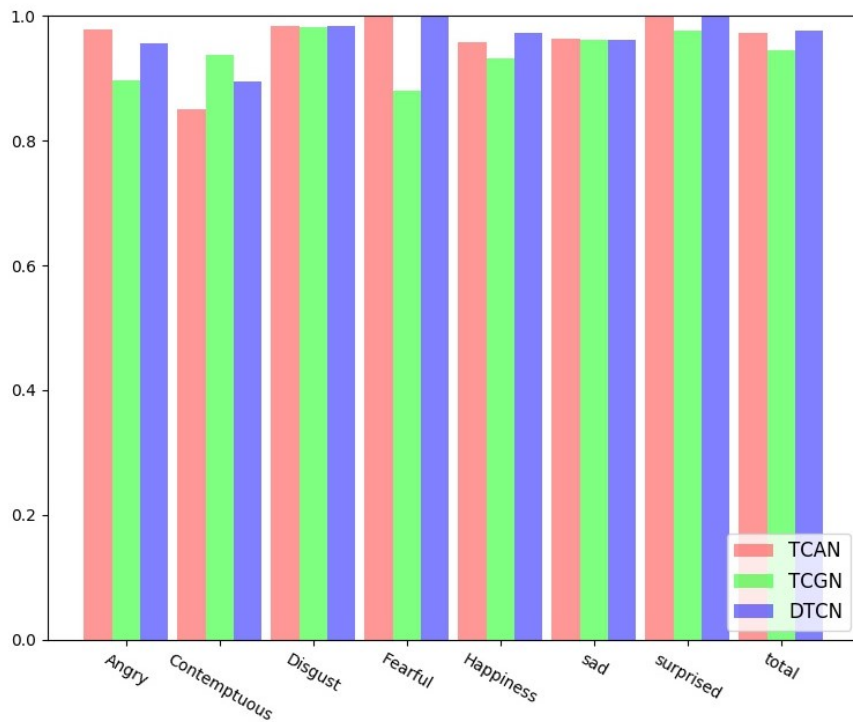


Figure 4-4 The comparison of our TCAN/TCGN/DTCN in CK+

Table 4-5 The accuracy of each expression in CK+ by our models

Model	Angry	Conpt	Disgust	Fearful	Happy	Sad	Surprised	Total
TCAN	97.73%	85.00%	98.33%	100.0%	95.83%	96.30%	100.0%	97.25%
TCGN	89.58%	93.75%	98.21%	88.00%	93.15%	96.15%	97.59%	94.29%
DTCN (weight)	93.62%	84.21%	98.31%	100.0%	94.44%	96.15%	98.80%	96.02%
DTCN (fine)	95.65%	89.47%	98.33%	100.0%	97.18%	96.15%	100%	97.55%

* “conpt”: short for contemptuous

* “weight” indicates that DTCN is combined by weighted-sum

* “fine” indicates that DTCN is combined by joint fine-tune.

TCAN (image sequence) contains more information than the input of TCGN (coordinates of facial landmarks). Relatively, the size of TCAN model is much larger than the size of TCGN and the computation cost of TCAN is much greater than TCGN.

Combination by weighted-sum may be too rough for different kinds of models. As shown in Table 4-5, the performance of DTCN with weighted-sum is even lower than the performance of TCAN. The performance of DTCN with joint-fine-tune is the best among those structures. The accuracy of DTCN is not outperform the TCAN, which we think TCAN is too good that the upside for DTCN is quite limited.

To summarize, if we want to apply our facial expression recognition in real application, we can choose different models according to different configurations. The system which has sufficient computational capability can use TCAN (or DTCN) and achieve high accuracy. The system which has small computing power, *e.g.*, mobile devices and system without GPU, can use TCGN with acceptable accuracy and achieve real-time processing. So it is a tradeoff between computation and time.

Table 4-6 Overall accuracy in the CK+ database

Method	Image/Video-based	Accuracy
HOG 3D [9]	Video-based	91.44%
3D Inception-ResNet [16]	Video-based	93.21%
STM-ExpLet [22]	Video-based	94.19%
3DCNN-DAP [39]	Video-based	92.40%
DTGN [3]	Video-based	91.04%
DTAN [3]	Video-based	91.44%
DTAGN [3]	Video-based	97.25%
PPDN(combined) [5]	Image-based	95.33%
VGG-Face(baseline)	Image-based	93.88%
TCGN (ours)	Video-based	94.29%
TCAN + partial contrastive (ours)	Video-based	97.25%
DTCN + partial contrastive (ours)	Video-based	97.55%

*PPDN(combined): input frame is randomly chosen from 7th to the last frame

c) Comparison with state-of-the-art methods

Table 4-6 is the comparison result on CK+ between the state-of-the-art methods and our work. DTAGN is well performed due to the joint fine-tuning method to combine two different kinds of models; if it just utilizes image as input (Deep Temporal Appearance Net), the accuracy of DTAN can only achieve about 91.44% which is non-competitive than our method.

Image-based method GCN [7] has a great performance in CK+ and Oulu-CASIA, but it does not explain clearly whether it used the peak frame as input for testing, so we do not list the results of GCN as comparison.

VGG-Face is the baseline method built and trained from scratch that inputs with one frame of each image sequence. PPDN [6] achieved 99.3% accuracy on CK+ but it is based

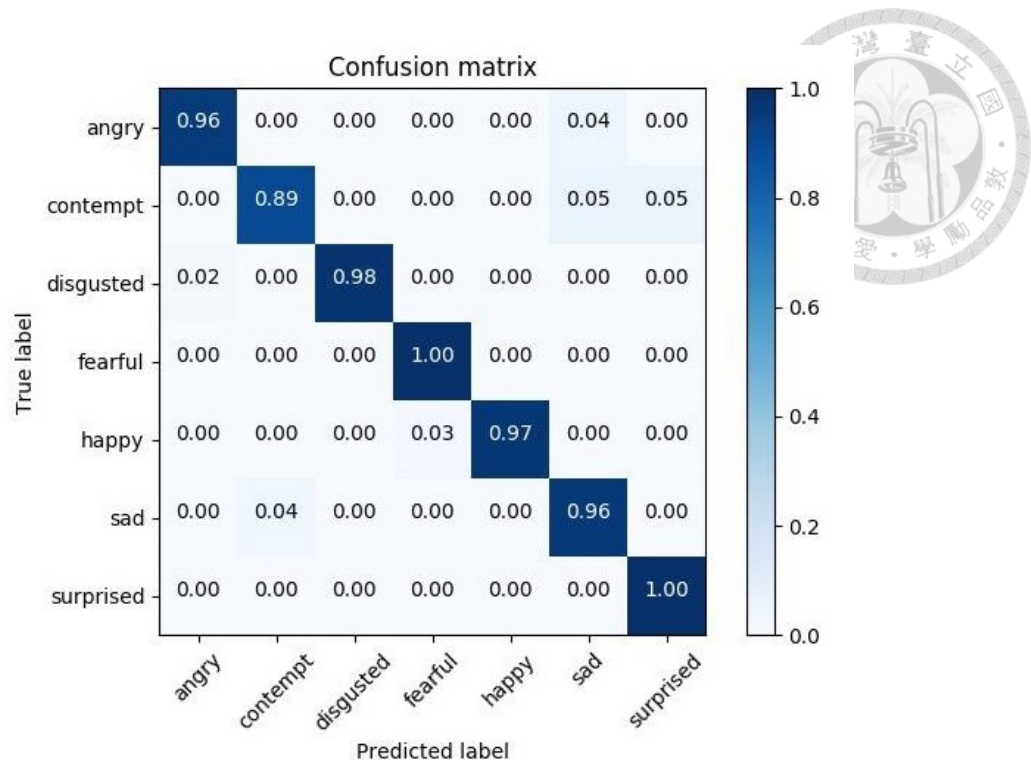


Figure 4-5 Confusion matrix of our DTCN in CK+

on manually pre-defined peak frame as input. For fair comparison, the peak frame should be determined automatically from 7th to the last frames. Based on this setting, the accuracy of PPDN is 95.33%. PPDN drives feature the non-peak expression toward peak expression that may cause low accuracy in non-peak expression because non-peak expression of different emotions may be similar.

Confusion matrix of DTCN (partial contrastive) is shown in Figure 4-5. From the figure, we can observe that DTCN achieves very high accuracy except for the contemptuous, which is not clearly distinguished from sad and surprised. But it still outperforms the state-of-the-art methods.

4.3.2 Results on Oulu-CASIA

a) Comparison with different losses function

Similar with the protocol in Section 4.3.1 (a), we only shows the results of TCAN with different loss functions in this part.

Table 4-7 Accuracy of our TCAN with different loss in the Oulu-CASIA database

Method	Accuracy
TCAN (baseline)	82.88%
TCAN-triplet	82.92%
TCAN-contrastive	82.71%
TCAN-partial contrastive	83.75%

The performance of TCAN with different losses is listed in Table 4-7. As we can observe, extra function is not always better than the baseline. As mentioned in Section 3.3.4, contrastive loss is a very strong limitation that may mislead the training process and the result of baseline is even better than the one with contrastive loss in Oulu-CASIA. And the TCAN with partial contrastive loss proposed by us still achieves great performance.

b) Comparison with TCAN/TCGN/DTCN

In this part, the accuracy of TCAN, TCGN and their combination, DTCN, will be

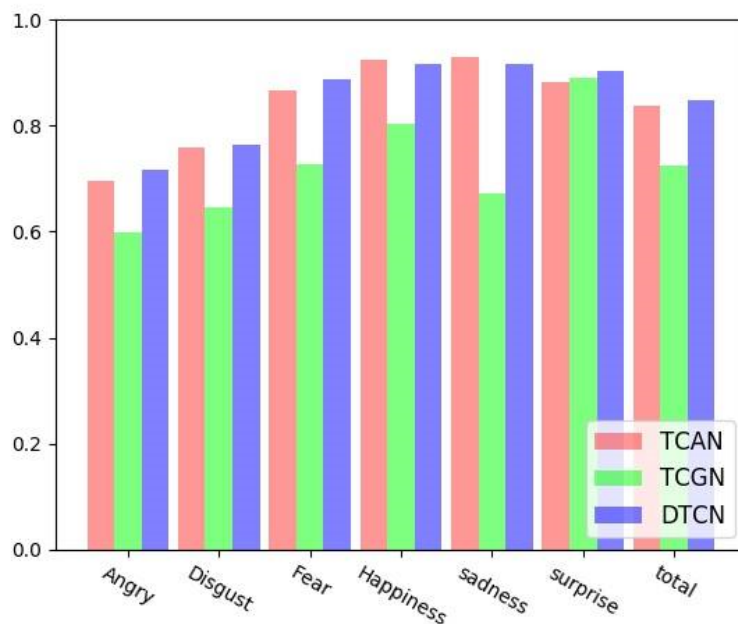


Figure 4-6 The comparison of our TCAN/TCGN/DTCN in Oulu-CASIA

Table 4-8 The accuracy of each expression in Oulu-CASIA by our models

Model	Angry	Disgust	Fearful	Happy	Sad	Surprised	Total
TCAN	69.57%	75.95%	86.67%	92.41%	92.86%	88.24%	83.75%
TCGN	59.76%	64.56%	72.73%	80.46%	67.12%	89.02%	72.50%
DTCN (weight)	67.02%	78.38%	87.67%	92.59%	87.67%	89.41%	83.33%
DTCN (fine)	71.59%	76.54%	88.89%	91.67%	91.55%	90.46%	84.79%

* “conpt”: short for contemptuous

* “weight” indicates that DTCN is combined by weighted-sum

* “fine” indicates that DTCN is combined by joint fine-tune.

compared in Oulu-CASIA dataset.

Figure 4-6 shows the comparison in Oulu-CASIA and in Table 4-8 shows the exact accuracy of each expressions among TCAN, TCGN and DTCN. The accuracy of TCGN is not as good as we expected. The most possible reason is that the facial landmark localization does not perform well in low resolution dataset like Oulu-CASIA. But TCAN still works in Oulu-CASIA. The accuracy of the fused model DTCN has been improved to 84.79% by joint fine-tune while DTCN with weighted-sum acts no better than TCAN, which proves that weighted-sum is not appropriate for the combination among different kinds of models again.

c) Comparison with the state-of-the-art methods

The comparison between our work and state-of-the-art methods are presented in Table 4-9. DTCN achieves the best accuracy, 84.79%, among all those methods and TCAN is still the best if the input data is only image sequence.

Although the accuracy of TCGN is not satisfying compare to the DTGN [3] which

Table 4-9 Overall accuracy in the Oulu-CASIA database

Method	Image/Video-based	Accuracy
AdaLBP [4]	Video-based	73.54%
HOG 3D [1]	Video-based	70.63%
STM-ExpLet [20]	Video-based	74.59%
DTAN [5]	Video-based	74.38%
DTAGN [5]	Video-based	81.46%
PPDN (combined) [6]	Image-based	74.99%
VGG-Face (baseline)	Image-based	80.63%
TCGN (ours)	Video-based	72.50%
TCAN+partial contrastive (ours)	Video-based	83.75%
DTCN+partial contrastive (ours)	Video-based	84.79%

*PPDN (combined): input frame is randomly chosen from 7th to the last frame

inputs facial landmark coordinates of all the video frames, TCGN still assists DTCN in recognizing facial expressions with high accuracy than TCAN. TCAN is better than DTAN [3] with the same input. PPDN [5] achieves 84.59% accuracy if it only takes the last frame of each sequence. But as we mentioned before, it's not fair to just pick the peak expression for expression recognition. The accuracy of PPDN whose input frame is randomly chosen from 7th to the last frame only achieves 74.99%. We can conclude that image-based method is not stable as video-based video.

Figure 4-7 shows the confusion matrix of DTCN in Oulu-CASIA. DTCN works well in fearful, happy, sad and surprised. However, negative expressions like angry, sad and disgusted are hard to be distinguished. As shown in the confusion matrix, angry is sometimes recognized as sad or disgusted.

Overall, in public datasets like CK+ and Oulu-CASIA, our DTCN method not only can outperform the other state-of-the-art approaches, but also only use two frames'

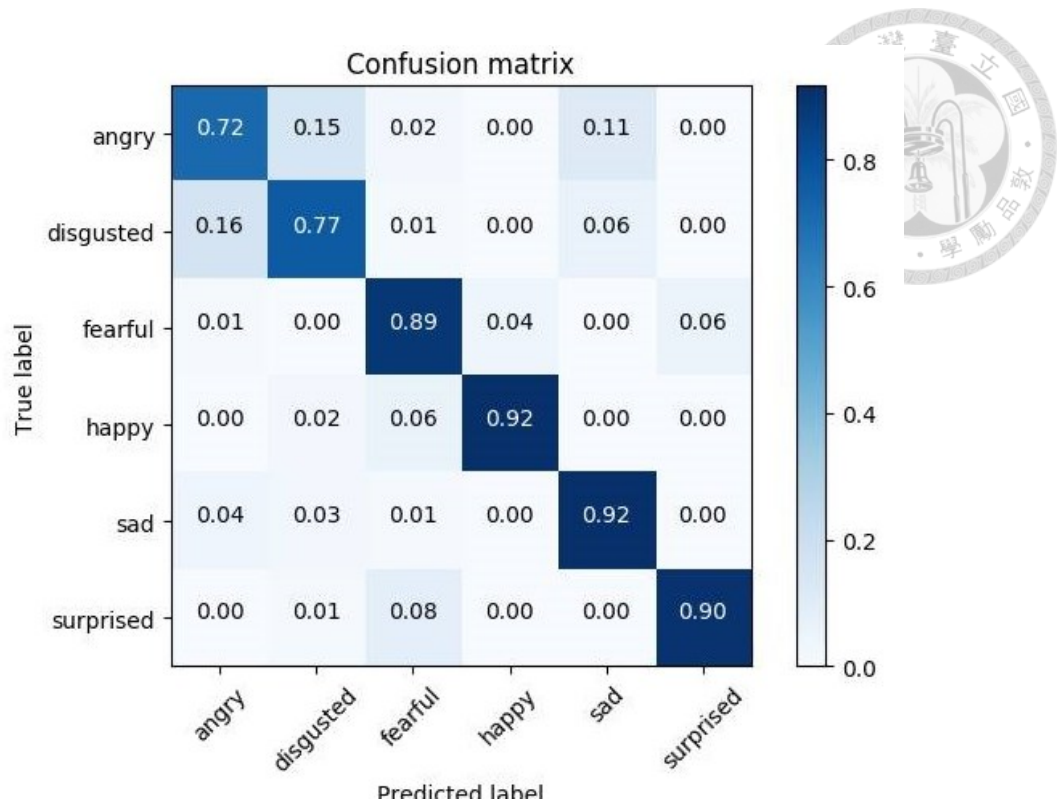


Figure 4-7 Confusion matrix of our DTCN in Oulu-CASIA

information in the final stage of FER classification.





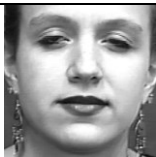

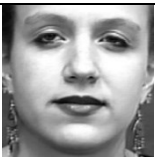

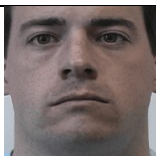
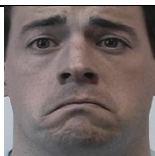

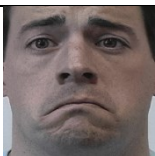
4.4 Quality Analysis

As Session III presents, feature vectors of neutral and peak expression frames can be picked by comparing Euclidean distance. In this session, we visualize the neutral and peak expression frames TCAN selected and present the corresponding prediction.

As shown in the right part of Table 4-10 and Table 4-11, those are the frames which have the largest Euclidean distance in feature vector level and is picked by TCAN. From the tables, we can know that TCAN is capable to choose the neutral and peak expression which proves the correctness of our method.

By reviewing the temporal location expression been picked out, we noticed that most of the neutral expression picked from the 1st or 2nd frame of image sequence, and the peak expression is usually picked from the last frame or near the end of image sequence. As

Table 4-10 The key frames picked by TCAN and its predictions in CK+

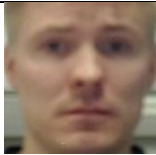

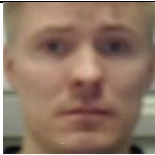
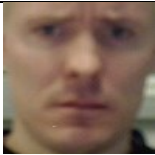
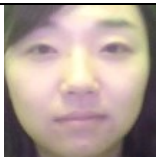
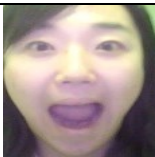
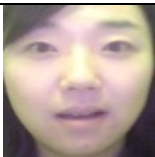
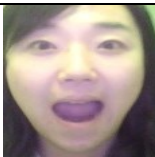
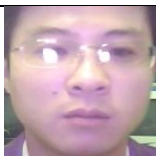
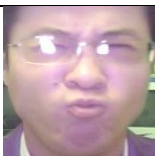
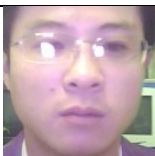
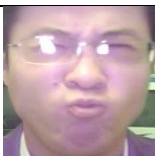
Ground-truth			TCAN		
Peak	Neutral	Label	Peak	Neutral	Prediction
 Frame 1	 Frame 12	Surprised	 Frame 2	 Frame 12	Surprised
 Frame 1	 Frame 45	Happy	 Frame 2	 Frame 29	Happy
 Frame 1	 Frame 42	Sad	 Frame 17	 Frame 43	Sad

we mention is the session of datasets, the sequences in databases are started with neutral expression and ended with peak expression. This result is matching with the description of those datasets. It can confirm that our TCAN can extract most important frames which is neutral and peak expression for the following FER task.

In the third case of CK+ and the second case of Oulu-CASIA, the “neutral” expression actually is expressing slight sad/surprised and this expression is not picked out from the first three frames. Even so, TCAN still can successfully recognize this case due to intensity of the “neutral” and peak expression are different and TCAN can capture the intensity change.

In the third case of Oulu-CASIA, TCAN wrongly recognizes the disgusted expression. From the figure we can observe, the expression of the individual is quite confusing and I think it’s reasonable to classify this expression as angry.

Table 4-11 The key frames picked by TCAN and its predictions in Oulu-CASIA

Ground-truth			TCAN		
Peak	Neutral	Label	Peak	Neutral	Prediction
 Frame 1	 Frame 30	Sad	 Frame 1	 Frame 25	Sad
 Frame 1	 Frame 25	Surprised	 Frame 5	 Frame 13	Surprised
 Frame 1	 Frame 17	Disgusted	 Frame 2	 Frame 17	Angry

Chapter 5 Conclusion and Future Work



In this thesis, we have proposed a novel method for facial expression recognition by combining the advantage of image-based and video-based methods. In order to capture significant temporal feature without incorporating redundant information, neutral and peak expression frames are picked out in feature vector level and contrastive representation of them is used to classify emotion.

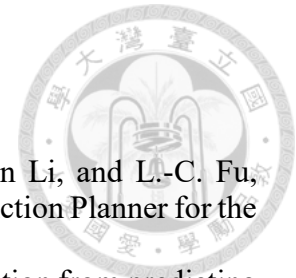
We proposed DTCN in this thesis work, which composed by TCAN and TCGN, where DTCN recognizes emotions by extracting spatial and geometry feature of facial expression. Our DTCN has achieved 97.55% on CK+ and 84.79% on Oulu-CASIA, which outperforms the state-of-the-art methods. By visualization, we can confirm that our approach can select the neutral and peak frame correctly, which can reduce the computational loading of sequence-based facial expression recognition and is without a strong assumption of temporal location of peak expressions.

Our method is more efficient and time-economical compared with other video-based methods, but each frame still needs to go through feature extractor (CNN) which costs large computation.

In future work, we plan to design a method that takes less computation to pick out the neutral and peak expression frames. Then, as a result only two frames need to go through CNN and such facial expression recognition system can be run in real-time.

What's more, we plan to transfer DTCN for facial expression detection and apply it to real scenario.

REFERENCE



- [1] Edwinn Gamborino, Vicente Queiroz, Zih-Yun Chiu, Zi-Jun Li, and L.-C. Fu, "Interactive Reinforcement Learning based Assistive Robot Action Planner for the Emotional Support of Children," *Under Review*, 2018.
- [2] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1891-1898.
- [3] H. Jung, S. Lee, J. Yim, S. Park, and J. Kim, "Joint fine-tuning in deep neural networks for facial expression recognition," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2983-2991: IEEE.
- [4] Y. Kim, B. Yoo, Y. Kwak, C. Choi, and J. Kim, "Deep generative-contrastive networks for facial expression recognition," *arXiv preprint arXiv:1703.07140*, 2017.
- [5] X. Zhao *et al.*, "Peak-piloted deep network for facial expression recognition," in *European Conference on Computer Vision (ECCV)*, 2016, pp. 425-442: Springer.
- [6] S. Ren, X. Cao, Y. Wei, and J. Sun, "Face alignment at 3000 fps via regressing local binary features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1685-1692.
- [7] K. Zhao, W.-S. Chu, and H. Zhang, "Deep region and multi-label learning for facial action unit detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3391-3399.
- [8] P. Ekman and W. V. Friesen, *Facial Action Coding System: Investigator's Guide*. Consulting Psychologists Press, 1978.
- [9] A. Klaser, M. Marszałek, and C. Schmid, "A spatio-temporal descriptor based on 3d-gradients," in *British Machine Vision Conference (BMVC)*, 2008, pp. 275: 1-10: British Machine Vision Association.
- [10] P. Scovanner, S. Ali, and M. Shah, "A 3-dimensional sift descriptor and its application to action recognition," in *Proceedings of the 15th ACM international conference on Multimedia*, 2007, pp. 357-360: ACM.
- [11] G. Zhao and M. Pietikainen, "Dynamic texture recognition using local binary patterns with an application to facial expressions," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 6, pp. 915-928, 2007.
- [12] G. Zhao, X. Huang, M. Taini, S. Z. Li, and M. Pietikäinen, "Facial expression recognition from near-infrared videos," *Image and Vision Computing*, vol. 29, no. 9, pp. 607-619, 2011.
- [13] Y. Fan, X. Lu, D. Li, and Y. Liu, "Video-based emotion recognition using CNN-RNN and C3D hybrid networks," in *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, 2016, pp. 445-450: ACM.
- [14] B. Hasani and M. H. Mahoor, "Spatio-temporal facial expression recognition using convolutional neural networks and conditional random fields," in *Automatic Face & Gesture Recognition (FG 2017), 2017 12th IEEE International Conference on*, 2017, pp. 790-795: IEEE.
- [15] J. Zhou, X. Hong, F. Su, and G. Zhao, "Recurrent convolutional neural network regression for continuous pain intensity estimation in video," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 84-92.

- 
- [16] B. Hasani and M. H. Mahoor, "Facial expression recognition using enhanced deep 3D convolutional neural networks," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 2278-2288: IEEE.
- [17] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137-154, 2004.
- [18] H. Li, Z. Lin, X. Shen, J. Brandt, and G. Hua, "A convolutional neural network cascade for face detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5325-5334.
- [19] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Z. Li, "S³FD: Single Shot Scale-invariant Face Detector," *arXiv preprint arXiv:1708.05237*, 2017.
- [20] A. Zadeh, Y. C. Lim, T. Baltrusaitis, and L.-P. Morency, "Convolutional Experts Constrained Local Model for 3D Facial Landmark Detection," in *Proceedings of the IEEE International Conference on Computer Vision Workshop*, 2017, pp. 2519-2528.
- [21] D. Merget, M. Rock, and G. Rigoll, "Robust Facial Landmark Detection via a Fully-Convolutional Local-Global Context Network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 781-790.
- [22] M. Liu, S. Shan, R. Wang, and X. Chen, "Learning expressionlets on spatio-temporal manifold for dynamic facial expression recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1749-1756: IEEE.
- [23] A. Kacem, M. Daoudi, B. B. Amor, and J. C. Alvarez-Paiva, "A novel space-time representation on the positive semidefinite cone for facial expression recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3180-3189.
- [24] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition," in *British Machine Vision Conference (BMVC)*, 2015, vol. 1, no. 3, p. 6.
- [25] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.
- [26] R. Ranjan, V. M. Patel, and R. Chellappa, "Hyperface: A deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.
- [27] R. Ranjan, S. Sankaranarayanan, C. D. Castillo, and R. Chellappa, "An all-in-one convolutional neural network for face analysis," in *Proceedings of 12th IEEE International Conference on Automatic Face & Gesture Recognition*, 2017, pp. 17-24: IEEE.
- [28] W. Li, F. Abtahi, Z. Zhu, and L. Yin, "Eac-net: A region-based deep enhancing and cropping approach for facial action unit detection," in *Proceedings of 12th IEEE International Conference on Automatic Face & Gesture Recognition*, 2017, pp. 103-110: IEEE.
- [29] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, no. Jul, pp. 1755-1758, 2009.
- [30] A.-S. Liu, Z.-J. Li, T.-H. Yeh, Y.-H. Yang, and L.-C. Fu, "Partially transferred convolution neural network with cross-layer inheriting for posture recognition from top-view depth camera," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 4139-4143: IEEE.
- [31] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network

- training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.
- [32] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun, "Bayesian face revisited: A joint formulation," in *European Conference on Computer Vision*, 2012, pp. 566-579: Springer.
- [33] T.-W. Hsu, Y.-H. Yang, T.-H. Yeh, A.-S. Liu, L.-C. Fu, and Y.-C. Zeng, "Privacy free indoor action detection system using top-view depth camera based on key-poses," in *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2016, pp. 004058-004063: IEEE.
- [34] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proceedings of IEEE computer society conference on Computer Vision and Pattern Recognition*, 2006, vol. 2, pp. 1735-1742: IEEE.
- [35] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815-823.
- [36] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *European Conference on Computer Vision*, 2016, pp. 499-515: Springer.
- [37] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2010, pp. 94-101: IEEE.
- [38] M. Abadi *et al.*, "TensorFlow: A System for Large-Scale Machine Learning," in *USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, 2016, vol. 16, pp. 265-283.
- [39] M. Liu, S. Li, S. Shan, R. Wang, and X. Chen, "Deeply learning deformable facial action parts model for dynamic expression analysis," in *Asian conference on computer vision*, 2014, pp. 143-157: Springer.