

# **Emotion recognition in video using Deep learning method with subtract pre-processing**

A DISSERTATION SUBMITTED TO THE CENTRE OF SIGNAL AND IMAGE PROCESSING, DEPARTMENT OF ELECTRONIC AND ELECTRICAL ENGINEERING AND THE COMMITTEE FOR POSTGRADUATE STUDIES OF THE UNIVERSITY OF STRATHCLYDE IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF PHILOSOPHY

By

Zhihao He

-2019

## **Declaration**

I declare that this thesis embodies my research work and that it is composed by myself. Where appropriate, I have to make acknowledgments to the work of others.

The copyright of this thesis belongs to the author under the terms of the United Kingdom Copyright Acts as qualified by the University of Strathclyde Regulation 3.50. The due acknowledgment must always be made of the use of and contained in, or derive from, this thesis.

Signed:

Date:

## Acknowledgement

First and foremost, I would like to express my deepest gratitude to my supervisors, Prof John J. Soraghan and Dr. Gaetano Di Caterina, for their guidance, support, and encouragement throughout my MPhil. They guide me in deep learning and image processing area, which made me feel brave when I have trouble learning. Besides, several reference chance they give me help me so much. Without their helpful advice and practical support to numerous personal concerns, I could not finish the work in this thesis.

I want to thank my family and friends and my girl friend FengLin , for their patient and encouragement to me. When I have a problem with work or life, they always made me feel bright and brave again. In the hard day of study, I can not be without them.

I also thank my University. The University of Strathclyde gives me a chance to study. Also, it is my pleasure to work and study with my colleagues in the Electronic & electrical engineering Centre for Signal & Image Processing (CeSIP) at the University of Strathclyde. Thanks for your help and support.

I dedicate this thesis to my parents, my friends, and my supervisors.

---

## List of Abbreviations

ANN:	Artificial neural network
CK:	Cohn-Kanade
CK+:	Cohn-Kanade plus
CVPR:	Computer Vision and Pattern Recognition
DCT:	Discrete cosine transform
HOG:	Histograms of oriented gradients
LBP:	Local binary patterns
RGB:	Red-green-blue
RNN:	Recurrent neural network
ROI:	Region of interest
SVM:	Support vector machines
VJ:	Viola and Jones algorithm
DPNM:	Deformable Part Mode UDPM
RAVDESS:	The Ryerson Audio-Visual Database of Emotional Speech and Song
AP:	Average Precision
mAP:	mean Average Precision
KL:	Kullback_Leibler
MSE:	Mean Square Error
NMS:	Non-Maxing Suppression
TSN:	Temporal Segment Network
SPP:	Spatial Pyramid Pooling
VGG:	Visual Geometry Group
RF:	Random Forests
ILSVRC:	Image Net Large-Scale Visual Recognition Challenge
FPS:	Frame Per Second
GPU:	Graph Processing Unit
YOLO:	You Only Look Once
3D:	three Dimension
LSTM:	Long-Short Term Memory
GRU:	Gated Recurrent Unit
SSD:	Single Shot MultiBox Detector
MTCNN:	Multi-task Cascaded Convolutional Neural Networks
SIFT:	Scale-invariant feature transform
COCO:	Common Objects in COntext

---

## List of Figures

Figure 1 . Color plate.....	15
Figure 2 . Color step .....	16
Figure 3 . Video frames.....	16
Figure 4 . Convolutional neural network neuron transfer data.....	17
Figure 5 . The simple neural network structure.....	18
Figure 6 . 2D convolutional process.....	19
Figure 7 . Full mode convolution.....	20
Figure 8 . Same mode convolution.....	20
Figure 9 . Valid mode convolution.....	21
Figure 10 . Max pooling and Average pooling .....	22
Figure 11 . application of Dropout layer.....	23
Figure 12 . Sigmoid function.....	24
Figure 13 . Tanh function.....	24
Figure 14 . Relu function.....	25
Figure 15 . Activation table. ....	26
Figure 16 . Traditional RNN structure.....	27
Figure 17 . Traditional LSTM structure.....	28
Figure 18 . GRU Unit structure.....	29
Figure 19 . BI-LSTM structure in sequences understanding.....	30
Figure 20 . Alex-net structure.....	31
Figure 21 . Google-net structure.....	32
Figure 22 . Google-net inception structure.....	32
Figure 23 . Deeper layer has a weaker behavior when network has over 20 layers	33
Figure 24 . Res-net Unit.....	33
Figure 25 . Different Res-net structure settings .....	34
Figure 26 . Traditional object detection method.....	35
Figure 27 . SPP net structure.....	39
Figure 28 . Fast R-CNN structure.....	40
Figure 29 . The structure of Faster - RCNN and RPN.....	41
Figure 30 . YOLO9000 setting and structure.....	43
Figure 31 . A comparison between two single shot detection Models: SSD and YOLO .....	44
Figure 32 . Four methods to extract temporal feature from video.....	47
Figure 33 . Spatial and temporal stream Conv-Net.....	47
Figure 34 . Conv LSTM structure.....	49
Figure 35 . 3D CNN structure 1.....	50
Figure 36 . 3D CNN hardwired layer.....	51
Figure 37 . Lip-net structure.....	51
Figure 38 . Haar-like feature model.....	54
Figure 39 . Calculation diagram.....	55
Figure 40 . MTCNN Flowchart.....	56
Figure 41 . P-net structure.....	57

Figure 42 . R-net structure.....	57
Figure 43 . O-net structure.....	57
Figure 44 . One-hot label.....	58
Figure 45 . (a) learning ratio is too small (b) learning ratio is too large.....	64
Figure 46 . Different learning behaviors.....	65
Figure 47 . Parameter of different model.....	66
Figure 48 . Difference of Traditional machine learning and transfer learning ....	67
Figure 49 . Transfer learning structure.....	67
Figure 50 . Different features extracted by convolution layers.....	68
Figure 51 . MNIST sample.....	69
Figure 52 . ILSVRC sample.....	70
Figure 53 . CIFAR-10 sample.....	71
Figure 54 . The sample of Fashion--MNIST.....	72
Figure 55 . Sample of SMILES.....	72
Figure 56 . Sample of dogs vs. Cats .....	72
Figure 57 . Sample of Pascal VOC (classification).....	73
Figure 58 . Sample of Pascal VOC (segmentation).....	73
Figure 59 . Sample of MS-coco.....	74
Figure 60 . HMDB-51 data set sample.....	74
Figure 61 . UCF-101 data set sample.....	75
Figure 62 . Sample of RAVDESS.....	76
Figure 63 . Tradition emotion recognition method .....	78
Figure 64 . Difference dataset sample with high FPS and high-quality camera .	79
Figure 65 . Sketch of new pre-processing system.....	80
Figure 66 . Haar filters sample .....	81
Figure 67 . Haar-like decision diagram.....	81
Figure 68 . Face detection.....	82
Figure 69 . Example of a rotated face.....	82
Figure 70 . Alex-net training process.....	85
Figure 71 . GoogleNet structure training process.....	86
Figure 72 . ResNet structure training process.....	87
Figure 73 . (a) Real test (camera vision: angry sad fear neutral happy).....	88

## List of Tables

Table 1 Parameters of measure model	36
Table 2 YOLO9000 result compared with other framework on VOC 2007+2012 dataset.....	42
Table 3 Result of lip-net.....	52
Table 4 Number of features of each membrane .....	54
Table 5 Details of database.....	83
Table 6 The different parameter values chosen for training the networks.....	83
Table 7 Accuracy Comparison.....	84
Table 8 Results of three structures.....	88

## Abstract

The main purpose of this paper is to distinguished human expression using a deep learning method. This paper present a new preprocessing method to extract the features of human expression from videos, and then uses deep learning methods to analysis the human emotions. A facial expression is usually regarded as a fixed moment of human disposition. However, recently, researchers have realized the importance of time information for expression recognition. At the same time, the ability of feature extraction in deep learning has also received attention. The method used in this paper uses time information to distinguish expressions. The paper is divided into several chapters to describe the background technology of expression analysis. Relevant information mentioned includes: picture and video information; Basic knowledge of convolution neural network; The basic principle of recurrent neural network; Background technology of face features and traditional classification methods; Application of depth learning method in video; Related technologies of object detection and the amount of classical deep learning models. With this background knowledge, a complete video facial expression analysis scheme can be formed.

Chapter 1 is the overall planning of the thesis. Chapter 2 systematically introduces the relevant knowledge of deep learning, including the composition of deep convolution network, the composition of deep loop network, the use of activation function, classical classification network and target detection network. Pretreatment is also needed in facial expression recognition, including face detection, face correction, etc. Chapter 2 introduces face processing algorithms, such as Viola&Jones face detection model and multi-person face detection model. At the same time, the knowledge of actual model training and the analysis experience of experimental results are introduced. At the end of the chapter 2, the classical data sets used in learning are listed. In chapter 3, a new pre-processing method for video temporal feature extraction is proposed. The whole expression recognition process includes face recognition, face alignment and the training of depth learning classifiers. RAVDESS video data set is used in training the depth learning model. After reasonable training, the model was tested in video and real-time video and achieved acceptable results. Chapter 4 summarizes and assesses practical work and discusses future work direction, combining the experience from this work with the problems encountered.

---

## Table of contents

Chapter 1.....	10
1.1 Preface.....	10
1.2 Motivation.....	11
1.3 Summary of Original Contributions.....	12
1.4 Organization of the Thesis.....	13
Chapter 2: Deep learning and emotion recognition.....	14
2.1 Introduction.....	14
2.2 Images and videos.....	15
2.3 Convolution neural network.....	16
2.4 Recurrent Neural Network.....	27
2.5 Classic CNN classification model.....	30
2.5.1 Alex-net.....	31
2.5.2 Google-net.....	31
2.5.3 Res-net.....	33
2.6 Classic CNN Object Detection Model.....	34
2.6.1 Indicators for measuring the object detection model.....	35
2.6.2 Deep Learning Target Detection Algorithm Based on Region Proposal.....	37
2.6.3 Deep learning target detection algorithm based on regression method.....	41
2.7 Application of deep learning in video expression analysis.....	44
2.7.1 Conv LSTM.....	46
2.7.2 3D CNN.....	49
2.7.3 Summary.....	52
2.8 Face Analysis.....	53
2.8.1 Face Detection Analysis Model Based on Ada-Boost Algorithm.....	53
2.8.2 Face detection ---MTCNN.....	55
2.8.3 Summary.....	58
2.9 Model training.....	58
2.9.1 Label ---one hot.....	58
2.9.2 Loss function and optimism.....	59
2.9.3 Under-fitting and over-fitting.....	62
2.9.4 Training Sets, Test Sets, and Validation Sets.....	63
2.9.5 Step and epoch.....	64
2.10 Transfer learning.....	65
2.12 Video data set.....	74
2.13 Summary of Chapter 2.....	76
Chapter 3: Human Emotion Recognition in Video using Subtraction Pre-Processing.....	77
3.1 Introduction.....	77
3.2 Related work.....	77
3.3 Our approach.....	79
3.4 Model structure.....	79
3.5 Face Detection.....	80
3.6 Face Alignment.....	82



---

3.7 Result and analysis.....	83
3.8 Conclusions.....	89
Chapter 4 : Conclusion and Future work.....	89
4.1 Conclusion.....	89
4.2 Future work.....	90
Reference.....	91

# Chapter 1

## 1.1 Preface

Facial recognition and emotion recognition technology appeared a few decades ago[1,2,3,5], but this technology was not perfect due to the development of technology at the time. In daily life, facial recognition or emotion recognition is one of the functions that we humans use long ago. For computers, they do worse than personal at that time. With the development of computer hardware, computers have been able to do this work initially[7]. Face recognition, emotion recognition, and other topics have attracted the interest of many researchers in various fields of science (graphics, computer science, pattern learning)[8,9,10,11]. Usually, human emotions present through the entire body and face, but the face is considered by us to be the area where emotions are most concentrated. In the past few decades, expressions have divided into several expression patterns. The computer recognizes changes in other people's feelings by obtaining specific patterns. Researchers also accept this classification and recognition method. As image processing and computing hardware advances, pattern learning becomes more efficient and facilitates the generation of other pattern recognition algorithms. However, there are still some unresolved challenges in various identification systems. Also, some noise that affects image processing. For example, different levels of light and face's angle.

Humans can easily discern the changes in other people's emotions, even in complex and largely disturbed environments. Computers are expected to have this powerful anti-interference ability in recognizing expressions, so expression recognition is still a hot topic. Since some achievements have made in the expression recognition in pictures, people began to pay attention to expression recognition in video. A video is a collection of pictures produced continuously over time.[12] Because the video is a collection of images, people can still perform facial expression recognition on each image in the video and conclude integrating the overall results. This method ignores the relationship of pictures in the video, which increases computational cost. [13] This paper proposes a new video preprocessing method, combined with a neural network classification method, to build a new expression recognition processing system in the video. Some face analysis techniques and image processing techniques, which have open-sourced, are used in the system. Finally, different neural network models were used to train and compare the results. Also, the video's expression recognition system and real-time expression analysis function are realized.

## 1.2 Motivation

The basis of human world communication is the transmission of information. Through some language and some non-verbal body movements, people judge a person's emotional changes.[14] The face is considered to be the most concentrated area that reflects the changes in people's feelings. In daily communication, people can judge people's emotional changes through facial movement information.[15] In terms of artificial intelligence, facial information is also often used as the main feature of emotional recognition. In the masterpieces of great scientists, artificial intelligence ushered in a new development climax. Deep learning and neural networks have become a new machine learning tool for dealing with various problems, and have achieved excellent results on some issues. Facial recognition and emotional recognition have also ushered in new developments. [16,17,18,19] The primary identifiers for expression recognition are pictures, videos, and sounds. Analyzing emotions will significantly help people's lives in criminal investigations, interviews, and daily communication and entertainment activities. People's daily communication activities are more similar to video mode, with time characteristics. The picture is a view at a specific time point. Therefore, expression recognition in the video is more conducive to the development of artificial intelligence devices. Information about human feelings can help machines or robots to make more accurate feedback, and also has significant effects in social work areas (e.g. the criminal investigation, entertainment, human-computer interaction, mental illness monitoring).[20,21,22]

### 1.3 Summary of Original Contributions

The main research results of this paper are as follows:

The first contribution is to summarize the classic neural network models that have emerged. The classification network, as well as the object detection recognition network and face detection algorithm, are studied. The structure, advantages, and disadvantages of traditional classification models, such as Alex-net, Google-net, Res-net, are discussed. The classification network is still developing. In the image-net competition, the classic classification model has achieved high accuracy. A well-built inside classification network needed in the object detection recognition network, which determines the performance of the net. This paper studies and summarizes the existing systems, including R-CNN, Fast-RCNN, Faster-RCNN, YOLO, SSD. Also, the model MTCNN specifically used to detect faces.

The second contribution is to introduce the problems that appear in model training and the concept of transfer learning. The quality of the model is inseparable from the setting of the parameters during training. The Loss function, the one-hot label, and the analysis of the training effects are skills that should know during the actual training model.

The third contribution is the use of a new image preprocessing method to deal with the video. Based on the inspiration gained from my discussion with my supervisor, we believe that there are too many background disturbances in the video expression recognition system. The traditional way is to treat each frame as an input, which causes the unrelated background to integrate into the calculation, and the computer cannot distinguish between valid data and invalid data. This new method can reduce the interference term. In the ideal case, each valid element will use, and the unrelated item will become zero. This method has been proven to be usable, and better processing techniques can improve this method. The ICMLC conference has accepted the papers on this approach.

## **1.4 Organization of the Thesis**

The research content of this paper is as follows:

The first chapter describes the purpose and motivation of the research, as well as the main contributions.

The second chapter first introduces the concept of the neural network and the advantages and disadvantages of the classic neural network.

First of all, the knowledge of images and videos is introduced. The origin and development of neural networks, classic classification networks, and object detection networks introduce to developing a new face recognition system. Then, discuss the training methods for sequence data, speech, and video information. What's more, in the realization of facial expression recognition, the face recognition model and the video analysis expression model are described.

The third chapter demonstrates a new video information pre-processing method in the video expression analysis system, in which a neural network and Haar-Like face detection algorithm are used.

---

## Chapter 2: Deep learning and emotion recognition

### 2.1 Introduction

In recent years, deep learning has become a hot research field. Based on machine learning, the accuracy and efficiency of computer work are once again improved. In January 2016, Alpha-go, a machine-based chess player based on intensive reinforcement learning, defeated the European Go Championship and once again triggered widespread development of deep learning applications.

Deep learning used in a wide range of applications. For example, character recognition in the criminal investigation, human-computer interaction in the robot industry, recognition of action behavior, speech recognition in the automotive industry, and development of autonomous vehicles. [21,22,23]It is also hot in expression recognition area. Human expressions are the primary source of interactive information for people to communicate. In the current society where GPUs are developing rapidly, various expression recognition algorithms have greatly recognized. However, there are still some problems that need to resolve. Deep learning neural networks mimic human thinking patterns, but some aspects are not as perfect as humans. It also needs to be continuously optimized when applied to real life. In the aspect of expression recognition in video, the computer can only judge essential features through the picture, while humans can easily find the transformation of the expression.[32] Humans focus on the features and determine the conversion of emotions. Deep learning also can automatically extract features and learn a pattern from them. Consider the similarities between deep learning and this study. The reasons for the final use of deep learning are as follows[32]:

1. Deep learning has strong feature extraction capabilities and has abstract features to learn patterns from features.
2. The development of GPUs provides dominant computing power that can be used to train complex enough deep neural networks to achieve high accuracy.
3. The face structure is different, but the expression patterns are similar. The use of a large amount of data to learn the expression of the expression can effectively improve the practical application.
4. This paper proposes a preprocessing method that enables deep learning to have better data to extract features and build networks.

In essence, this study preprocesses the input data according to human expression recognition habits and uses the deep learning classifier to process the data to construct logical model of video expression recognition.

## 2.2 Images and videos

Before using the deep learning classifier, some of the knowledge of pictures and videos needs to introduce. Humans see the colorful world through the reflection and refraction of light. The computer gets an array from the camera. The most classic image structures are images in JPG and PNG formats. There are three channels in such images, which will correspond to the neural network channels introduced later. The three channels in the picture are red, yellow, and blue channels[33,34]. Red, yellow, and blue is believed to constitute any color known to man[34]. As shown in Figure.1, pictures in machine vision use color gradation to show the degree of brilliance. The commonly used color gradation has 256 and 65536 steps. As shown in Figure 2, zero means no color, the larger the number, the brighter the color at that position.

Neural networks abstract feature points from these numbers and use them to build models. In addition to the convolutional neural networks used in deep learning, there are several ways to extract features from these numbers, such as Haar-like[35], LBP[36,37], SVM[38], and the classic Viola & Jones[2]algorithm.

Video is like a time-added feature compared to images. Make the picture look more vivid and real, and more close to the environment of daily human communication. You can get a video by taking pictures continuously with your camcorder. Every picture in the video is the same size and has the same quality. As shown in Figure 3, the video changes over time. An essential feature in the video is called FPS, which is short for frame per second. The value of FPS determines the number of pictures taken in a unit of time and also determines how much the two adjacent images change.



Figure 1 Color plate [4]

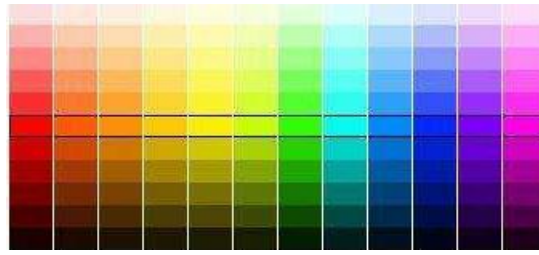


Figure 2 Color step [4]



Figure 3 Video frames [4]

## 2.3 Convolution neural network

Convolutional neural networks are at the heart of deep learning. Convolutional neural networks can extract features from pictures or signals. Supervised learning and unsupervised learning are the two training ways of deep learning neural networks. [39]Supervised learning requires a label to correspond to the input, while non-supervised learning only acquires features from existing data and learns autonomously[39]. Convolutional neural networks use computational nodes to simulate neurons in human thinking. The neurons are connected and extract the characteristics of the input information layer by layer[39].

Similarly, the artificial neural network simulates this process to process the obtained image data. Supervised convolutional neural networks constructed from several essential layers. An introduction to the layer will describe later. A sample example of the neural network shown in Figure.4. This figure illustrates the meaning and role of the neural node. Some symbols need to explain.

- $X_0$ : the value of  $X_0$  means the input of the signal
- $W_0$ : weight of the input signal
- $f$ : activation function



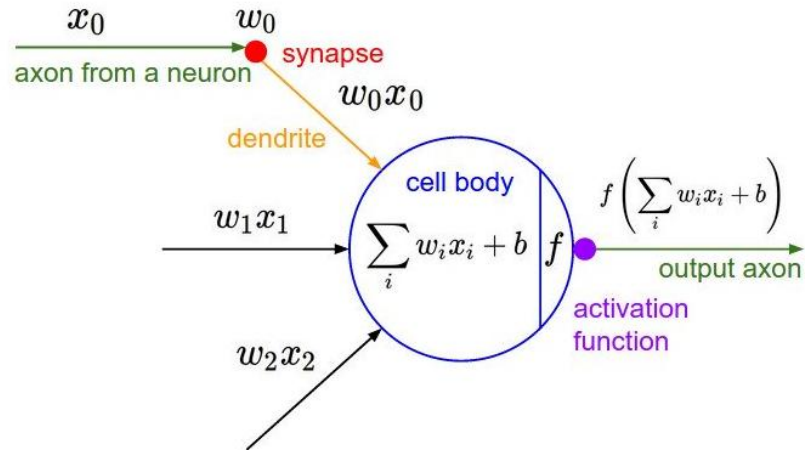


Figure 4 Convolutional neural network neuron transfer data[7]

In Figure.4,  $X_1$  and  $X_2$  represent other analog neural signal inputs. In a single neuron, the data is combined using a linear relationship, and the activation equation  $f$  gives a nonlinear correspondence to each node. This picture shows the basic algorithm of the neural node[39].

### Convolution neural network layers

The CNN structure extracts features through convolutional layers. CNN generally consists of two main parts: the input and output layers and the hidden layer.

#### Part1: Input layer and output layer

The input layer usually uses as the first layer of the neural network. The line between two nodes is the weight from into node to the output node. The input layer and output layer always set at the start and the end of the neural network. The layer between these two layers is the Hidden layer, which is the main body of the neural network.[40]

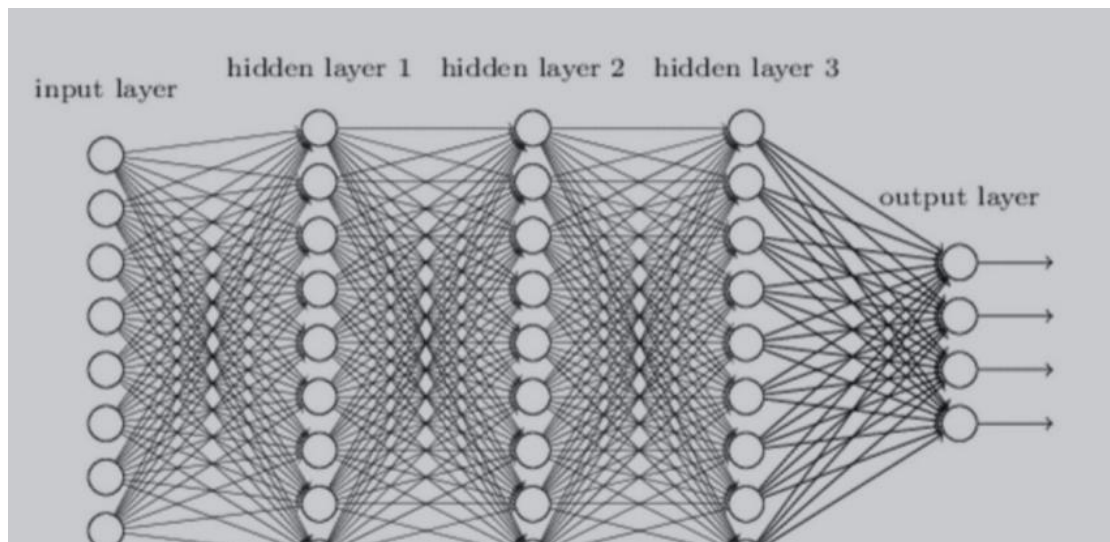


Figure 5 The simple neural network structure. [39] The input and the output layer set at the start and the end of the whole structure, hidden layers are placed in the middle.

## Part 2: Hidden layers:

Hidden layers include plenty of function layers. Almost all the calculation takes place in the hidden layers. The number of hidden layers determines the depth and complexity of the entire neural network.[40] Usually, the hidden layer contains the following common layers. Each layer has unique features, and each layer will describe below.CNN layer.

- Fully connected layer
- Pooling layer
- Dropout layer
- Activation function layer
- Softmax layer

### 1. CNN layer

The convolutional layer is essential in the deep learning structure. Also, the convolutional layer is used to extract features from the raw data. [40] Each layer has a limited ability to obtain features, so multiple convolutional layers are needed to get depth features. Figure.6 shows the 2D convolutional process. The 3x3 matrix in Figure.6 called filter (also called convolution core). From the beginning to the end, a feature map will generate when the convolution operation finishes by one filter.[40,41,42]

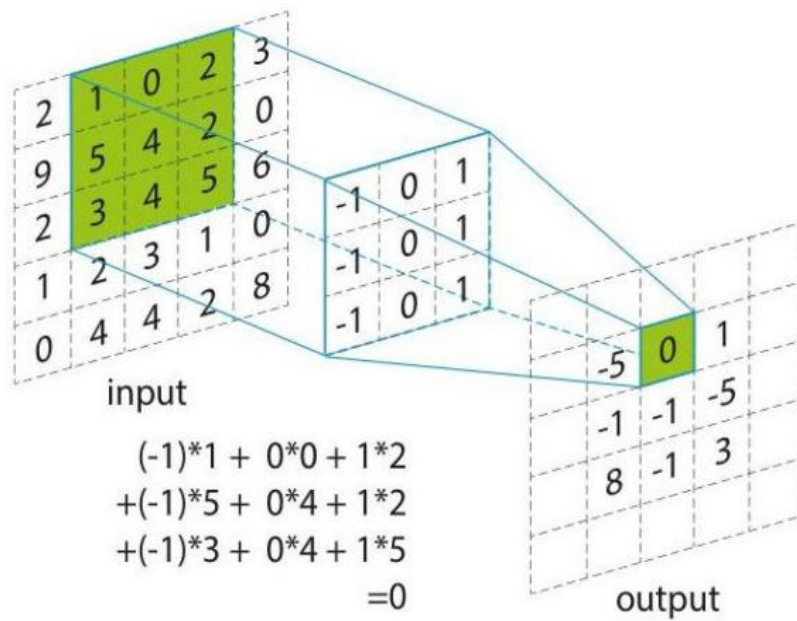


Figure 6. 2D convolutional process [40]

Convolution has three modes: full, same and valid. Figure 7- figure.9 shows different convolution effects and ranges.[40]

The orange portion of the figure is the original image data, the cyan portion is the convolution kernel, and the gray part is the coincident portion. The convolution kernel size is 3x3, and the original image size is 7x7. The white part is 0. The convolution process begins at the moment that the convolution kernel and the image just start to overlap if convolution operation is set in full mode. Full mode will cause the convolved features size to be bigger or equal to the original image. The Same mode convolution ensures that the convolved features are the same size as the original image. [40]The beginning of the same mode is the moment when the center of the convolution kernel begins to intersect the original image. The beginning point of the valid mode convolution is the moment when the convolution kernel completely coincides with the original image. [40]As can be seen from Figure.9, the convolution feature will be smaller than the original image. When processing an oversized picture, the amount of calculation can reduce by such a convolutional form.

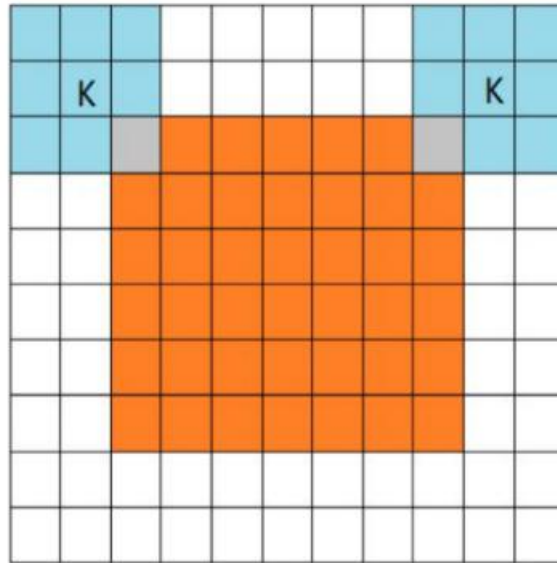


Figure 7. Full mode convolution [44]

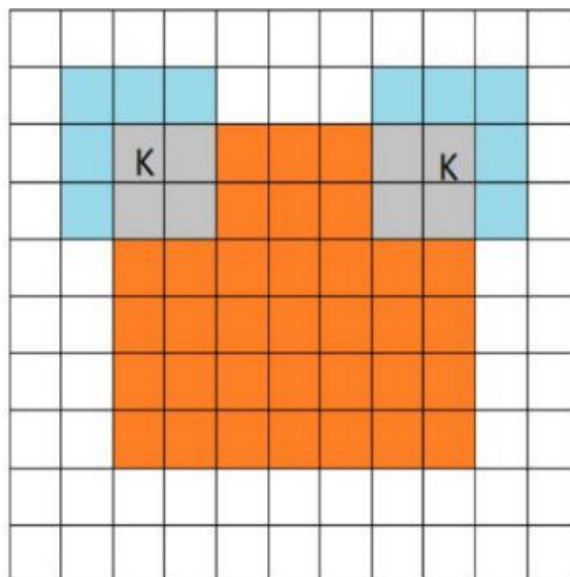


Figure 8. Same mode convolution [44]

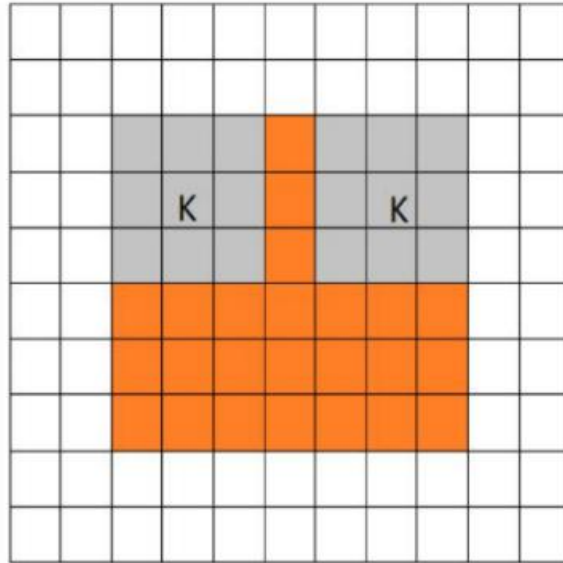


Figure 9. Valid mode convolution [44]

## 2. Fully connected layer

The fully connected layer usually used in the end part of the classification network. After the last convolution layer, the features of the image will connect to the full convolutional layer for the final classification. For example, when there are 1000 categories, the last fully connected layer will be 1000 dimensions (map 1000 classes). The fully connected layer maps the convolved 3D features to one-dimensional features so that the backpropagation algorithm can run and get the final result. When 2D convolution does not appear, the neural network usually consists of the fully connected layer. However, too much running and computational cost are not practical, and the spatial information in the image also disappears in learning.[44]. 2D convolution keeps the spatial feature, which is a significant feature and makes deep learning framework more powerful.

## 3. Pooling layer

The pooling layer is set up to reduce the size, which will inevitably reduce the number of features, but because of the large number of features, pooling is feasible. [40,42,44]The right amount of pooling can speed up the calculation and focus on training important features in training. However, excessive pooling can also lead to information loss seriously, and the loss function cannot converge, resulting in an unsatisfactory deep learning model.

There are also several pooling modes in the pooling layer. Figure.10 shows two standard methods of pooling, Max pooling, and Average pooling. ‘Stride’ is also an essential parameter in the pooling process. This parameter has the same meaning as the parameters in CNN layers, indicating the size of the sliding area when processing

an image or feature. As shown in Figure.10, when the stride set to 2, the 4x4 matrix is reduced to 2x2 by the pooling layer size. During the operation, the green part of the 4x4 matrix will replace by a feature, which represents the green part. In max pooling, the maximum of the four numbers is selected to represent the 2x2 feature. In the average pooling, we choose the average of the four numbers to replace the original 2x2 matrix. Pooling layers formally reduce the dimension of features in this way. Although some high-frequency features have sacrificed, the overall computing speed will be significantly improved. The new feature can regard as a new feature throughout the statistical methods.

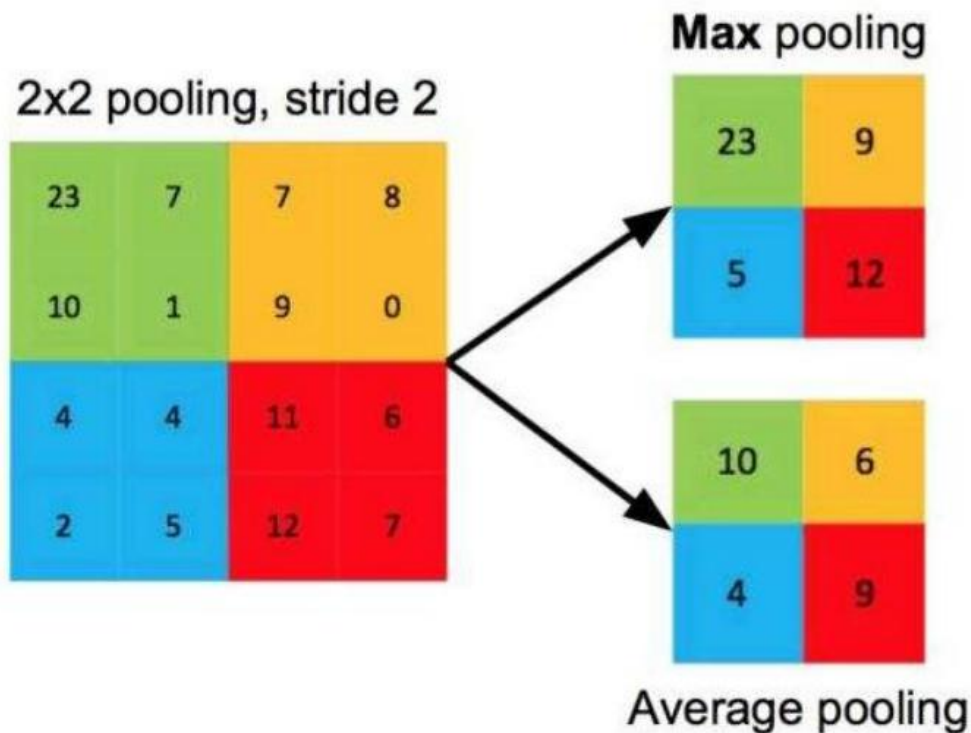


Figure 10. Max pooling and Average pooling[45]

#### 4. Dropout layer

In the process of neural network training, the network design is very complicated, it is easy to cause over-fitting, and the generalization ability will greatly reduce. In a 2012 paper, Hinton proposed a new structure called the dropout layer. The role of the Dropout layer is to train only a random subset of neurons during training. This training mode reduces the dependence between each neuron and improves generalization.[45,46] Although fewer neurons used during training, the need to set random functions and changes in parameter updates due to randomness reduces the training speed. The setting of the random function refers to setting the random neurons to perform weight update. However, the actual application and the speed of the model testing phase will not be affected. [45,46] The establishment of the Dropout

layer is not required, and it is necessary to refer to the actual project data volume to make adjustment decisions. Figure.11 shows the difference before and after using the Dropout layer. It can be seen from Figure.11, that some neurons have no parameter updates during the update process.

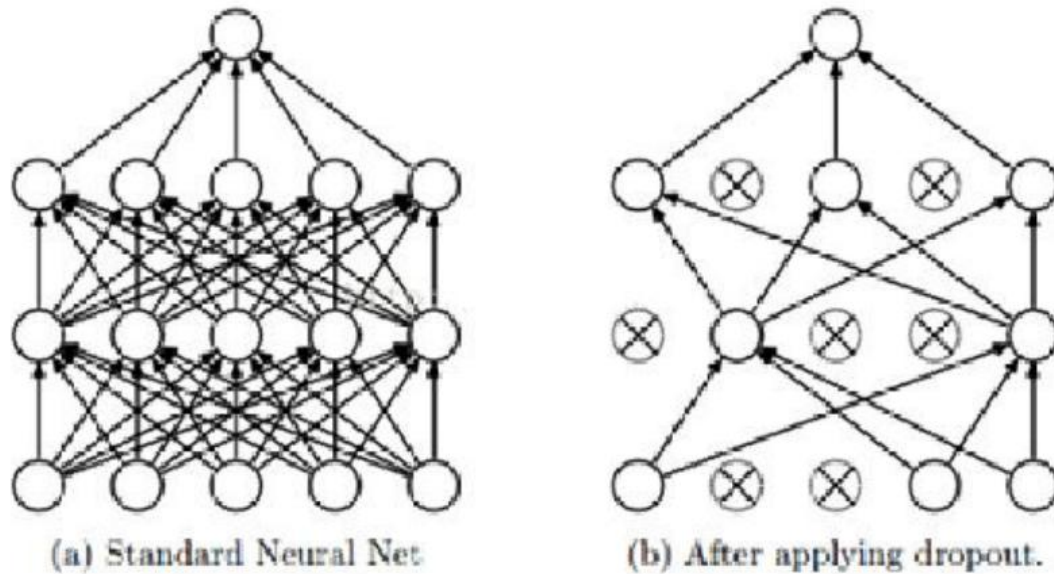


Figure 11. application of Dropout layer.[46]

## 5. Activation function layer

The activation function layer usually placed behind the convolutional layer in the neural network. The convolution process is a linear function-driven computational model. In real life, logic and problems are generally non-linear. We also think that the image features and the understanding of the signal also have some implicit non-linear relationships. The activation function deals with the features of the convolutional layer and uses a built-in non-linear logic function to realize the non-linear relation algorithm.[46]. The activation function also maps the original values to a more manageable threshold to solve the problem of large parameters. The activation function increases the computer's ability to understand pictures and signals.

Several commonly used non-linear activation equations:

### ***Sigmoid function:***

The Sigmoid function is a kind of non-linear function that maps data to the range of (0, 1). Due to the high computational cost of this non-linear function and the influence of the gradient explosion problem, it replaced by other non-linear functions. It is a non-linear activation function that used few years ago[46]. Equation (2.1) is the mapping formula for sigmoid function,  $x$  for raw data and  $S(x)$  for output. Figure.12

shows the shape of the sigmoid function. According to Figure.12 that sigmoid is non-linear, and the output of the sigmoid in the range (0 ,1).

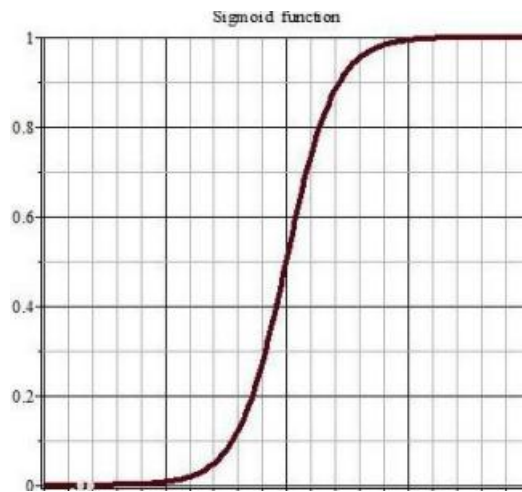


Figure 12. Sigmoid function[46]

### ***Tanh* function:**

The *Tanh* function[47] maps the data to (-1 1). This non-linear function is somewhat better than the sigmoid function, but the gradient explosion and the problem of negative parameters are also inevitable in training. Equation (2.2) is the mapping formula for *Tanh* function,  $x$  for raw data and  $\tanh x$  for output. Figure.13 shows the shape of the *Tanh* function. It indicates that *Tanh* is non-linear, and the data is in the range(-1,1).

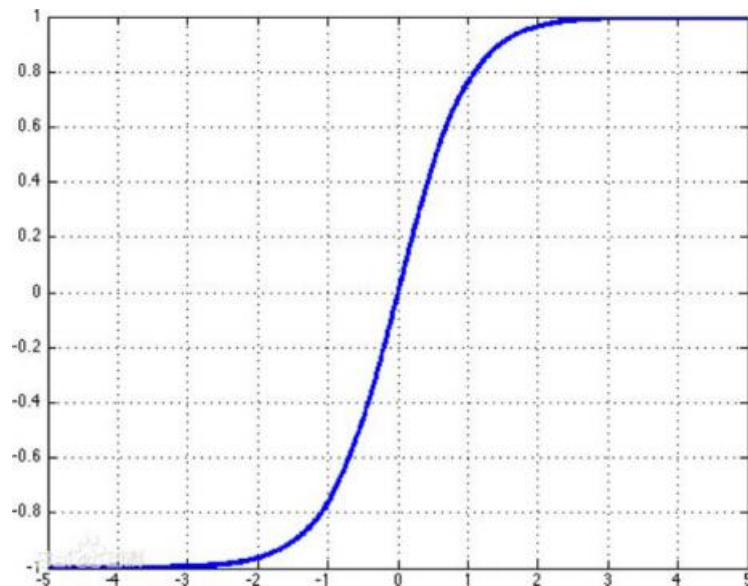


Figure 13. Tanh function[47]

### ***Relu/P-Relu:***



The Relu function[25,48] currently used as the activation function that is considered to be optimal now. The *Relu* function has many branches, such as leaky *Relu0*.[49] Relu function turns the negative value to 0, and only selects positive values for calculation, which is faster than other functions in processing speed. Equation (2.3) is the mapping formula for *Relu* function,  $x$  for raw data, and  $f(x)$  for output. Figure.14 shows the shape of the *Relu* function. It indicates that *Relu* is non-linear, and the data is eliminated negative values and keep the position value.

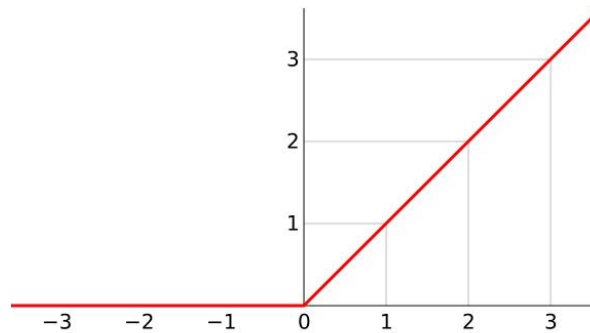


Figure 14. Relu function[25]

$$f(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{if } x \leq 0 \end{cases} \quad (2.3)$$

Other activation functions show in Figure.15. Different activation formulas have different functions. They set in different neural networks for facing different problems and tasks. The choice of the activation function is also related to the network structure. For Recurrent Neural Networks, some activation function might be more practical. For example, *Tanh* is usually used instead of sigmoid in RNN[40]. The choice of activation function requires plenty of experimentation to conclude.






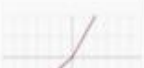





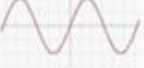
Name	Plot	Equation	Derivative
Identity		$f(x) = x$	$f'(x) = 1$
Logistic (a.k.a Soft step)		$f(x) = \frac{1}{1 + e^{-x}}$	$f'(x) = f(x)(1 - f(x))$
Tanh		$f(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
ArcTan		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2 + 1}$
Rectified Linear Unit (ReLU)		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Parameteric Rectified Linear Unit (PReLU) <sup>[2]</sup>		$f(x) = \begin{cases} \alpha x & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Exponential Linear Unit (ELU) <sup>[3]</sup>		$f(x) = \begin{cases} \alpha(e^x - 1) & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} f(x) + \alpha & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
SoftPlus		$f(x) = \log_e(1 + e^x)$	$f'(x) = \frac{1}{1 + e^{-x}}$
Bent Identity		$f(x) = \frac{\sqrt{x^2 + 1} - 1}{2} + x$	$f'(x) = \frac{x}{2\sqrt{x^2 + 1}} + 1$
SoftExponential		$f(\alpha, x) = \begin{cases} -\frac{\log_e(1 - \alpha(x + \alpha))}{\alpha} & \text{for } \alpha < 0 \\ x & \text{for } \alpha = 0 \\ \frac{e^{\alpha x} - 1}{\alpha} + \alpha & \text{for } \alpha > 0 \end{cases}$	$f'(\alpha, x) = \begin{cases} \frac{1}{1 - \alpha(x + \alpha)} & \text{for } \alpha < 0 \\ 1 & \text{for } \alpha = 0 \\ e^{\alpha x} & \text{for } \alpha > 0 \end{cases}$
Sinusoid		$f(x) = \sin(x)$	$f'(x) = \cos(x)$
Sinc		$f(x) = \begin{cases} 1 & \text{for } x = 0 \\ \frac{\sin(x)}{x} & \text{for } x \neq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x = 0 \\ \frac{\cos(x)}{x} - \frac{\sin(x)}{x^2} & \text{for } x \neq 0 \end{cases}$

Figure 15. Activation table. [50]

## 6. Softmax layer

The role of the Softmax layer is to give the classification a fully connected layer a rating. When testing a classification model, the final classification output has no clear range of intervals. So, we cannot find the most suitable answer from the final classification output. We tend to give a ratio to all the final results. The softmax layer does this through a nonlinear equation. The Softmax function shows as equation 2.4. [51]. Where  $K$  represents the number of all categories and  $k$  represents the output of

each category received from the fully connected layer. The final result  $\sigma(Z)_j$  is the result of each classification after passing through the softmax layer[51]. The softmax function maps the results of the fully connected layer into a range of (0, 1).

$$\sigma(Z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad (2.4)$$

## 2.4 Recurrent Neural Network

Recurrent neural networks are also a kind of deep learning. Unlike CNN, RNN pays more attention to the relationship in time information sequences. Time series refers to the information flow with time characteristics, such as the context in machine translation, the time context in the video, the arrangement of the password.[50] CNN learns the spatial pattern in the image, while the RNN learns the time information pattern. RNN is Similar to CNN; they all have an input layer, an output layer, and a convolution layer. In the RNN structure, we hope to obtain the time information (text, video) through deep learning methods, so in the original RNN structure, sequence nodes are set. A simple RNN schematic shows in Figure 16.

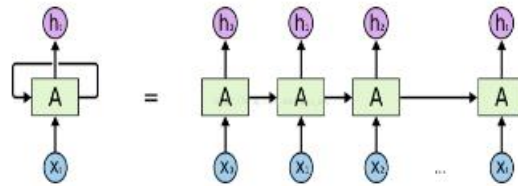


Figure 16. Traditional RNN structure

From Figure 16, we can see that each node  $A$  receives an input to produce a result and also create gain to the adjacent node. This gain ensures that each subsequent output is not only related to the current input, but also the previous state. The relationship between such sequences can be understood better by the network.[52]

LSTM:

LSTM[53] is short for Long-Short-Term-Memory. From a structural point of view, RNN is destined to be used to understand the information of the sequence. However, from the simple structure in the Figure.16, we find that only gain or feedback from the previous state does not understand the global structure well. For example, the pronouns used by people may be far away from the verbs. It is likely to cause the problem of losing the subject in machine translation. To solve this problem, based on the understanding of RNN, LSTM[53] was proposed by Hochreiter & Schmidhuber (1997) and improved and supplemented by Alex Graves[54]. The improved LSTM has achieved excellent results in many problems. Figure.17 shows the structure of a

traditional LSTM.

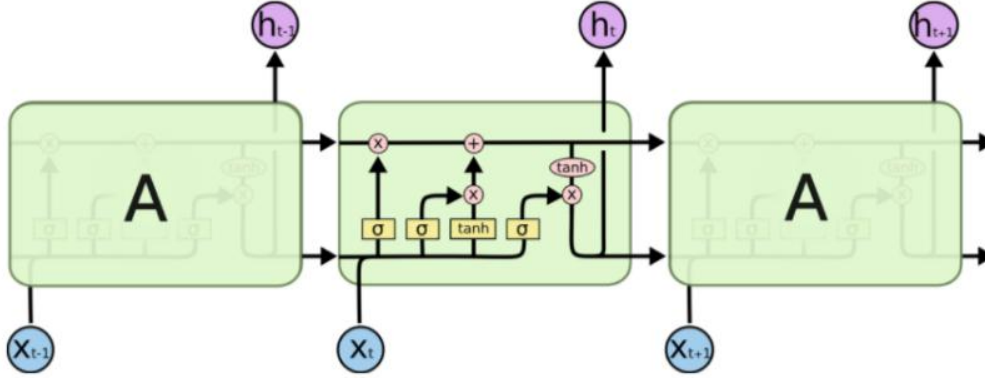


Figure 17. Traditional LSTM structure[53]

Compared to the traditional RNN architecture, LSTM[53,54] uses a 'gate' structure to achieve the overall sequence information trade-off. The gate structure has a sigmoid function and a bitwise multiplication operation. The Sigmoid function maps all number to the range of (0,1), and then multiply by bit depends on how much information can pass through the 'gate' structure. For example, if the gate is closed, then no data can pass through the 'gate.' If the gate is open, then all information can pass.

Equations (2.5)-(2.10) use to describe the output of each part of the LSTM structure[53,54]: where  $i_t$  represents the input,  $f_t$  represents the forgetting gate,  $\bar{C}_t$  represents the candidate memory gate,  $C_t$  represents the current time state,  $o_t$  represents the output gate, and  $h_t$  represents the final result.

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i) \quad (2.5)$$

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f) \quad (2.6)$$

$$\bar{C}_t = \tanh(W_C * [h_{t-1}, x_t] + b_C) \quad (2.7)$$

$$C_t = f_t * C_{t-1} + i_t * \bar{C}_t \quad (2.8)$$

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o) \quad (2.9)$$

$$h_t = o_t * \tanh(C_{t-1}) \quad (2.10)$$

GRU:

Based on LSTM, more variants of LSTM have been developed, such as GRU and bidirectional LSTM. GRU[55] is consistent with LSTM in its overall structure, but there are differences in function updates. Figure.18 shows a structure of GRU Unit. [55]. The specific function update mode is given by Equation (2.11)-(2.14). Where  $r_t$  stand for reset gate,  $z_t$  stand for renew gate,  $\hat{h}_t$  represents the candidate memory gate and  $h_t$  represents the current time state. Compared with LSTM, GRU[55] has

similar time information memory function. But, the way to update parameters is more simple than LSTM[54].

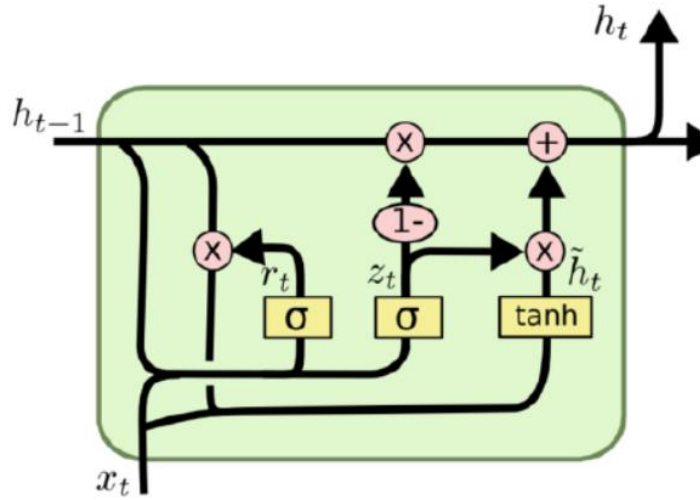


Figure 18. GRU Unit structure[54]

$$r_t = \sigma(W_r X_t + h_{t-1} U_r + b_r) \quad (2.11)$$

$$z_t = \sigma(W_z X_t + h_{t-1} U_z + b_z) \quad (2.12)$$

$$\hat{h}_t = \tanh(W X_t + h_{t-1} r_t U + b) \quad (2.13)$$

$$h_t = \hat{h}_t (1 - z_t) + z_t h_{t-1} \quad (2.14)$$

### Bi-LSTM:

LSTM is still a one-way sequence structure, while the emergence of Bi-LSTM[56] solves this problem, which learns a time series from both positive and negative directions. Bi-LSTM[56] uses LSTM as the basic unit for feature extraction in both forward and reverse directions. Similarly, other RNN structures can connect in a bidirectional mode. The Bi-LSTM[56] makes the Deep learning structure more complicated. Although the difficulty of training is improved, the ability to extract features of the network also enhanced so that RNN can handle more difficult time series problems.

Figure. 19 shows a block diagram of a Bi-LSTM[56]. The bidirectional LSTM has two sequences of positive and negative. The output of the final unit obtains features in two directions, so the final result combines the information in two directions. This method works well and achieve plenty of state of the art results. It plays an important role in machine translation and video recognition.

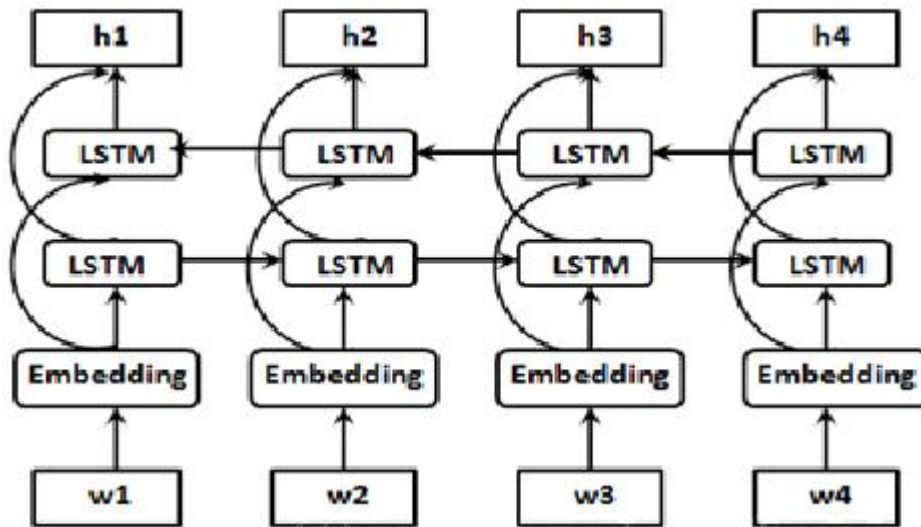


Figure 19. BI-LSTM structure in sequences understanding[56]

## 2.5 Classic CNN classification model

The previous section introduces the knowledge of CNN and RNN algorithms, and the classic architecture of the deep learning model is also the key to the effect of the classifier. A good model should make full use of the characteristics of CNN to obtain excellent parameters while saving computation time and computational cost. Now that the image-net competition has ended, the classic champion model in this competition has played an essential role in future development. This module will introduce Alex-net[23], google-net[24], and Res-net[25]. These three deep network models are selected because the three models have unique characteristics in all structures, and most of the other models base on these models for improvement.

### 2.5.1 Alex-net

Alex-net[23] is the 2012 image-net champion. In this competition, the deep learning model defeated the traditional machine learning model. Figure.20 is a model diagram of Alex-net in the paper.

The features divided into two parts after the image pass through the first convolution layer. The Alex-net uses two GPU to train, and each GPU is responsible for half of the calculation task.[23]. Two GPUs' result joined together to the final classified fully connected layer.

The entire structure consists of eight learning layers, including five convolutional layers and three fully connected layers. The setting of the number of filters is set more and more as the number of convolution layers increases, and the pooling layer is used to reduce the dimensions of the features.[23]. The overall structure is a one-way deep, stacking structure.

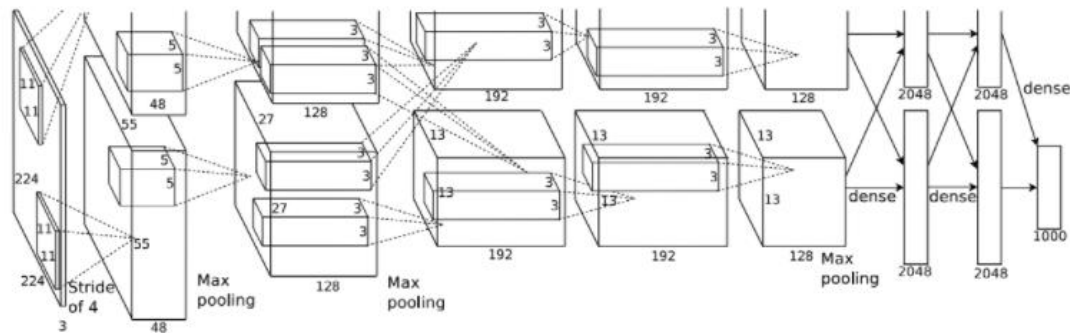


Figure 20. Alex-net structure[23]

### 2.5.2 Google-net

After the advent of Alex-net[23], the development of deep neural networks is moving toward deeper. The deeper the layer symbolizes the deeper dimension, the deeper understanding of the picture[24]. However, due to problems such as a gradient explosion in deep learning algorithms, the network needs to be designed to be lighter and has more feature mining capabilities. Google-net designed with a completely new network structure shows in figure.21.

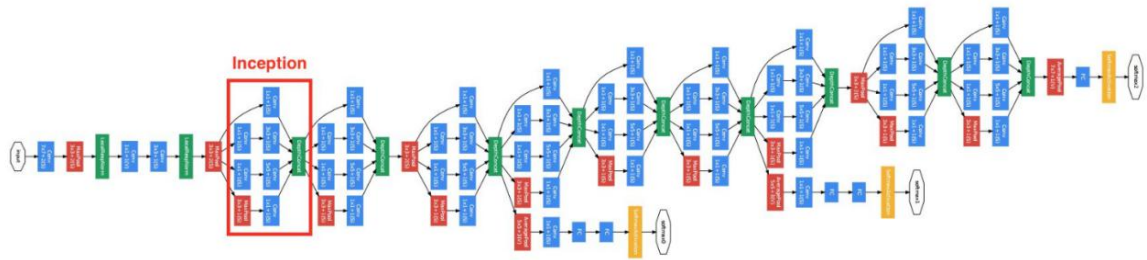


Figure 21. Google-net structure[24]

The overall network is still in the form of vertical accumulation. However, a unique inception structure and three auxiliary output structures designed in Google-net[24]. To increase the learning and understanding ability of the network, Google-net adds depth to 22 and uses the inception structure (shown in figure.22). Google-Net refers to the method of multi-scale processing using fixed multiple Gabor filters. Figure.22 shows the unique inception structure of Google-net:

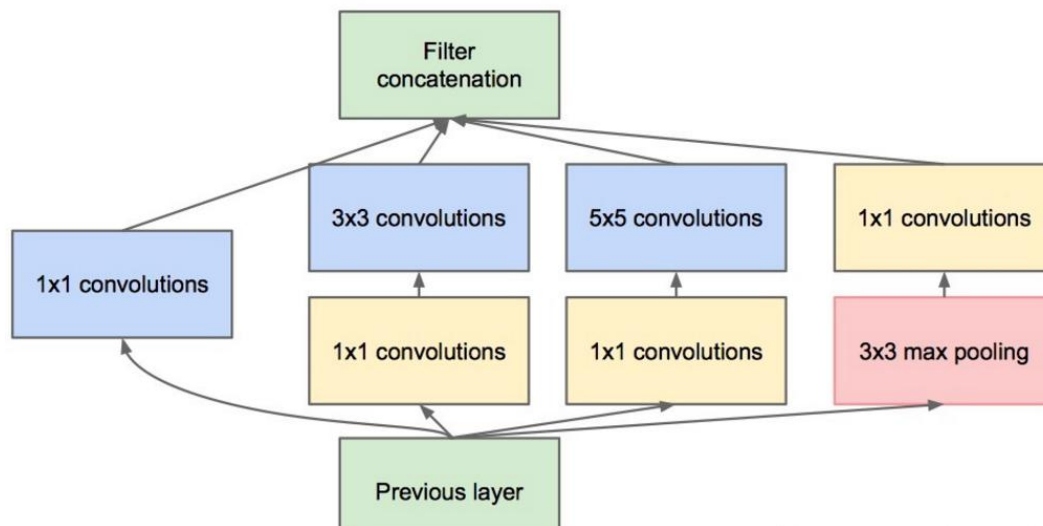


Figure 22. Google-net inception structure[24]

Google-Net's use 1x1 convolution to increase the depth of the network, and limit the size of the net. The parameters in pooling layers and convolution layers are designed to keep output size the same. [24]. In the subsequent development, the 3x3 size filter is confirmed to be the most suitable convolution size. The inception structure does not commonly use now.

The three output layers are set at different depths to represent the classification results given by various depth features. The first two classification layers whose depth is not high, given a weight of 0.3 and the last classification layer is used together to calculate the final classification result. Results at different depths are similar to the fusion of different models, which plays a positive role in the training of the entire Google-net and inception.



### 2.5.3 Res-net

In 2015, ImageNet shined and scored the first place in ImageNet's classification, detection, localization, and COCO detection and segmentation. The network before 2015 faced the same problem, the gradient explosion problem. When we use deeper networks to improve the network's ability to understand images, finding deeper networks does not produce more accurate results, but rather a greater error rate. The Kai Ming He team also explained the impact of this problem on deep networks in the paper. As shown in Figure.23[25]:

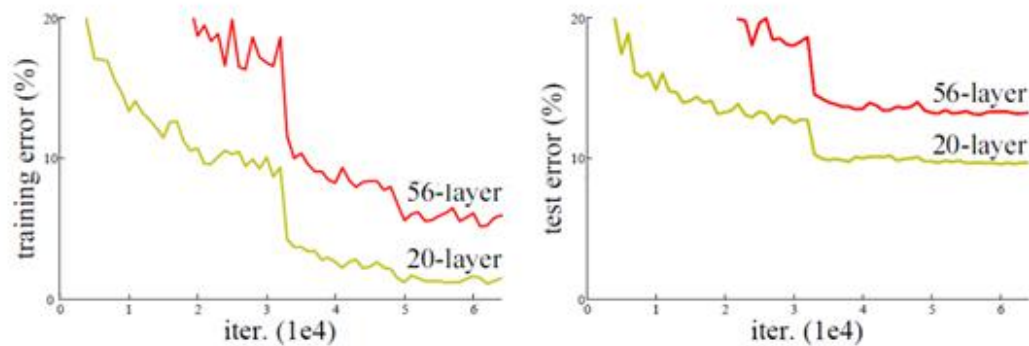


Figure 23. Deeper layer has a weaker behavior when network has over 20 layers[25]

Regardless of the test set or the training set, the performance of the 56-layer convolution structure is not as good as that of the 20-layer. This phenomenon is caused by a gradient explosion. In the gradient descent method training, since there is a multiplier less than 1 in the iterative process, the weight will be close to 0 in a certain depth network, and the ability to understand the feature is lost. Lead to a weaker fitting ability.[25] The problem is called gradient explosion or the gradient disappears. In response to this problem, the author of Res-net proposed a Residual structure as shown in Figure. 24:

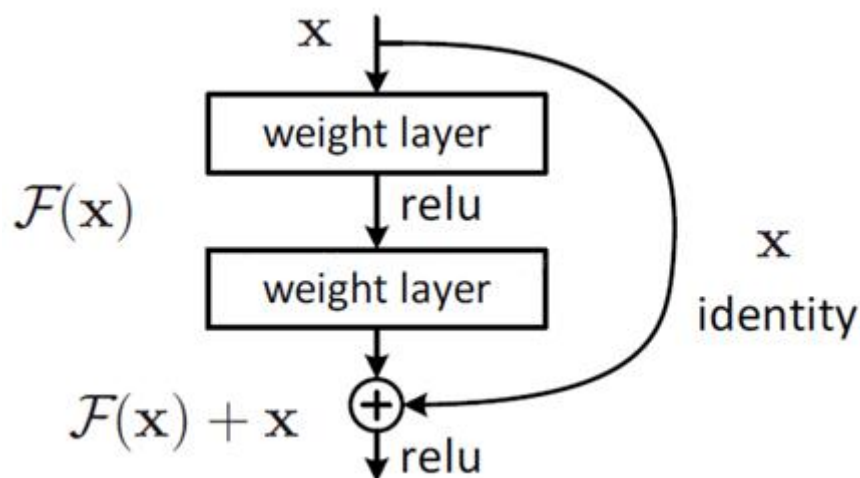


Figure 24. Res-net Unit[25]

The original input  $X$  obtains the high-dimensional feature after passing through two convolution layers. Before entering the next round of convolution, the high-dimensional features merge with the low-dimensional features, and the merged features are in the gradient.[57] The descent method will slow down the trend towards 0, which can set the network deeper, which increases the learning and understanding ability of the network. The deep residual network can set the depth of the network more freely. The original authors also give some setting parameters advice for different depth networks, as shown in Figure.25.

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2_x	56×56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		$1.8 \times 10^9$	$3.6 \times 10^9$	$3.8 \times 10^9$	$7.6 \times 10^9$	$11.3 \times 10^9$

Figure 25. Different Res-net structure settings[25]

## 2.6 Classic CNN Object Detection Model

The object detection is to find all the objects in the image and determine their position and size, which is one of the core issues in the machine vision field. Object detection applied to many scenarios, such as auto driver. Object detection has always been the most challenging problem in machine vision because of the different appearances, shapes, poses of various objects, and the interference of illumination and occlusion during imaging. As shown in Figure. 26, the traditional object detection method divided into three steps:

1)Area selection, which uses different sizes slides windows convoluted with a image

as a candidate area and then extracting the vision related to the candidate area. Such as, Haar features commonly used for face detection.

2) Feature extraction, such as HOG[58,59,60] (Histogram of Orientation Gradient) and SIFT[61,62] features widely used for human detection and standard object detection.

3) Classifier classification, that is, using classifiers training classified, such as the widely used Support Vector Machine (SVM[63]) model, Ada-boost[64], DPMC[65], RF (Random Forest) models[66].



Figure 26. Traditional object detection method[58]

In the traditional object detection, Felzenszwalb et al. proposed a multi-scale deformation component model (Deformable Part Mode UDPM) in 2008. DPMF performs based on HOG[58,59,60] and SVM[63], which makes full use of the advantages of HOG[58,59,60] and SVM[63]. Also, DPMF[65] made important breakthroughs in tasks such as image processing and face recognition. However, the traditional target detection algorithm has two main defects:

- 1) The sliding window-based region selection strategy has not targeted, the time complexity is high, and the window is redundant.
- 2) The hand-designed features are for diversity changes and Not very robust. The complexity of the DPM[66] model is relatively low, and the speed and accuracy of the target detection are low.

Although the development of deep learning has improved the accuracy of object detection, the performance has difficulty to breakthrough. In 2013, Girshick et al. proposed R-CNN[34,67], which increased the mAP on the VOC2007 test set to 48%. (mAP is a measure of the model; the details will introduce in the first section of this chapter.) By modifying the network structure in 2014. The mAP has risen to 66%, and the mAP of the ILSVRC 2013 test set has also raised to 31.4%. Since 2014, following Girshick et al. Proposed R-CNN with breakthrough results in the field of object detection, SPP-net, Fast R-CNN, and Faster R-CNN have appeared. R\_FCN[69], YOLO[29], SSD[30] the object detection algorithm based on deep learning has become one of the hot areas in the field of machine learning.

### 2.6.1 Indicators for measuring the object detection model

Before introducing the object detection, first, identify the indicators that measure the quality of the model in the target detection. In a classification model, when the model training ends and the test set verifies the accuracy, we use table.1 to perform statistics and obtain some parameters for measuring the model.

	predicted:P	predicted:F
actual:P	TP	FN
actual:N	FP	TN

Table 1. Parameters of measure model[70]

The horizontal labels represent the predictions in the test set, and the vertical cousins represent the actual labels. [70]. The results of the test set divided into four cases: TP (true/positive), FN (false/negative), FP (false/positive), and TN (True/negative), the meaning of which is the positive sample prediction is a positive sample, the negative sample prediction is a negative sample, the negative sample prediction is a positive sample, and the positive sample prediction is a negative sample, Respectively. After having these basic statistics, the formula for accuracy (P) and recall (P) obtained:

$$P = \frac{TP}{TP+FP} \quad (2.15)$$

$$R = \frac{TP}{TP+FN} \quad (2.16)$$

The accuracy rate P is used to measure the accuracy of a model; that is, the accuracy of the predicted result matches the label. The recall rate R is used to measure the effects of different classifications. In other words, whether all classes can well predict or have a good accuracy at the same time is shown through the value of parameter R. Accuracy and recall are statistical information based on the results of the test set. [70,71] If there is an imbalance between positive and negative samples in the test set, the reference value of the accuracy and recall values will decrease. To illustrate this situation, AP (Average Precision) and mAP (mean Average Precision) measurement parameters have proposed.

AP (Average Precision) also averages the accuracy. If the accuracy of one test set is equal to P1, multiple results can obtain when there is more then one test set. Use different test sets to reduce or avoid positive the problem of negative sample imbalance. [71,72]The formula of AP shown in equation 2.17.

$$AP = \frac{\sum_{i=0}^N P_i}{N} \quad (2.17)$$

N represents a different test set and represents the accuracy of different test sets. While the mAP (mean Average Precision) represents the average accuracy of overall categories, this measure can be a more comprehensive measure of a model.

---

## 2.6.2 Deep Learning Target Detection Algorithm Based on Region Proposal

The core component of the Region Proposal algorithm is the Convolutional Neural Network (CNN). Professor Yann Le-Cun first proposed the convolutional neural network in 1998.[72] First convolutional neural networks used as classifiers, mainly for image recognition. In 2006, Hinton et al. proposed a deep learning based on the concept of artificial neural networks. [73]Deep learning is a deep neural network with more hidden layers. It can extract deeper data features that cannot be learned by algorithms such as machine learning and can express data more abstractly and accurately. Based on this, Region Proposal solves the two main defects of the traditional object detection mentioned above. Since then, the CNN network has developed rapidly. The Top-5 error of Microsoft's Res-Net[27] and Google's InceptionV4 [18-19] model has dropped to less than 4%. Therefore, after object detection is used to obtain candidate regions, CNN can use to classify images to a certain extent. Improve accuracy and detection speed.

### 2.6.2.1 R-CNN

In the object detection algorithm, the characteristics of the exhaustive method have fully utilized for traversal. R-CNN[26] proposed by Girshick et al. uses selective search and uses clustering to group images. Get a hierarchical group of multiple candidate boxes. The test results of R-CNN on PASCAL VOC2007 have directly increased from 34.3% of DPMHSC[71] to 66% (mAP), which shows the great advantage of R-CNN. However, there are also many problems with the R-CNN framework[26]:

- 1) The training process has divided into multiple stages, and the steps are cumbersome, including fine-tuning the network + training SVM[38,58,59,60] + training border regression;
- 2) The training takes time and takes up a lot of disk space. For example, 5000 images will generate several hundred G feature files.
- 3) Slow speed, using GPU, VGG16[72] model takes 47s to process an image. For the problem of slow speed, SPP-NET gives a better solution[30].

### 2.6.2.2 SPP-Net

The SPP-NET algorithm is proposed in 2014 by Kaiming He[72]. The main steps of the object detection are:

- 
- 1) Regional nomination, use Selective Search to generate 2000 or so candidate windows from the original image;
  - 2) Area size scaling, SPP-NET[72] no longer normalizes the area size, but scales to  $=h$ , which is the shortest side length  $s$  of the uniform length and width, selected from  $\{480, 576, 688, 864, 1200\}$ , selected the standard is to make the scaled candidate frame size closest to  $224 \times 224$ .
  - 3) Feature extraction, which is, extracting features using the SPP-NET network structure.
  - 4) Classification and regression, similar to R-CNN, using SVM[38,58,59,60] to train the classifier model based on the above features and using border regression to fine-tune the position of the candidate frame.

SPP-NET solves the problem of bias caused by crop/warp and proposes SPP (Spatial Pyramid Pooling, SPP) layer[72,73,74], which makes the input candidate box size different. Although the use of SPP-NET has greatly sped up the target detection compared to R-CNN[26], there are still many problems:

- Training has divided into multiple stages, and the steps are cumbersome, including fine-tuning network + training SVM[58,59,60] + training border regression;
- SPP-NET fixes the convolution layer when fine-tuning the network, and only change the fully-connected layer. For a new task, it is necessary to refine the convolutional layer (the features extracted by the classification model pay more attention to the layer-by-layer semantics). In addition to the semantic information, the target detection task also needs the location information of the target.

In response to these two problems, in 2015, Ross Girshick proposed a streamlined and fast target detection framework for Fast R-CNN[27]. SPP-NET[72] structure shown in figure.27. From Figure.27, different sizes of scales' features are extracted and integrated by the final fully connected layer.

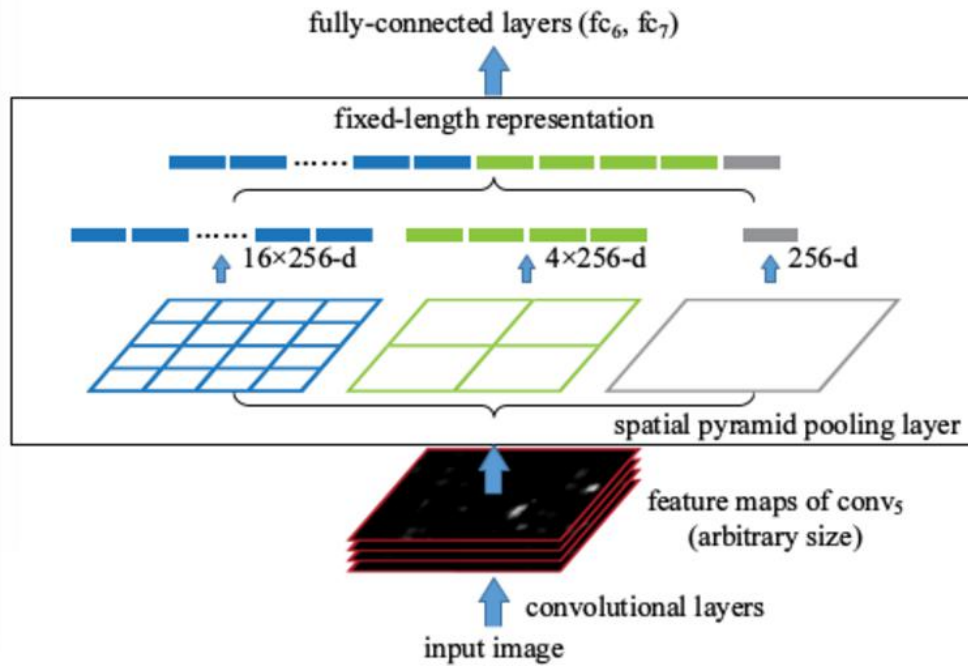


Figure 27. SPP net structure[26]

### 2.6.2.3 Fast R-CNN

Fast R-CNN[27] combines the essence of R-CNN[26] and SPP-NET[28] and introduces a multi-task loss function to train the entire network.

Also, testing has become very convenient. Trained on the Pascal VOC2007 training set, the result of the VOC2007 test was 66.F9% (mAP); if using the VOC2007+2012 training set training, the test result on VOC2007 was 70%. With VGG16, each image takes about 3s in total. Fast R-CNN[27] also has problems, such as, the use of selective search for region proposal extraction spend lots of time, which causes the differently in real-time application. Also, it does not achieve the actual end-to-end training test. In 2016, Ren et al. The Faster R-CNN[28] algorithm is proposed. The algorithm introduces the RPN (Region Proposal Network) network on the original Fast R-CNN[27] algorithm, which significantly shortens the Proposal extraction time.

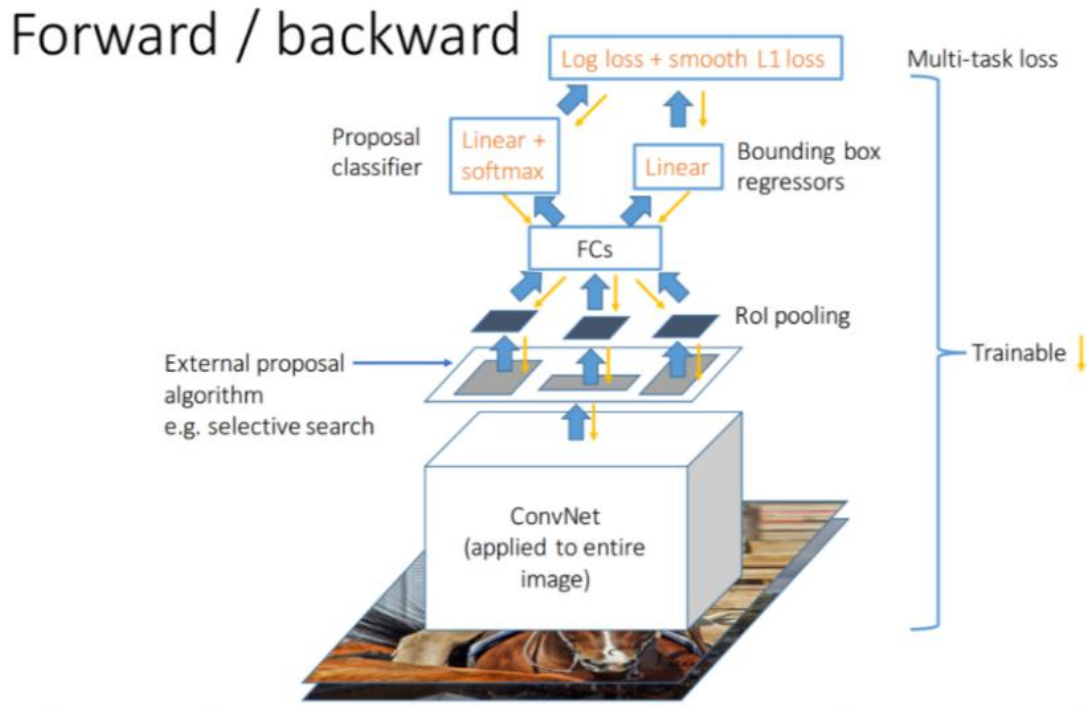


Figure 28. Fast R-CNN structure[28]

#### 2.6.2.4 Faster R-CNN

In the object detection framework of the Region Proposal series CNN classification, the quality of the Region Proposal directly affects the accuracy of the object detection task. If there is a way to extract only a few hundred or fewer high-quality pre-selected windows, that would not only speed up the algorithm but also improve the performance. The RPN network was designed to fix this situation. The core idea of RPN is that it uses a convolution neural network to generate Region Proposal directly. The method used is essentially a sliding window. The design of the RPN [69] slides it over the last convolution layer because the anchor mechanism and the border regression can get the Multi-scale, multi-width aspect ratio of the Region Proposal. For allowing the RPN network and the Fast RCNN[27] network to share the weights of the convolution layer, a four-stage training method has used to train the RPN and the Fast RCNN[27]. The structure of faster RCNN[28] shown in Figure.29.

- 1) Initialize network parameters using a pre-trained model on ImageNet to fine-tune the RPN network;
- 2) Using the RPN network[69] in 1) to extract Region Proposal and training the Fast RCNN network[27];
- 3) Reinitialize the RPN using the fine-tuning the convolution layers;
- 4) Fix the convolution layer of Fast RCNN, use the Region Proposal network to combine the Region Proposal and Fast CNN classifications, using an end-to-end system. Object detection has improved in both speed and accuracy. However, Faster



RCNN still does not achieve real-time target detection, pre-fetching Region Proposal, and then calculating the amount of calculation for each Proposal classification. The emergence of target detection methods, such as YOLO, makes real-time performance possible.

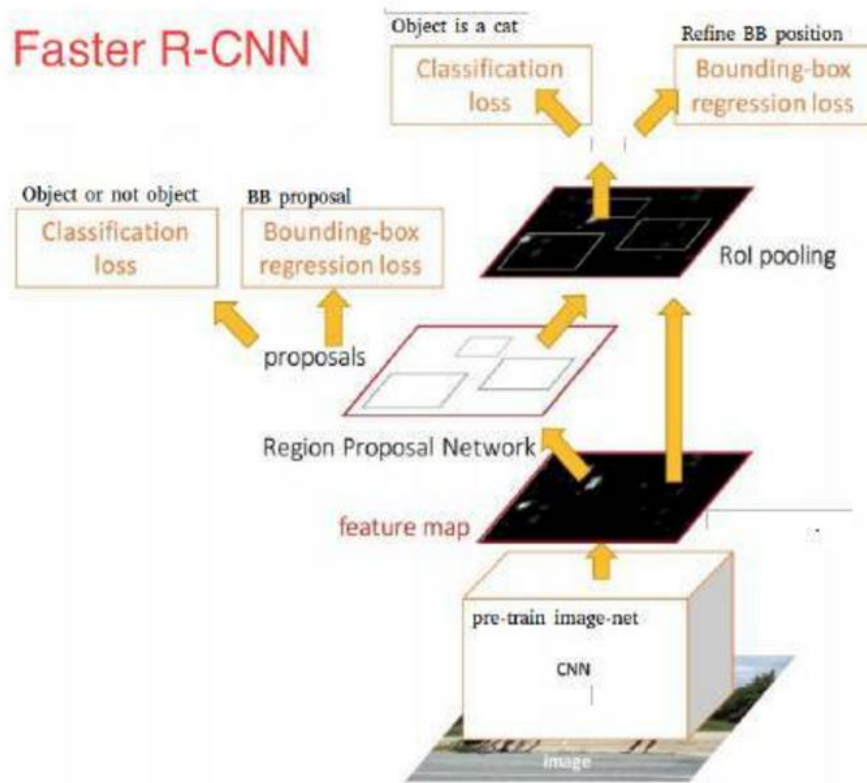


Figure 29. The structure of Faster – RCNN and RPN[29]

### 2.6.3 Deep learning target detection algorithm based on regression method

The Faster RCNN[27] method is currently the mainstream object detection method, but the speed does not meet the real-time requirements. The YOLO[32] method shows its importance. This type of approach uses the idea of regression, that is, given the input image, the target frame, and the target category at this position has directly returned at multiple positions of the image.

### 2.6.3.1 YOLO

The YOLO algorithm proposed by Redmon in 2016 is a convolution neural network that can predict multiple Box locations and categories at once.[32,75] The network design strategy of the YOLO algorithm continues the core idea of Google-Net[24], transforming the target detection task into a regression problem, which greatly speeds up the detection. However, YOLO also has issues:

The lack of the YOLO is that only 7x7 mesh regression used in the Region Proposal structure, which leads to the YOLO detection accuracy is low. In 2016, YOLO9000[76,77,78] was created based on a new training method, YOLO9000 algorithm trained by the joint training algorithm, improved the accuracy under the condition of guaranteeing the speed of the original YOLO algorithm. YOLO9000 has 9000 categories of classification information, which used the ImageNet classification data set, while object position detection used the COCO detection data set. The training network of the YOLO v2[77] algorithm adopts the down sampling method to adjust under certain circumstances dynamically. This mechanism enables the network to predict different sizes of pictures so that the speed and accuracy of detection are balanced. Table 2 shows the YOLO9000 result compared with other classic models on the VOC 2007+2012 data set. As can be seen from Table.2, object detection in the picture, YOLOv2 reached the advanced level, and the mAP on VOC 2007 was 78.6%, still higher than the average level[78]. The structure of YOLO9000 shown in Figure.30

Detection Frameworks	Train	mAP	FPS
Fast R-CNN	2007+2012	70.0	0.5
Faster R-CNN VGG-16	2007+2012	73.2	7
Faster R-CNN ResNet	2007+2012	76.4	5
YOLO	2007+2012	63.4	45
SSD300	2007+2012	74.3	46
SSD500	2007+2012	76.8	19
YOLOv2 288 × 288	2007+2012	69.0	91
YOLOv2 352 × 352	2007+2012	73.7	81
YOLOv2 416 × 416	2007+2012	76.8	67
YOLOv2 480 × 480	2007+2012	77.8	59
YOLOv2 544 × 544	2007+2012	78.6	40

Table 2 YOLO9000 result compared with other framework on VOC 2007+2012 data-set.[77]

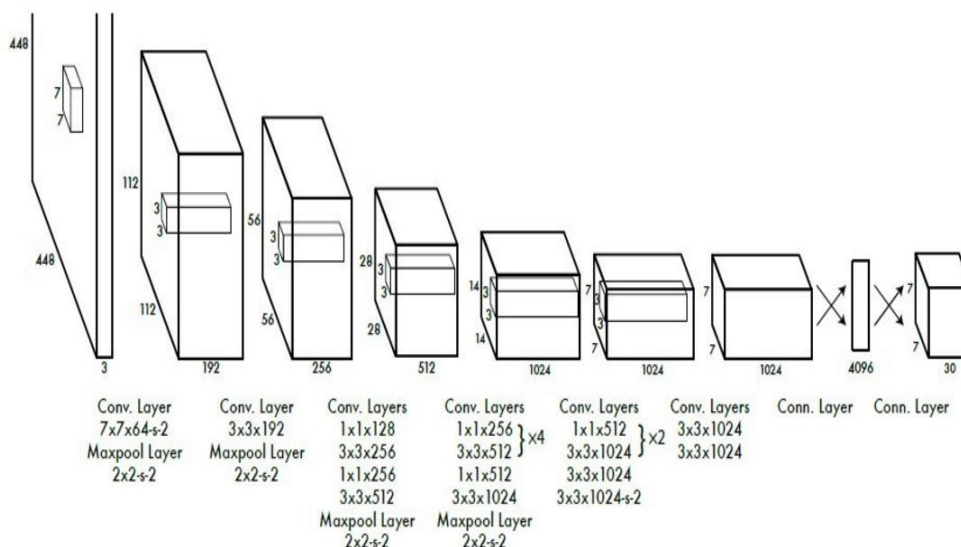


Figure 30. YOLO9000 setting and structure[77]

### 2.6.3.2 SSD

In 2016, Liu proposed the SSD[31] algorithm. The algorithm obtains the target location and category, and the YOLO-like method used.

Regression, but YOLO predicts that a location uses the characteristics of the full graph. The SSD predicts that a location uses features around the location and uses the anchor mechanism of Faster RCNN to establish the correspondence between this location and its characteristics. Unlike Faster RCNN, this anchor is on multiple feature maps, which makes use of multiple layers of features and naturally achieves multiple scales. SSD combines the regression idea in YOLO and Faster RCNN. Also, it uses the multi-scale regional features of each position, which not only maintains the fast YOLO characteristics but also guarantees the accuracy of window prediction. The SSD mAP on VOC2007 can reach 72.1%[79], and the speed reaches 58 frames/s on the GPU.[79] The compare of YOLO and SSD has shown in Figure.31. From the information of Figure.31, we know that the structure of SSD and YOLO are very similar, while SSD is more complicated than YOLO. However, in real test, the accuracy of YOLO and SSD are not very high.

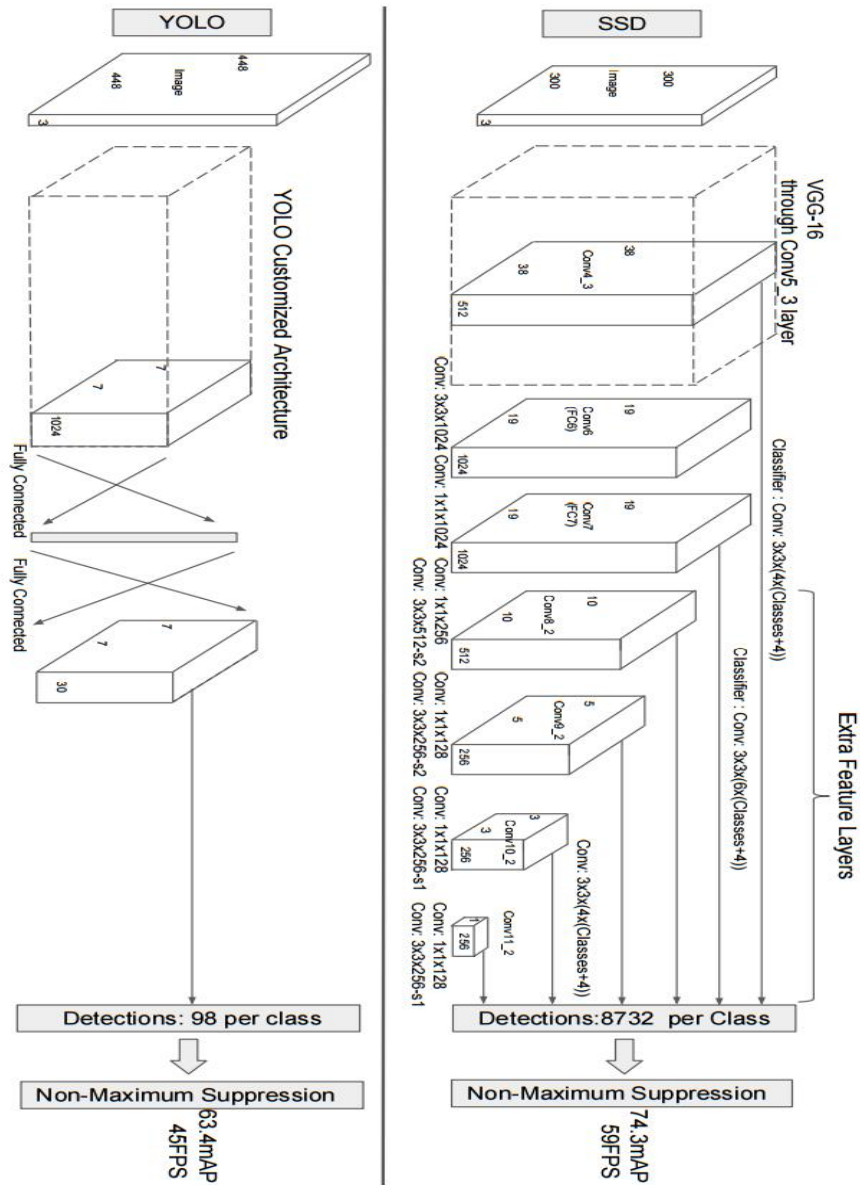


Figure 31. A comparison between two single shot detection Models: SSD and YOLO[79]

## 2.7 Application of deep learning in video expression analysis

Expression recognition is a training example that gives images or videos of predefined expression categories to predict the category labels of any unknown image or video. After entering the 21st century, with the development of artificial intelligence technology and computer technology science, the demand for human-computer

---

interaction has become stronger. Also, the research on new human-computer interaction technology has received more and more attention. We not only hope computers or robots can speak, listen, but also understand the inner human emotions. As for the expression of feelings, psychologist gave a formula: emotional expression = 7% of words + 38% of voice + 55% of facial expressions, which fully illustrates the importance of facial expressions in human communication. [80,81,82] Research on expression recognition has attracted the interest of scientists in various fields, including computer science, neurology, and psychology. If computer systems sense human emotions, they will widely use in a wide range of areas, including security, education, neurology, law, and communication technologies. [83,84] According to different research objects, expression recognition can divide into two types based on static images and video-based. The expression recognition based on static images has greatly developed in the past few decades. It has the advantages of simple, fast, and convenient feature extraction, and it can achieve a good recognition effect under certain circumstances. However, the features of static images contain very limited expression information, which is easily affected by the external environment and individual differences.[85] With the rapid development of computers, people are paying more and more attention to expression recognition based on video (image sequence).[86] Hey, because of the process of generating and disappearing expressions, video-based expression recognition can better reflect the motion process of complete expression, including more facial motion and time information. Therefore, video-based expression recognition research is more practical and more challenging.

In this section, we have developed a series of researches on video-based expression recognition, and the ultimate goal is to verify the impact of changes in expression on expression recognition. [87] For a given video containing an expression, we first extract adjacent frames for contrast comparison and use different frames to perform expression recognition. The traditional recognition method treats the entire video as a source of information, performs 3D CNN[88] or uses each frame in the video as a source of information, performs single frame recognition of the expression, and integrates the entire video recognition result. However, in human-computer interaction, if a robot needs a long time to read and judge the whole expression of the person, this will lead to a terrible experience. Also, the robot can't handle some emergencies. The detection has improved, but the traditional method also has significant learning value and can provide more thinking directions in the future identification and recognition system. The next two summaries will introduce two popular video processing methods.

---

### 2.7.1 Conv LSTM

Video has more information than pictures. Generally, the spatial scene information has referred to as the relation between frames in a video, and the target motion information carried between adjacent frames has referred to as time information. [89]. How to better explore the spatial and temporal information in video has always been a hot spot in the field of video behavior recognition research. It is well known that convolution neural networks (CNN) can automatically learn features from the multi-level processing of the visual cortex. Deep learning directly uses the original video for end-to-end training, eliminating the need to manually design features, providing an efficient feature representation for video behavior recognition. Convolution neural networks can extract spatial features. [90]. Based on the cyclic neural network, deep learning can more effectively model the timing information between video frames. Based on the ability of deep learning, researchers have made a series of progress in the field of video behavior recognition. Existing deep learning-based behavior recognition methods mainly include video frame recognition methods, LSTM network recognition methods, and three-dimensional convolution recognition methods.

Li Feifei and others first applied the method of deep learning to the field of behavior recognition. Before the LSTM network identification method, Li Feifei first adopted the single frame mode (Single Frame) mode. [91] As shown in the figure.32, a single video frame was input into the CNN network for identification, and then Li Feifei tried to mine the timing information between the video frames. The second picture is Late Fusion. At intervals, some video frames have skipped, and the features of the sampled frames have fused at the full connection layer. The information volume of the network is improved. [92]Also, the degree of the network has increased to some extent. The timing information between the video frames has extended. The third picture is Early Fusion, which inputs an adjacent video frame into the CNN, greatly retaining the timing information between adjacent frames in the video clip. The fourth picture shows the Slow Fusion method. [92,93]This fusion method draws on the advantages of the first two fusion methods. Each layer of convolution pays attention to mining timing information between video frames. The experimental results show that the slow fusion method can make the best use of the spatial and temporal information,[93] but the experimental results are worse than the traditional manual model.

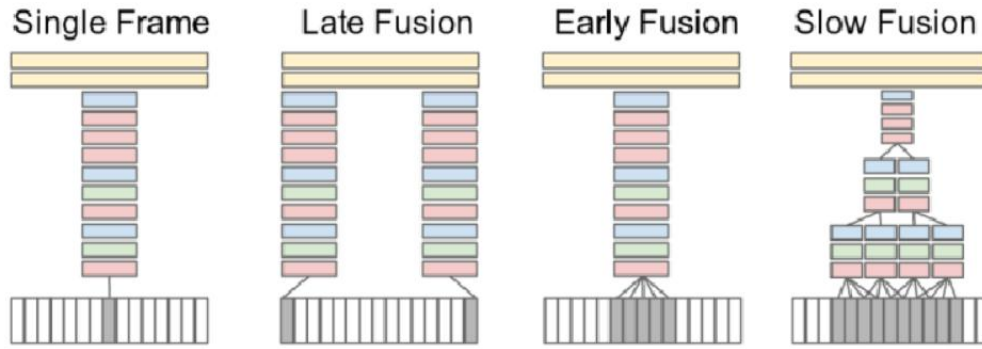


Figure 32. Four methods to extract temporal feature from video[94]

In addition to the single frame recognition method that described, Simonyan[94] also proposed to develop timing information to form another information chain for learning, that is, the dual-stream method. Simonyan[94] first proposed the two-channel identification method in 2014. The basic idea of Simonyan[94] is to calculate the dense optical flow for each adjacent two frames in the video sequence and obtain the optical flow information of the video. Then, using the video decoded frame and the optical stream extracted by the frame, the two CNN recognition models are respectively trained.

Finally, the recognition results of the two-way CNN model based on the stacked L2 normalized softmax score and multi-class linear SVM has used. Although the two-channel recognition method is similar to the traditional manual feature method, it is greatly improved the ability of time features extraction. Also, the two-channel recognition method proves that the deep learning method can replace the manual feature method. [95] Subsequent researchers have made many improvements on this basis. Other researchers also explore some new models such as multiple options for fusion in different convolution layers, Temporal Segment Networks (TSN)[96]. What's more, the sparse sampling method was used to solve the two problems of training network models under limited samples and capturing long-term features of the video.

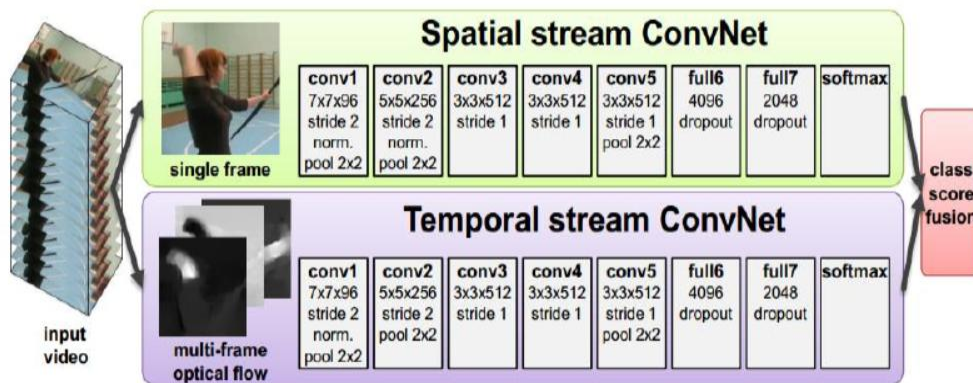


Figure 33. Spatial and temporal stream Conv-Net[95]

---

The video content is rich in timing information, and the convolution neural network cannot fully utilize the time domain information of the video. Although the single frame identification method and the dual-channel identification method use the video timing information, the video with a large interval and period still need deeper timing information. The output of the LSTM determined by the current input and the output of the previous time. It can represent the sequence information of the sequence that has widely used in processing timing problems, such as image description, speech recognition, document digest.

Convolution neural networks (Google-Net[26], VGG[67].) are generally used to extract the spatial features of the image, then spatial features are extracted by the LSTM network. Also, timing information to identify behavior. Donahue proposed a video behavior recognition architecture that uses CNN and LSTM together.[96] First, the video frame sequence sent to CNN, and the extracted sequence space feature used as the LSTM input. The average value of each time of the LSTM unit took as the prediction behavior. The final output. Sharma introduced an attention mechanism based on LSTM. By dividing the convolution feature map into  $k \times k$  regions and scoring these regions, the model captures the key parts of learning video motion and helps to learn the video. Refined features. In actual use, CNN can be used to extract the convolution characteristics of the image. [97].As shown in Figure.34, the LSTM cells of the same layer represent timing extensions and can also be modeled using multiple layers of LSTM cells stacked. The top of the network is the output of the LSTM at different times. Generally, the output of the last moment selected for prediction. LSTM is used to extract temporal features. Some new time series cyclic networks (such as GRU, Bi-LSTM) developed by LSTM can also replace LSTM in this structure. However, the overall network structure of Conv-LSTM leads to many model parameters, and it is difficult to train and test. Timeliness is also one of the main problems.



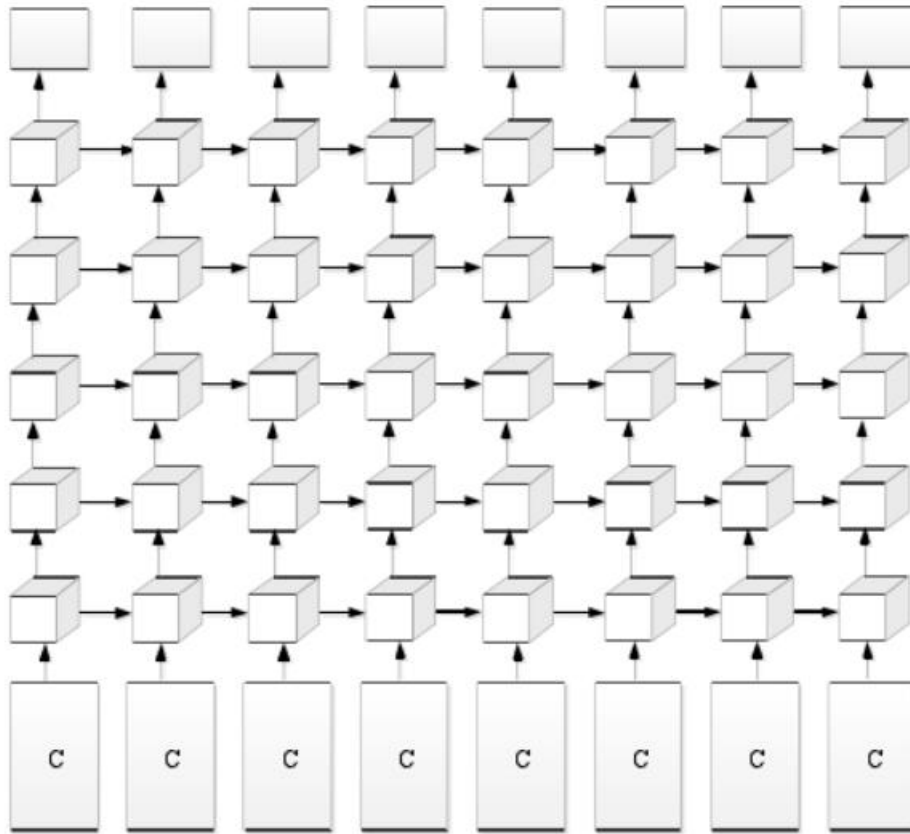


Figure 34. Conv LSTM structure[95]

### 2.7.2 3D CNN

The previous section introduced the conv-LSTM structure to analysis video, but in reality, video recognition considered as two issues (frame feature extraction, frame space association). [88].The traditional convolution can effectively extract the spatial features of the image, but the video recognition, which operated as independent images, will lose a lot of timing information. Because of the drawback of two-dimensional convolution, researchers created the three-dimensional convolution. This method extracts features from the spatial and temporal dimensions. 3D convolution is better than 2D convolution on some data sets. in other words, 3D convolution can improve the ability of video analysis.

3D CNN, as its name suggests, is convolution or feature extraction in three dimensions. The image has two dimensions, which are length and width. The stacking of different time frames in the video forms the space-time dimension, which is the third dimension. At the time of three-dimensional convolution, the window for feature extraction also becomes a three-dimensional block window. [99]. This way of convolution, when proposed, requires depth feature extraction to extract temporal

features between frames. However, with the data changes from 2D to 3D, and the dimension of the sliding window increases, the parameters of the entire network are greatly increased. Also, the training of 3D CNN will be more difficult and spent more time. Figure.35 shows the diagram of 3D CNN and the structural setup of the entire model.

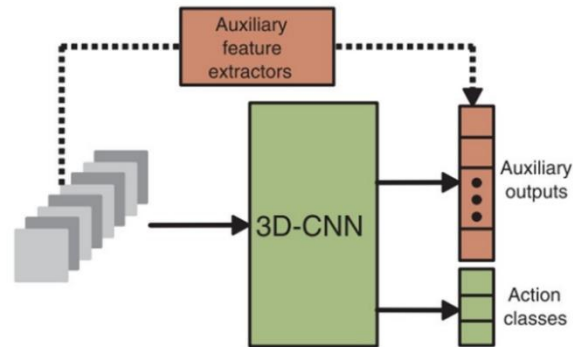


Figure 35. 3D CNN structure 1[100]

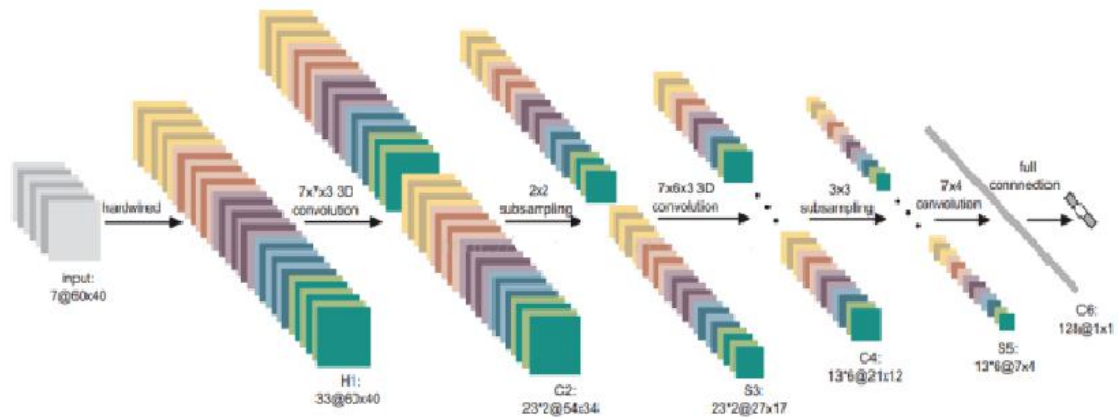


Figure. 35 3D CNN structure 2[100]

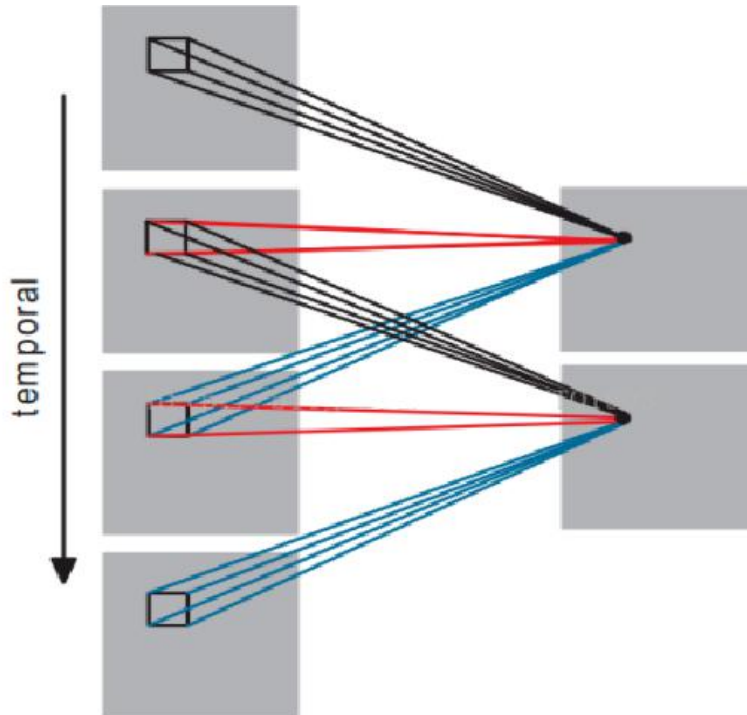


Figure 36. 3D CNN hardwired layer[100]

In Figure.36, convolution extracts the information of two frames, which has spatial and temporal information in the obtained features. The first layer in 3D CNN also proposes a new layer of extracted layers, hardwired layer. This layer is different from the convolution layer of custom extraction features. This layer extracts five channels of information per frame (gray, gradient-x, gradient-y, opt-flow-x, and opt-flow-y). The first three channels can directly obtain from each frame, and the subsequent optical stream (x, y) needs the information of two frames. [101]. This layer is the artificial help network to extract some valuable information. However, the network model after development also directly applies 3d convolution to extract features, such as lip-net, to recognize speech text through the change of the lips of the video. As shown in the Figure.37, Lip-net[102] does not set the first layer of artificial extraction feature layers, all of which are automatically extracted by convolution.

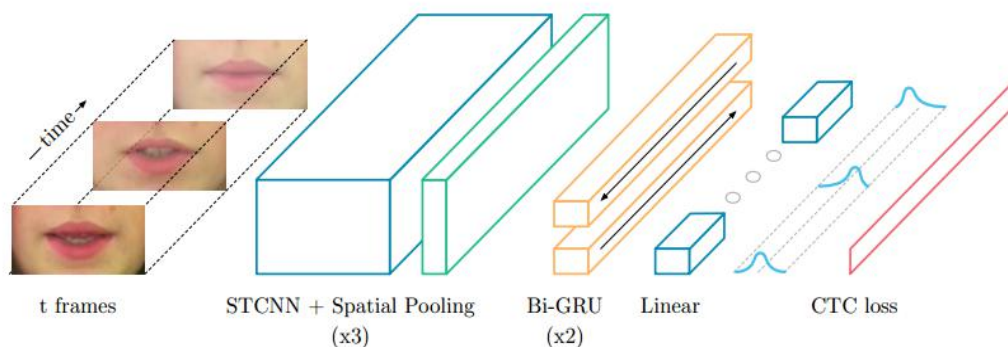


Figure 37. Lip-net structure[102]

Although the 3D CNN[88] structure is not too deep, the amount of parameters is huge. The author of the article also published results compared with other speech recognition networks. As shown in the table.3, it can be seen that lip-net shows its superiority in terms of correctness. Moreover, the accuracy of 95.2% in the larger data set has already had certain practical value. [103]In the actual comparison results, the state of the art result has reached.

Method	Dataset	Size	Output	Accuracy
Fu et al. (2008)	AVICAR	851	Digits	37.9%
Hu et al. (2016)	AVLetter	78	Alphabet	64.6%
Papandreou et al. (2009)	CUAVE	1800	Digits	83.0%
Chung & Zisserman (2016a)	OuluVS1	200	Phrases	91.4%
Chung & Zisserman (2016b)	OuluVS2	520	Phrases	94.1%
Chung & Zisserman (2016a)	BBC TV	> 400000	Words	65.4%
Gergen et al. (2016)	GRID	29700	Words*	86.4%
LipNet	GRID	28775	<b>Sentences</b>	<b>95.2%</b>

Table 3. Result of lip-net[102]

### 2.7.3 Summary

This chapter introduces the development of algorithms for video analysis and the more popular video analysis algorithms. According to the existing video analysis method, includes single-frame analysis, sequence-based analysis, and 3d CNN, the critical point of the video analysis is to extract and use spatial and temporal information. Due to the algorithm structure, we can not know what the spatial and temporal information is or whether it has fully extracted. However, from the obtained algorithm information and the result, we found that the spatial and temporal information has a positive effect on the final result. In the final chapter, a new pre-processing method has been proposed to extract spatial and temporal information in the video.

## 2.8 Face Analysis

In the single-frame expression recognition, the data set is an image, and the proportion of the face in the picture is huge, reaching 80-90%.[103,104] In reality, video expression recognition usually has a small face or side face. Before performing video expression recognition, it is necessary to perform reasonable pre-processing to remove interference factors other than the face. Face detection in pictures is a feasible and necessary method.

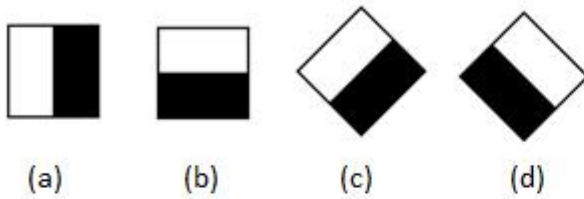
### 2.8.1 Face Detection Analysis Model Based on Ada-Boost Algorithm

The first step of the face recognition system, face detection algorithm can accurately detect the face and the location of the face. The size of the face directly affects the effect and accuracy of subsequent face feature extraction. At present, the research methods of face detection are endless, but most of them are subject to detection accuracy and detection speed. Until Viola and Jones[2] proposed the face detection method based on the Ada-Boost algorithm, face detection technology is efficient. The method mainly includes three parts: Haar-like[106] feature extraction, Ada-Boost algorithm, and cascade classifier[105].

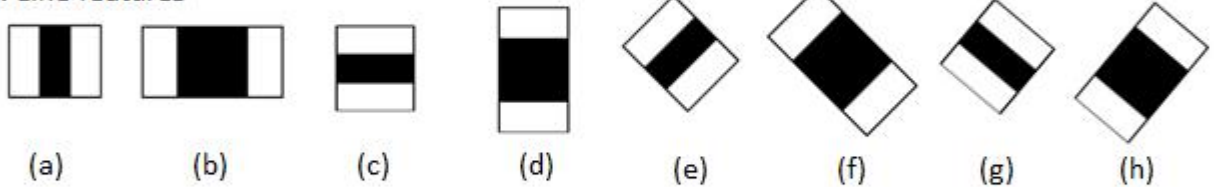
#### Haar-like feature extraction

Compared with direct pixel processing, simple feature extraction can encode images in a specific area, thus speeding up the running speed of the later algorithm. The face detection algorithm based on Ada-Boost uses rectangular features: matrix features are sensitive to basic graphic structures such as edges and line segments. [106]. The face, nose, and mouth of the face are generally darker than other parts so that they can roughly represent by rectangular features. The rectangular feature in Viola's method called Haar-like features, which initially was only five. After that, Lienhart improved it as needed, adding a 45-degree angle[106]. Figure.38 shows the 14 Haar-like feature currently in use.

## 1. Edge features



## 2. Line features



## 3. Center-surround features

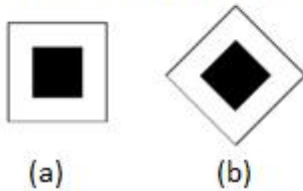


Figure 38. Haar-like feature model[107]

As can be seen from the figure, the Haar-like feature rectangle is composed of two or more adjacent squares of equal size.[107,109].The square divided into white and black regions. The corresponding feature value of a template defined as the sum of the pixels in the white region minus the sum of the pixels in the black area. Viola's paper[2] also found the number of features of each membrane[110], as table.4 shown:

Feature Type	w/h	X/Y	#
1a; 1b	2/1; 1/2	12/24; 24/12	43,200
1c; 1d	2/1; 1/2	8/8	8,464
2a; 2c	3/1; 1/3	8/24; 24/8	27,600
2b; 2d	4/1; 1/4	6/24; 24/6	20,736
2e; 2g	3/1; 1/3	6/6	4,356
2f; 2h	4/1; 1/4	4/4	3,600
3a	3/3	8/8	8,464
3b	3/3	3/3	1,521
Sum			117,941

Table 4 .Number of features of each membrane[107]

As shown in Table.4, the number of features is too many to fast calculations. Therefore, in 2004, Viola proposed a Haar-like feature extraction method based on integral graphs. For upright rectangular features, the defined integral map  $SAT(x,y)$  represents the sum of all pixels in the rectangular region from the upper left corner to the point  $(x,y)$  in the image window[111,112,113]. The formula expresses in equation 2.18:

$$SAT(x,y) = \sum_{x' \leq x, y' \leq y} I(x',y') \quad (2.18)$$

The calculations of  $SAT(x,y)$  can easily calculate without having to re-count from the top left corner each time.[111,112,113] The data processing progress is shown in Figure.39 and calculated as equation 2.19 and 2.20:

$$SAT(x,y) = SAT(x,y-1) + SAT(x-1,y) + I(x,y) - SAT(x-1,y-1) \quad (2.19)$$

$$SAT(x-1,y) = SAT(x,y-1) = SAT(x-1,y-1) = 0 \quad (2.20)$$

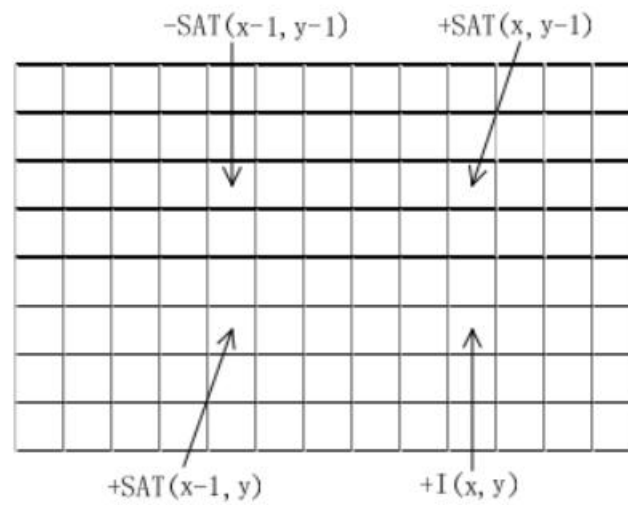


Figure 39 . Calculation diagram[111]

Overall, the successful introduction of the integral graph greatly improves the calculation speed of the rectangular feature, which enhances the practicability of the overall algorithm.[113,114] The face recognition method is very mature and has strong practicality.

### 2.8.2 Face detection ---MTCNN

The previous section introduced a classic Viola and Johns face extraction feature. This section will present a multi-face recognition model, MTCNN[34], which is also a perfect face recognition model. MTCNN is a face detection deep learning model of multi-tasking cascade CNN[34], Face regression, and facial keypoint detection algorithm are considered in this model. The cascaded CNN network structure includes P-Net[34], R-Net[34], and O-Net[34]. The overall flow chart shown in Figure.40.

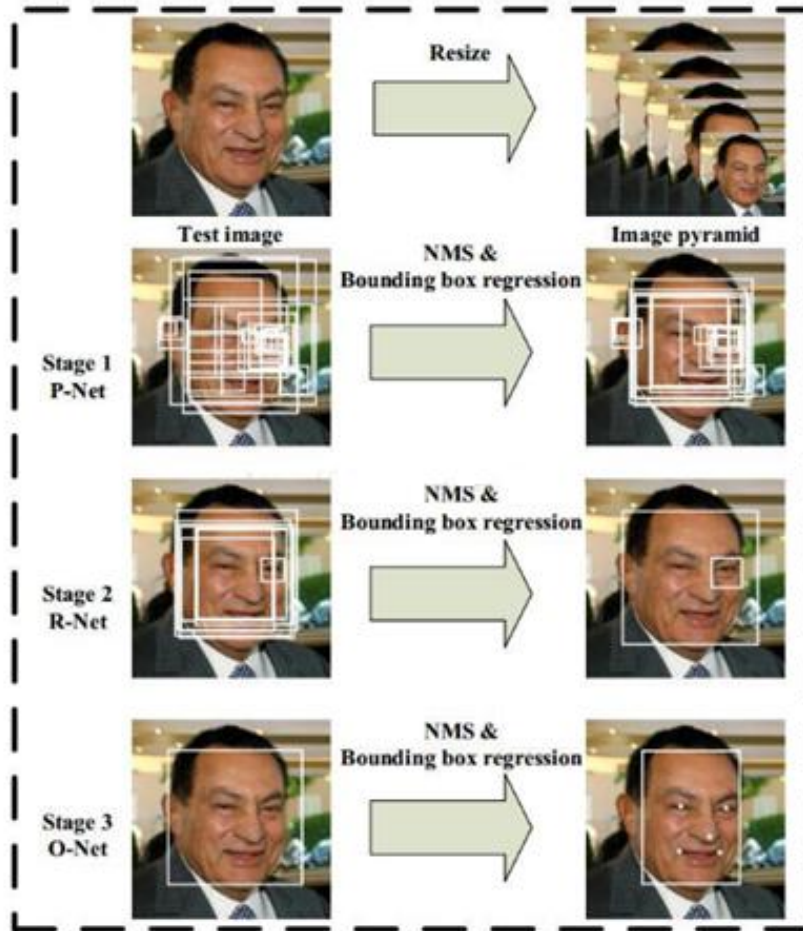


Figure 40. MTCNN Flowchart[34]

Scaling the image and constructing the image pyramid before feeding the image into the network, and different size images help the subsequent network extract different level features. Training in various dimensions to make the system learn large scales face and small scale face also have a specific regression effect.

P-Net (Propose network) is used to obtain the regression vector of the face window and the corresponding candidate frame. Then, the generated face candidate frame merged by non-maximum suppression (NMS)[115]. The setting of R-net is similar to that of P-net. When using R-net again, NMS[115,116] will be used to filter through the P-net candidate box. The last o-net is to use more stringent regression criteria (5 face key points). The entire network is a cascading structure, screening through the upper-level goal until the entirely returned.[116] During the training process, there are multiple tasks that regression at the same time, include, face key-points location, boundary regression box, and face determination.

P-Net, R-net, and O-nets structures shown in Figure.41-43



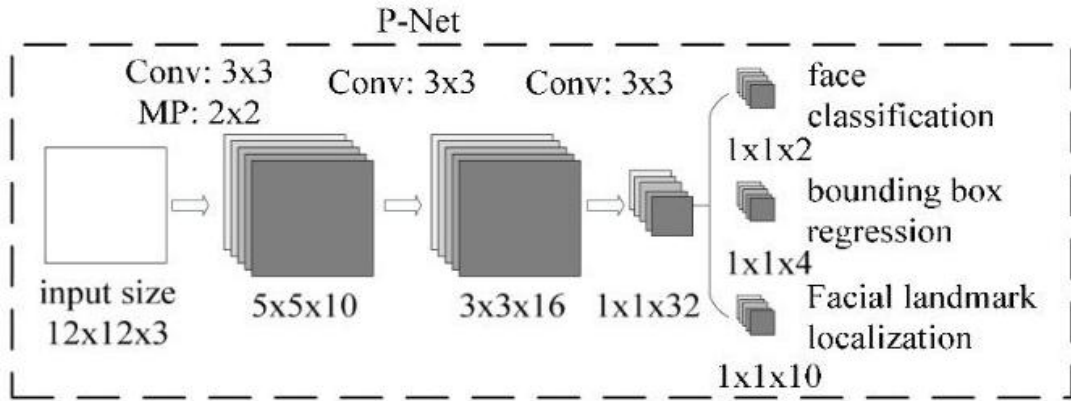


Figure 41. P-net structure[34]

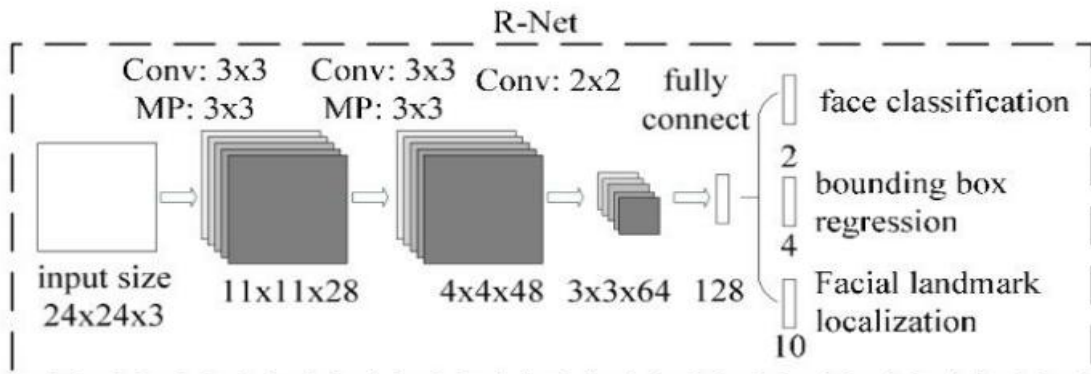


Figure 42. R-net structure[34]

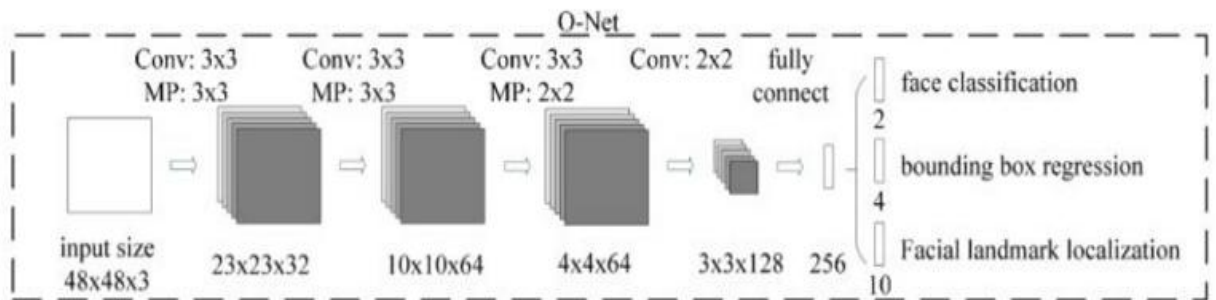


Figure 43. O-net structure[34]

The emergence of MTCNN[34] solved the multiple-faces problem and some factors that interfered face recognition in the natural environment, such as illumination and side faces. These problems are solved, which rely on the ability of deep CNN learning.[116]

### 2.8.3 Summary

This chapter briefly introduces the face feature extraction project and the classic algorithm MTCNN for face detection. As a preprocessing of real-time human expression detection, the face detection algorithm plays a vital role in video expression recognition. In the analysis of MTCNN, the cascading training method and the multi-task simultaneous training method have achieved excellent results. A similar cascading discriminant network can also implement in future video expression recognition.

## 2.9 Model training

As a kind of machine learning, deep learning divided into supervised learning, unsupervised learning, and semi-supervised learning. The classic CNN model and the object detection model introduced above all use the method of supervised learning to train the model. Before the introduction of the model training, some terminology and knowledge of training need to introduce.

### 2.9.1 Label ---one hot

In supervised learning, the setting of labels is crucial. Models that have not trained need to know the label corresponding to each training sample and use the label as a basis for the back propagation algorithm. One-hot type labels often used for training in classification networks.[118,119,120,121] Figure.44 shows a simple one-hot format label for handwritten numbers. Among the ten categories in the digital recognition, only six corresponding positions have data of 1, and the rest are 0. When the deep learning network processes the one-hot label show in the figure.44, the sample image is considered to be 6, rather than other categories. In the back propagation algorithm, the output of the model continually approaches the label data in the training process progresses, resulting in the highest score for the tagged category. In many frameworks (Tensorflow[122], Pytorch[123], Karse[124]), you can directly convert data labels into one-hot form.[122,123,124]

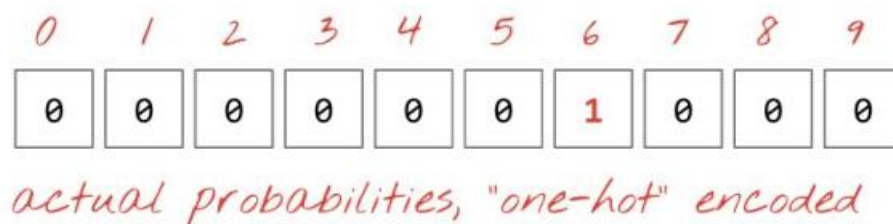


Figure 44. One-hot label[118]

## 2.9.2 Loss function and optimism

Before the training began, the internal data of the model was chaotic and irregular. Through training, the model parameters gradually changed, making the whole model have a certain accuracy. The loss function determines the change in model parameters. When a sample data tested in an untrained model, the difference between the sample data and the label data can be found, and the parameters in the model will change to decrease the difference. The purpose of the loss function is to define the difference and feed the difference to the deep neural network optimizer[125,126]. The training process of the neural network, that is, the process of making the difference close to zero. Using the one-hot type label makes the final difference not a single data, but an array or column matrix. The built-in loss functions in tensor-flow are described later in this section. The mathematical basis of some functions needs to explain before introducing the tensor-flow built-in functions. The Loss function usually has two major classes, the mean-variance, and the cross-entropy.[127]

### MSE loss function

The mean square error is also a function abbreviated as MSE. The value of MSE reflects overall data stability or the similarity between the two sets of data. The MSE equation shown as 2.21:

$$\frac{1}{m} \sum_{i=1}^m (y_i - \hat{y}_i)^2 \quad (2.21)$$

Where  $y_i$  is an array, in deep learning,  $y_i$  represents the difference between the value actually calculated by the network and the value in the label. As long as the mean square error approaches zero, it means that the calculated value approaches the label, which also means that the effect of the model is better. MSE performs well in linear regression and binary classification, but perform not that good in multi-classification tasks.[128] Because when the overall value approaches 0, it does not mean that each value and the label approach 0 at the same time. Therefore, it has a good effect only in the two classifications and the linear classification of this particular classification task.

The Mean Square Variance series also has some variant functions, and their use of data features is basically the same. Such as, RMSE[129], MAE[130], R-Squared[131]. The equation of these function shown as equation 2.22-2.24.

$$\sqrt{\frac{1}{m} \sum_{i=1}^m (y_i - \hat{y}_i)^2} \quad (2.22)$$

$$\frac{1}{m} \sum_{i=1}^m |(y_i - \hat{y}_i)| \quad (2.23)$$

$$\frac{\frac{1}{m} \sum_{i=1}^m (y_i - \hat{y}_i)^2}{\frac{1}{m} \sum_{i=1}^m (\hat{y}_i - y_i)^2} = 1 - R^2 \quad (2.24)$$

## Cross entropy

Both the cross entropy and the MSE series[128] are loss functions. The difference is that cross entropy solves multi-classification tasks and nonlinear classification tasks better. Before introducing cross entropy, the concept of entropy must be introduced. Entropy first appeared in chemical area to describe the level of chaos of molecules or atoms, and was later introduced into the electronic information discipline.[132] Suppose there are multiple events, each of which has a probability, represented by the symbol  $p$ . The amount of information for an event, capital  $I$ , can be found using the probability of an event. The equation 2.25 shows the relation between possibility of event and the amount of information, where  $i$  is the number of events.[133]

$$I = -\log(p_i) \quad (2.25)$$

Now there is a definition of the amount of information, and entropy can be interpreted as the sum of the expected quantities of information for all events.[133,134] The value of entropy can be found using equation 2.26.

$$H(X) = \sum_{i=1}^n p(x_i) * \log(p(x_i)) \quad (2.26)$$

$n$  represents all events, and  $H$  is the information entropy of the event set.

The derivation of cross entropy is obtained from the formula of relative entropy. [135].The following is the derivation process of cross entropy. Relative entropy is also known as KL divergence.[135] If there are two separate probability distributions  $p(x)$  and  $Q(x)$  for the same random variable  $x$ , we can use the KL divergence (Kullback-Leibler (KL) divergence) to measure the Difference between the two distributions. The wiki's interpretation[136] of relative entropy is “In the context of machine learning, DKL (P||Q) is often called the information gain achieves if P is used instead of Q”.

For example, in the classification task, we often give the image a classification  $p$ , that is, to determine that the image belongs to a certain class, and does not belong to other classes. In reality, we may make a fuzzy judgment on a certain picture, like, 70% (Q)

determines that the picture belongs to a certain class. In the training process of machine learning and deep learning, the model analyzes the image, extracts the feature, and gives a judgment result.[137] The result is not to determine that the image must belong to a certain class, but a probability return. If P was used to describe the sample, then it is perfect. Using Q to describe the sample is not so perfect, although it can be roughly described.[137] The amount of information is insufficient, and additional "information increments" are needed to achieve the same perfect description as P.

The equation for calculating relative entropy is shown in equation 2.27:

$$DKL (P||Q) = \sum_{i=1}^n p(x_i) * \log \left( \frac{p(x_i)}{q(x_i)} \right) \quad (2.27)$$

Decompose this formula further (2.28-2.30):

$$DKL (P||Q) = \sum_{i=1}^n p(x_i) * \log(p(x_i)) - \sum_{i=1}^n p(x_i) * \log(q(x_i)) \quad (2.28)$$

$$= H(p(x)) + \left[ - \sum_{i=1}^n p(x_i) * \log(q(x_i)) \right] \quad (2.29)$$

$$H(p,q) = - \sum_{i=1}^n p(x_i) * \log(q(x_i)) \quad (2.30)$$

$H$ , is the definition of entropy introduced in the equation 2.26, and the other part in equation 2.27 is called cross-entropy,[137,138] which shown as  $H(p,q)$  in equation 2.30. When the model is untrained, the returned results will deviate significantly from the actual label given. Relative entropy is used to compare the results of two labels or models. When the value of relative entropy becomes smaller, the probability distribution given by the model is closer to the label. Since the first part of the relative entropy does not change for fixed events, the neural network usually implements training of the model by reducing the cross-entropy. The cross-entropy loss function is better than the traditional linear MSE loss function in the multi-label classification task.[137]

There are four built-in functions for cross entropy under the tensor-flow framework:

Function\_1:

```
tf.nn.sigmoid_cross_entropy_with_logits(_sentinel=None, labels=None, logits=None,
name=None)[137]
```

Regularization is performed by sigmoid before calculating the cross entropy. Applicable to multiple classification tasks that are independent of each category.

---

There are multiple targets in one image. The output loss is the sum of all loss.

Function\_2:

```
tf.nn.softmax_cross_entropy_with_logits(_sentinel=None, labels=None, logits=None,
dim=-1, name=None)[137]
```

Softmax regularization before calculating cross entropy, suitable for single classification tasks, one image belongs to only one class.[137]

Function\_3:

```
tf.nn.sparse_softmax_cross_entropy_with_logits(_sentinel=None, labels=None, logits=
None, name=None)[137]
```

Function\_4:

```
tf.nn.weighted_cross_entropy_with_logits(labels, logits, pos_weight, name=None)[137]
```

The cross entropy loss function that no longer averages weight when passing through the softmax layer.[137]

Which loss function to use requires a full understanding of the classification problem, and a reasonable loss function is vital for the convergence of the entire model. An unreasonable loss function may not be able to converge or have a weak training effect. This section describes the loss function in the classification problem. In other fields (such as the face comparison system), there is also a ternary method and a loss function for measuring features relative distance. However, the goal is the same, which is to decrease the difference between the probability distribution of model prediction and given label.

### 2.9.3 Under-fitting and over-fitting

Before training a model, two phenomena in the training process need to be paid attention to, which are over-fitting and under-fitting. Each neural network has its structure, and different structures have different convergence capabilities. Each model also has its limits. There may be cases where the model training results are not good during the training process. In the case of determining that the database is correct and clean, there are two states in the model training process, over-fitting, and under-fitting. Under-fitting means that the training process is not sufficient, and the model stops the iterative process when it has not reached its limit. The parameters in the model did not converge to the best state, which caused the model to perform poorly at the end. The under-fitting situation can be checked by observing the accuracy of the test set.[138]

Similarly, excessive training will not make the model perform better. If the model is well-trained, continuing to train the model will lead to an over-fitting model. The over-fitting model only has a good performance on the training set. Whether it is

---

under-fitting or over-fitting is the training failure caused by improper training. Monitoring the accuracy rate on the test set and the return value of the loss function can avoid improper training. The model training is a process of data fitting. In order to make model to learn certain rules, effectively training is especially necessary.

#### 2.9.4 Training Sets, Test Sets, and Validation Sets

In the deep learning training process, the data sets usually divided into a training set, a test set, and a verification set. The training set is the data that participates in the adjustment of the model parameters. One batch of training set data completes one iteration process through the network model and the loss function. The test set and the verification set assume a test and verification function during the training process. The verification set is used to test the accuracy of the model in each model iteration. The performance of the model can be checked by verifying the accuracy of the validation set. If the accuracy of the verification set is gradually increasing, then the training is correctly fitted. While, if the accuracy of the validation set is gradually decreasing, the model ' s structure might needs change. If the accuracy of the validation set does not change, the model may not converge due to the database unclear classification database or the impact of dirty data.

After training for some time, the model will test on the test set. The test on the test set is for over-fitting detection. In the case of a small loss value, further training will make the model tend to overfit. In order to check the model situation, people usually monitored the test set accuracy. If the test set accuracy tends to be flat, the entire model regarded that reaches the convergence limit. If the test set accuracy begins to fall again after the flatness, the whole model training is over-fitting, making test set behave poorly.

There are corresponding functions in the TensorFlow framework to directly divide the database into training sets, test sets, and validation sets. After training for some time, the model will test on the test set. The test on the test set is for over-fitting detection. In the case of a small loss value, further training will make the model tend to overfit. In order to check the model situation, people usually monitored the test set accuracy. If the test set accuracy tends to be flat, the entire model regarded that reaches the convergence limit. If the test set accuracy begins to fall again after the flatness, the whole model training is over-fitting, making test set behave poorly.

There are corresponding functions in the TensorFlow framework to directly divide the database into training sets, test sets, and validation sets.

### 2.9.5 Step and epoch

Due to the increase in computing power, the neural network has once again ushered in a research boom. The GPU usually provides computing power. However, the memory of the GPU is limited. Ideally, the neural network can learn the entire database images at the same time, but since the GPU memory cannot accommodate all the pictures. So the dataset will divide into batches and then feed into the framework to learn. Batch size refers to the number of images sent to the GPU in a batch. The neural network learns a batch of pictures called a step. [138] When all pictures in the train set learned by the model, the process of which called an epoch. The training process is a data convergence process, and also can understand as the process of finding the optimal solution. The batch calculation model makes the whole training process change from finding the optimal global solution to finding the optimal local solution. After enough training, the optimal local solution also approximates the optimal global solution.

However, in actual training, we don't know when the optimal solution will obtain, so the batch size is a significant factor in the final model.

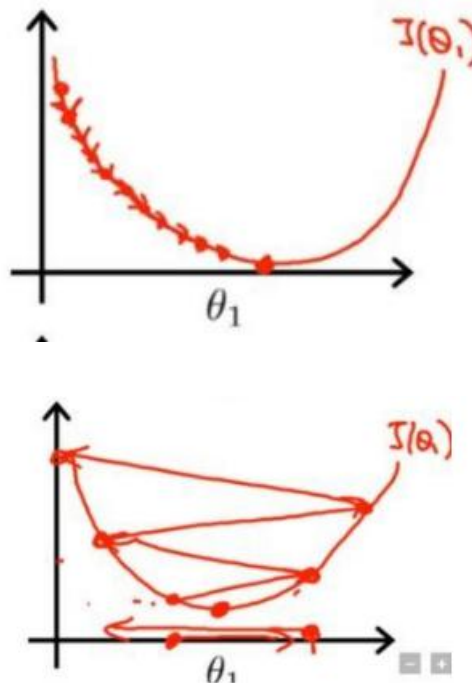


Figure 45. (a) learning ratio is too small (b) learning ratio is too large[136]

The training process of the model can describe as achieving the optimal problem.[136] In Fig. 45(a) and (b), it is assumed that the optimization problem is to reach the lowest point of the quadratic function. Each iteration in the gradient descent algorithm is considered to be an action close to the lowest point. In Figure 45(a), the step size is too small and requires many iterations to reach the lowest point. In Fig. 45(b), the step size of each iteration is too long, which causes the update direction is wrong. The iteration direction frequently changes, which also needs a long time to reach the



lowest point. In summary, the learning rate setting determines the training quality. Fig. 46 shows more intuitively the changes in the loss function with different learning rate settings.

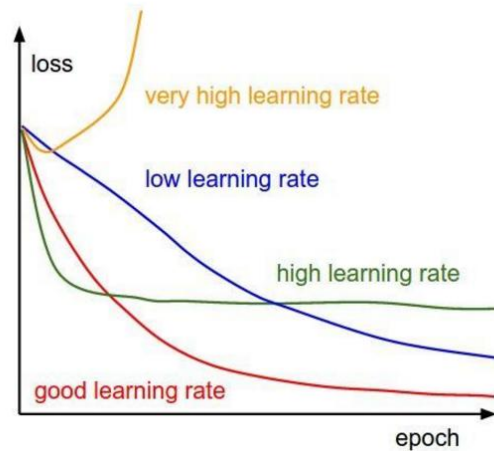


Figure 46. Different learning behaviors[136]

When the learning rate is too large, the loss may not decrease, but instead rises. When the learning rate is set too large, the value of loss decreases rapidly during training, but it will tend to be constant after a few epochs. At this time, the learning rate is too large to learn more precise parameters. When the learning rate too small, the overall loss will drop slowly, but it will eventually converge. A proper learning rate setting can quickly adjust parameters at the beginning of training and slowly adjust parameters at a later stage.[138]

Since the learning rate is difficult to set when training the model, there has a learning rate decay setting. The learning rate is continuously attenuated during training to achieve the best training effect.

## 2.10 Transfer learning

With the application of deep learning technology, a general problem that hinders the promotion of this technology is also becoming prominent: the massive data necessary to train the model is difficult to obtain. Researches are more base on the existing models' parameters, the reason of which is that the transfer learning method can reduce the data burden of the training tasks, and the risk of over-fitting. The technique of transfer learning is very popular in the actual training network process. Transfer learning also has certain limitations and restrictions when using. This section introduces several concepts of transfer learning and how to use transfer learning. [139].Figure.47 shows the parameter quantities of the depth models of the classical networks in different fields and the sample sizes used by the training models.

	VGGNet	DeepVideo	GNMT
Used For	Identifying Image Category	Identifying Video Category	Translation
Input	Image 	Video 	English Text 
Output	1000 Categories	47 Categories	French Text
Parameters	140M	~100M	380M
Data Size	1.2M Images with assigned Category	1.1M Videos with assigned Category	6M Sentence Pairs, 340M Words
Dataset	ILSVRC-2012	Sports-1M	WMT'14

Figure 47. Parameter of different model[138]

As can be seen from Fig. 47, the better-performing deep network model requires a large number of samples to train, and the training parameters are also very large. If only minor changes a little, preparing and training the new network again waste plenty of time and energy.[137,138,139] while transfer learning can reduce or avoid this waste.

Take the famous image-net models as an example. When training the image-net models, there are 1000 categories, a huge database of more than one million images. Each year's competition has new models and higher classification accuracy. However, if just one new category added to the image-net database, then retrain the model with random parameters will take a lot of time and effort. At this point, transfer learning can help deal with this situation. The old image-net model can regard as having the ability to classify 1000 classes but cannot distinguish the new category. So, learn the new category and combine the ability with the old model would be the best way. That is the transfer learning, the ability of the old model transfer to the new task.

In summary, transfer learning uses the well-trained model parameters to initialize a new model, which decreases the training period. Also, prevent the model from over-fitting and insufficient sample size. Figure.48 shows the comparison between transfer learning and traditional learning. It shows that the transfer of knowledge makes the model training more concise and fast.

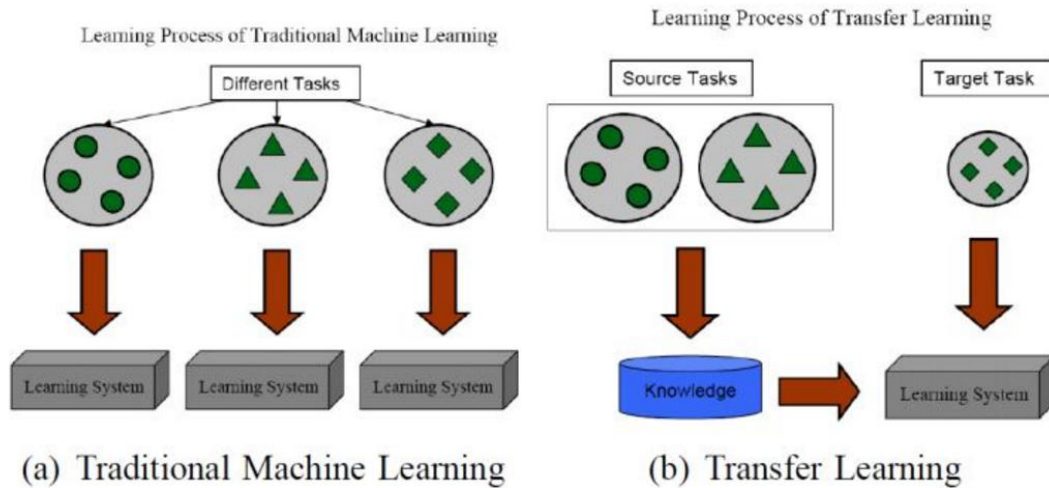


Figure 48. Difference of Traditional machine learning and transfer learning[139]

In transfer learning, the trained model is called the source domain, and the new model is called the target domain. Figure 49 shows a model after transfer learning. In Figure. 49, some parameters of the pertained model have transferred to another network. [137].The Nano-net is a network structure constructed according to the needs of its tasks. The Nano-net parameters are initially random. The final fully connected layer will change to the shape to fit the new mission.

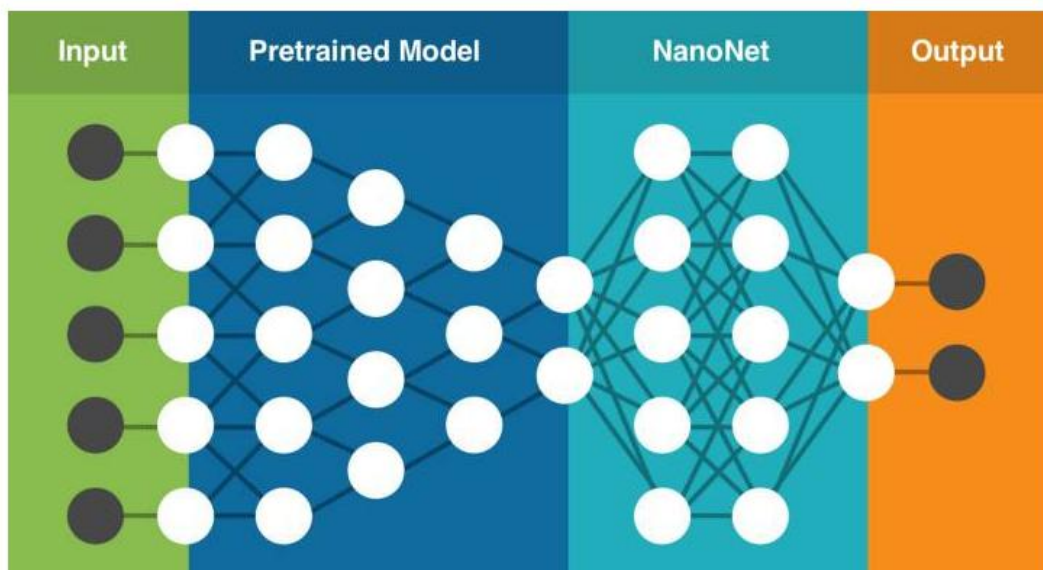


Figure 49. Transfer learning structure[137]

In transfer learning, not all models can migrate. In the above example, the source domain is a classification task, and the new task is the same as the source domain classification category. So, transfer learning can use to meet such conditions. If the target domain of the training is not closely related to the source domain, it will harm the training of the new target domain.[138] This situation is called negative transfer. It is also possible that the transfer learning fell because of the transfer method is not

good enough. For example, transfer learning usually migrates higher-order classification capabilities. Because the parameters of low-order extraction features are different, and the classification ability is determined by the higher-order features so that low-order transfer learning may cause negative transfer.[138] Figure.50 shows the visualization of the layered neural network. It shows that the lower-order layer can extract low-order features (points, line segments), while the high-level can classify high-order features.

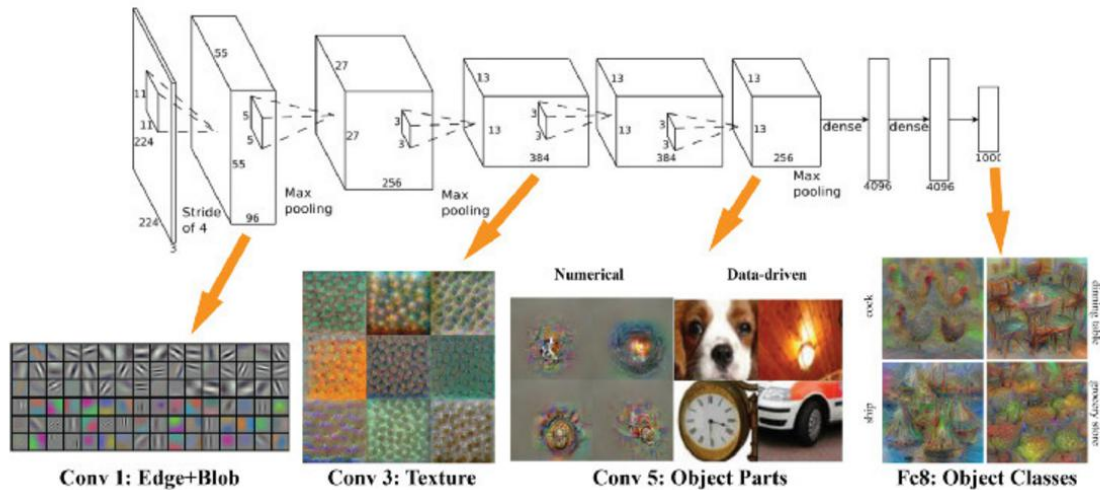


Figure 50. Different features extracted by convolution layers[138]

In addition to the role of training, transfer learning has a positive influence when the amount of data is small. It is also possible to effectively avoid over-fitting. However, there are still many difficulties to be overcome in transfer learning. These problems include[138]:

1. Obtain a relatively large amount of pre-trained data
2. Choose a suitable pre-trained model
3. It is difficult to find out which model is not working
4. I don't know how much extra data is needed to train the model.
5. It is challenging to judge under what circumstances should stop pre-trained
6. Decide on the level of the pre-trained model and the number of parameters
7. Agent and service model
8. Pre-trained models are difficult to update when getting more data or better algorithms

In short, transfer learning is similar to the process of human beings. In the future of deep learning, it will be improved together with the deep network model. Moreover, various tasks can produce better results.

## 2.11 Classic data sets

A good model also requires a good data to show the power of deep learning. In the classic problem of machine vision (classification and target detection, semantic segmentation), different type of models have different types of data sets. This chapter describes some data sets.

### MNIST[139]

As the most classic data set of classification networks, handwritten digital identification data sets are often used to verify algorithms. It contains 60,000 training sample and 10,000 testing samples. This dataset is an excellent database for practicing neural networks and pattern recognition models, also, spending the least amount of time and effort on data pre-processing. There are command statements that can call directly in tensor-flow and other architectures. The folder size is 50M, and the picture is grayscale. The sample of this data set shown in Figure.51.

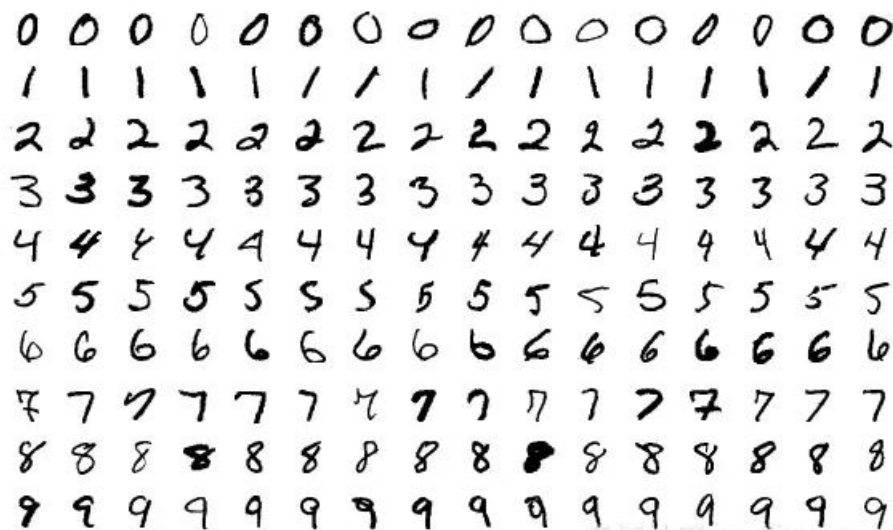


Figure 51. MNIST sample[139]

### ILSVRC[140]

ILSVRC initially originated from the Kaggle competition, which provided by Google with 1000 sorts and more than 1 million color pictures. The image labels also offer a maddening position because it also uses as an object detection data set. The data size is 150G, about 1.5 million color photos. The sample of this dataset shown in Figure.52.[140]

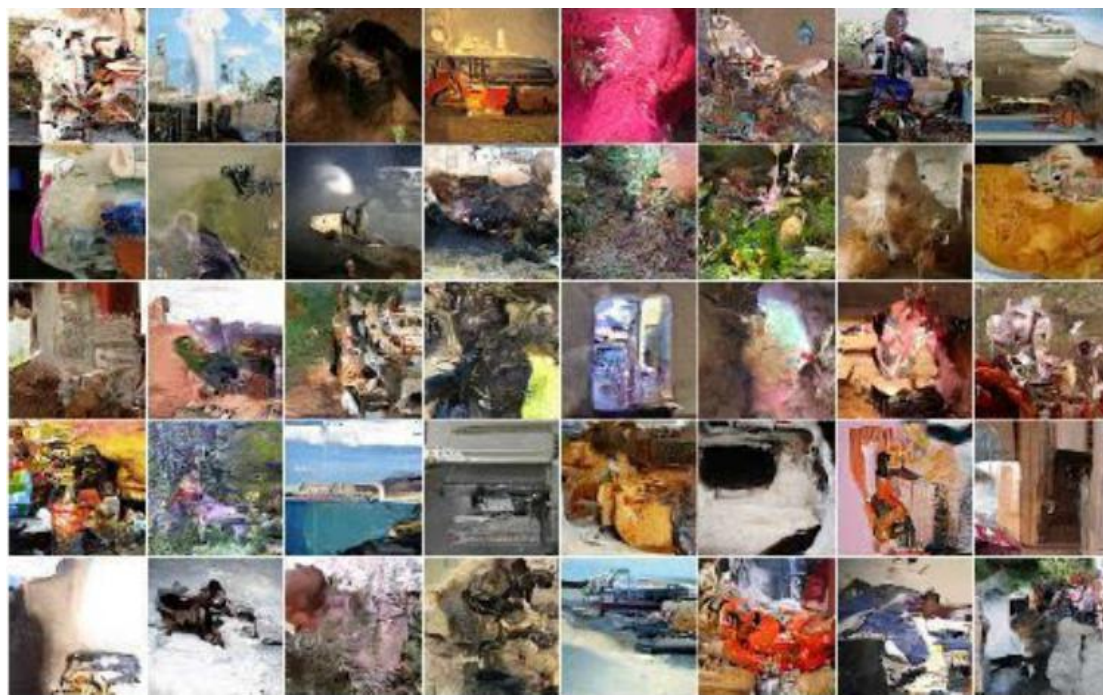


Figure 52. ILSVRC sample[140]

#### CIFAR-10[141]

This data set is another data set for image classification, which consists of 60,000 images with ten classes (each class represents a row in the sample below). There are a total of 50,000 training images and 10,000 test images. The data set is divided into six sections: 5 training batches and 1 test batch with 10,000 images per batch. The data size is 170M. The sample of this data set shown in Figure.53.[141]

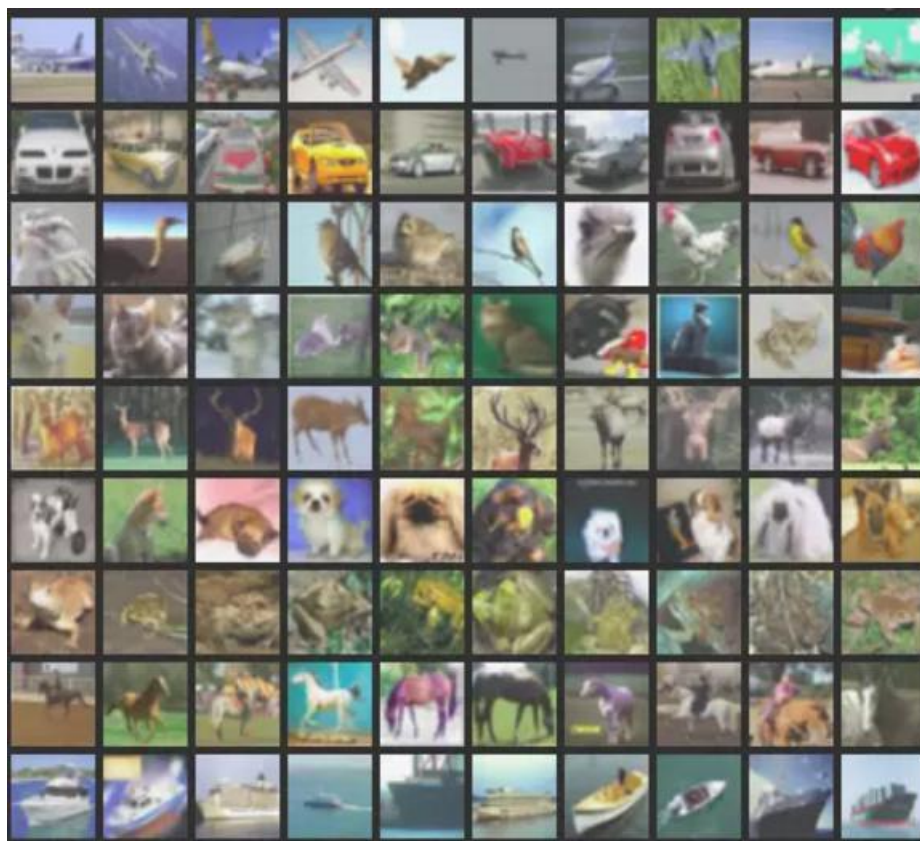


Figure 53. CIFAR-10 sample[141]

#### Fashion--MNIST[142]

With 60,000 training images and 10,000 test images, Fashion-MNIST is a fashion product database, which is similar to MNIST. Developers believe that MNIST has overused, so they use it as a direct replacement for this data set. Each picture is displayed in grayscale and is associated with ten categories of labels. The data size is 30M. It often used in natural language and classification tasks, and is more complicated than the traditional MNIST. The sample of this dataset shown in Figure.54.[142]

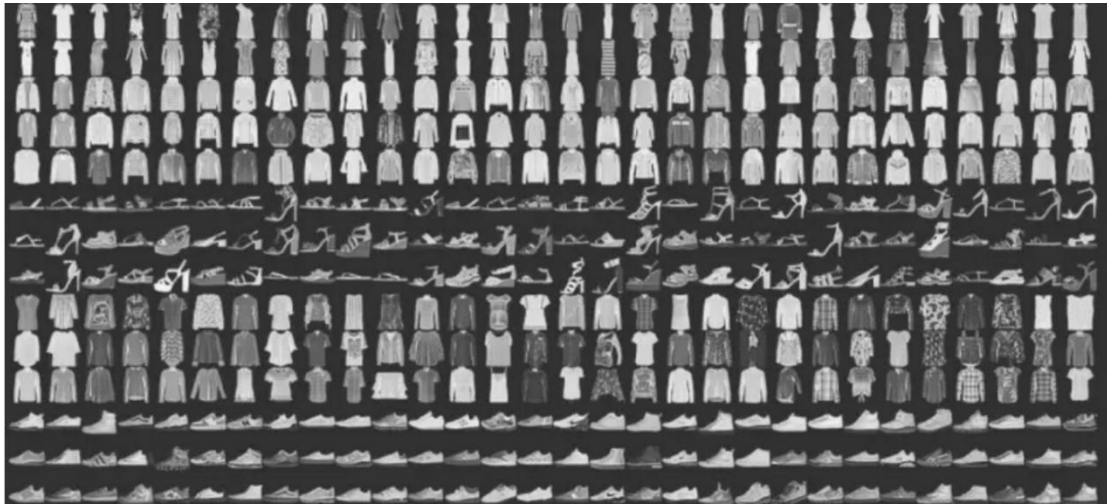


Figure 54. The sample of Fashion--MNIST.[142]

### SMILES[143]

The SMILES dataset contains face images that either laugh or not, with a total of 13165 grayscale images in the dataset, and the size of images are 32\*32. The image is around the face, which allows us to design a machine learning algorithm that focuses on smile recognition. The sample of this dataset shown in Figure.55.[143]



Figure 55. Sample of SMILES[143]

### Kaggle: Dogs vs. Cats[144]

This dataset is a part of the Kaggle competition and designed to classify cats and dogs from the images. A total of 25,000 images with different resolutions. How to pre-processing the dataset is the first step to do, also directly influence the final classification result. The sample of this dataset shown in Figure.56.[144]

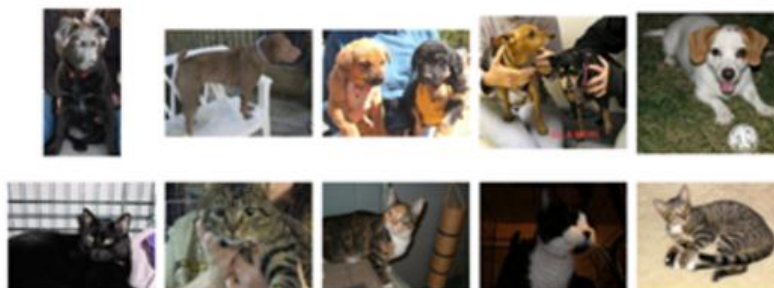


Figure 56. Sample of dogs vs. Cats [144]



## Pascal VOC[145]

The VOC data set is a well-known data set in the fields of target detection, classification, segmentation, etc. It will be held from 05 to 12 years (the game has tasks: Classification, Detection (all the targets in the picture are bounded by the bounding box), Segmentation (split all the targets in the image), Person Layout). The PASCAL VOC contains approximately 10,000 images with bounding boxes for training and verification. However, the PASCAL VOC data set contains only 20 categories, so it is considered a benchmark data set for target detection problems. Figure.57 shows the sample of classification dataset and Figure.58 shows the sample of segmentation dataset.[145]



Figure 57. Sample of Pascal VOC (classification)[145]

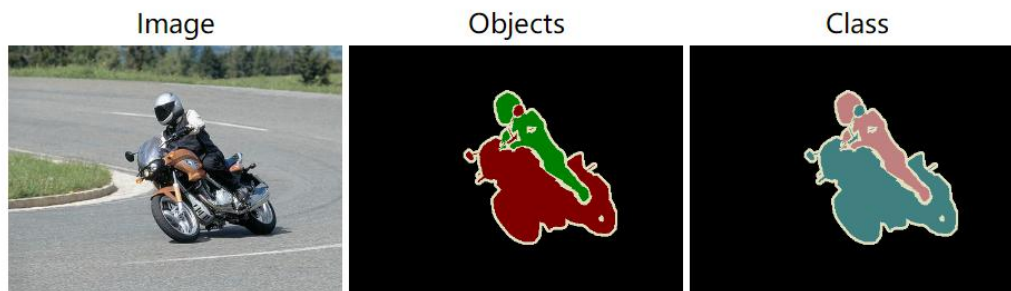


Figure 58. Sample of Pascal VOC (segmentation)[145]

## MS-coco[146]

This data set is used in a variety of competitions: image classification, object detection, keypoint detection, and object segmentation. For object detection missions, COCO consists of 80 categories. The annual training and validation dataset contain over 120,000 images and over 40,000 test images. The label for the test set is not exposed to avoid overfitting on the test set. The sample of this dataset shown in Figure.59.[146]

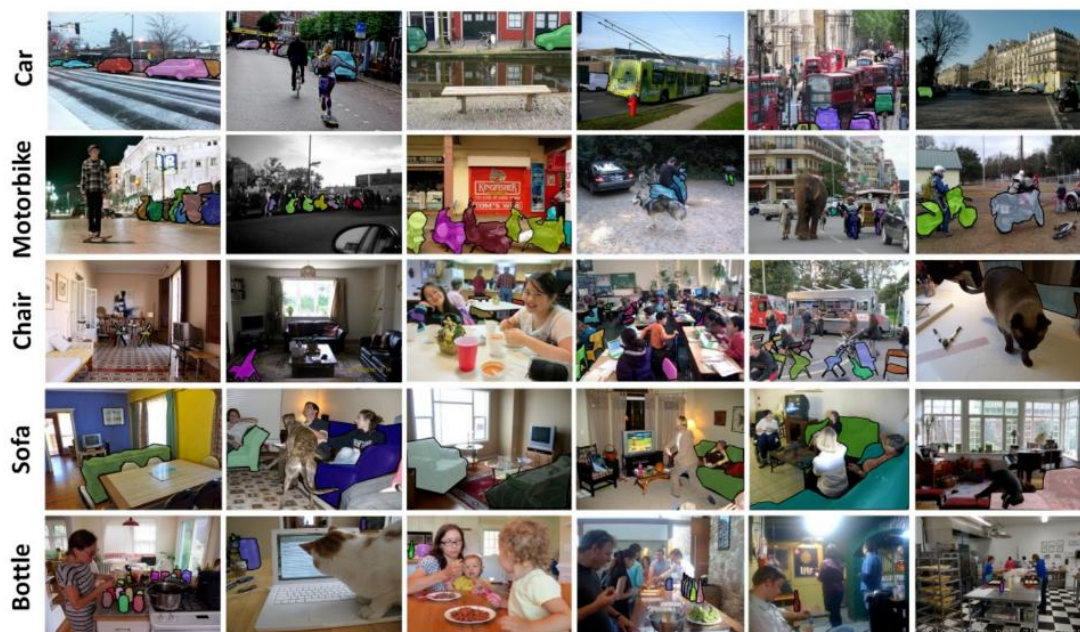


Figure 59. Sample of MS-coco[146]

## 2.12 Video data set[147]

Advances in video recognition have attracted the interest of researchers, as well as a large number of competitions and high-quality data sets. Data sets widely used in academia include UCF-101 [150], HMDB-51 [151], Youtube-8m [149]. In addition to the data set, related competitions have also promoted research in the field of video behavior recognition, most notably the TRECVID [148] competition and the Thumos [147] competition. TRECVID[148] is organized and implemented by the National Institute of Standards and Technology. It has been held annually since 2001, and the Thumbs competition has held since 2013. The figure below is a sample of some data sets.

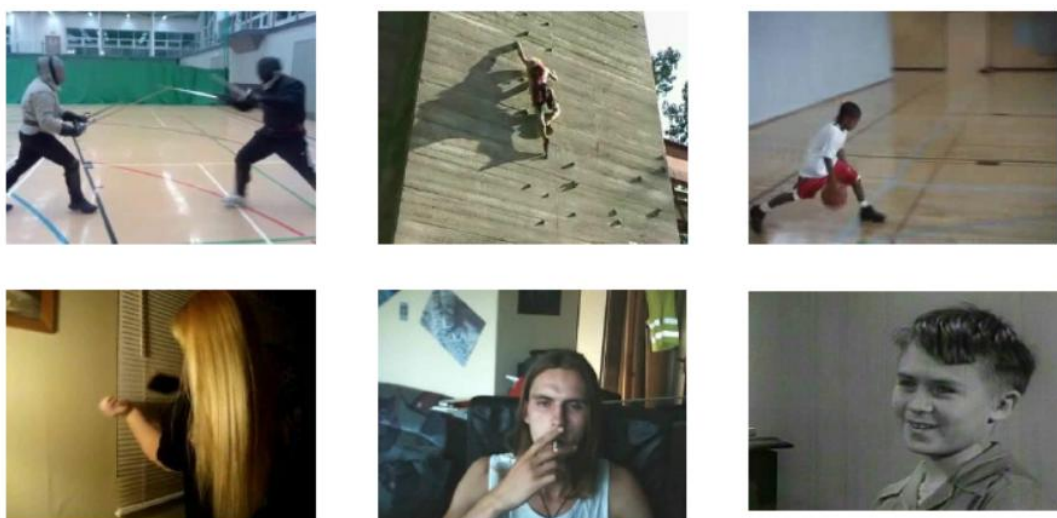


Figure 60. HMDB-51 data set sample[147]

The HMDB-51[150] data set contains a total of 6849 videos with 51 different behavior categories. As shown in the Figure.60, this article selects some of the categories of videos as an example. The behavior of the data set summarized into five types: 1) standard facial actions: smile, talk, chew, laugh; 2) facial activities interacting with the objects: drink, smoke, eat; 3) ordinary body movement climb, clap hands, Cartwheel; 4) interact with objects body movement: brush hair, dribble, catch; 5): interact with people body movement: hug, kick, punch.

The UCF-101[151] data set is extended from the UCF-50 data set and contains 101 action categories, as shown in Figure.61. The UCF-101 dataset behaviors can group into five categories: 1) normal body movement; 2) interaction with objects; 3) interpersonal interaction; 4) instrumental performance; 5) physical activity. The data set has 13,320 video clips. The data set contains many uncertain factors such as complex background, different illumination, camera movement, which is very challenging and attracted the attention of many researchers.

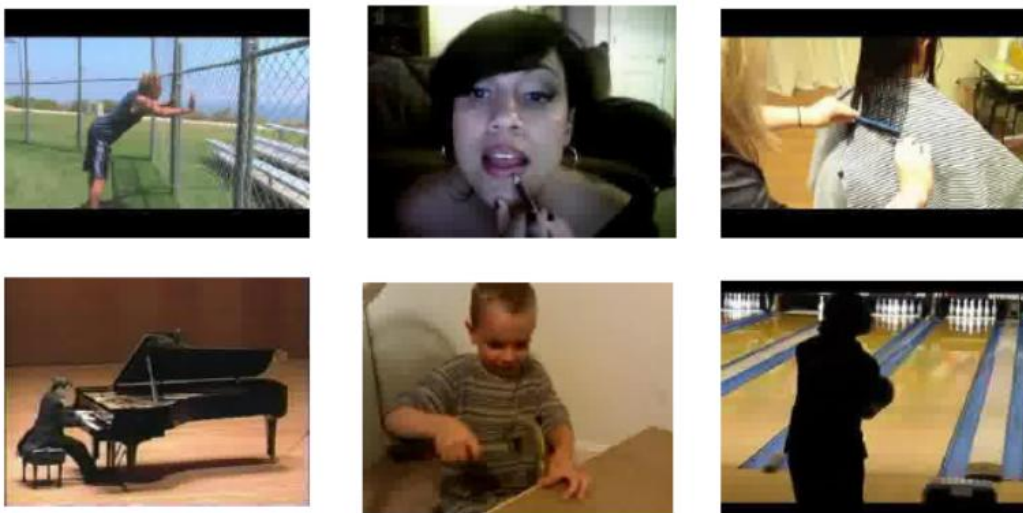


Figure 61. UCF-101 data set sample[151]

### **RAVDESS[152]**

The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) is a high-quality data set, which used many years. This data set contains the complete set of 7356 RAVDESS files (total size: 24.8 GB). Each of the 24 actors consists of Three modality formats: Audio-only (16bit, 48 kHz .wav), Audio-Video (720p H.264, AAC 48 kHz, .mp4), and Video-only (no sound). Note, there are no song files for Actor\_18. In the video of this database, volunteers were asked to say a sentence with different feelings, which lasted for 3 seconds. [152]At the same time, there was a strong tone and a normal tone to express the degree of emotion and repeated twice to increase the number of videos in the data set. The original video frames are 1280\*720\*3.



Figure 62. Sample of RAVDESS[152]

### 2.13 Summary of Chapter 2

This section introduces the concepts of CNN and RNN and the process of feature extraction operations. Commonly used layers in neural networks are presented. The classic classification network, including Alex-net, Google-net, and Res-net, are introduced and discuss the differences of various networks. According to different design methods in the object detection algorithm, introduce two types of object detection. The algorithm based on region proposal introduces RCNN, SPP-Net, Fast RCNN, Faster RCNN, and presents YOLO series, SSD, which based on the regression algorithm. After the algorithm of the picture, this chapter also introduces two video analysis methods, conv LSTM, and 3D CNN for the video processing algorithm. Besides, the basic face model and the face recognition introduced in algorithm 2.9. Section 2.10 mentioned the neural network concept of model training and transfer learning. The final section presents some classification, segmentation, and object detection data sets.

---

## **Chapter 3: Human Emotion Recognition in Video using Subtraction**

### **Pre-Processing**

#### **3.1 Introduction**

This chapter introduces a new image pre-processing method, which shows essential temporal in videos clearly. Deep learning methods get a higher result than traditional machine learning in many areas, which attract people's attention in recent ten years.

In multi-classes classification challenges, the Deep learning series' model shows its excellent performance. Video Facial analysis system is a hot topic and will become necessary in the robotics industry and auto-motion traffic domain. The new system presented in this chapter combine by Convolution Neural Network (CNN) and a new video pre-processing technology. Because we found personal emotions are dynamic, then we proposed this new pre-processing method. The key point in this method is the movement of the features in the face. RAVDESS[152] is the chosen video set, which is used to train and test the whole system. The video without audio set in RAVDESS[152] is taken for focusing on video frames abnormalities. RAVDESS[152] includes six different emotions in the chosen video set. Each volunteer in the video presents a sentence acting with emotion. Based on the chosen video set, a system has designed and trained with a new video pre-processing method. Also, the result of this new method has compared with other emotion recognition method in which RAVDESS[152] data set used.

#### **3.2 Related work**

The project of facial emotion recognition has different methods to achieve. In short, it is a classification task of emotions. The difference between video and image leads to different solving methods. A single image is unique, while videos have temporal features between each frame. Researchers also found that the relationship between frames should take into consideration. This new pre-processing approach inspired by the traditional facial recognition shown in the figure.63

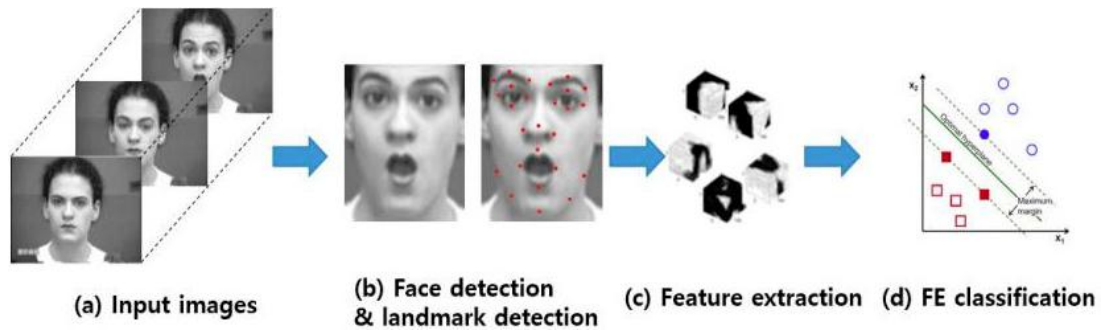


Figure 63. Tradition emotion recognition method [5]

There are four fundamental steps in the traditional video analysis method. First, frames in the video are sampled. Second, the frames from the previous step will do face detection or face landmark. This process is shown in Figure 1(a) and 1(b). Face landmark and face detection technology can extract facial features using CNN or other machine learning approaches shown in figure 1(c). Finally, the classifier will use the features in each frame to show the result of the frame, which shown in figure 1(d). This method is reasonable, while this method's leakage is that the relationship between the frames does not take into account. The traditional method is very similar to an image classification method. Also, the traditional method wastes time in extracting useless features in the video's frames.

Researchers found that the relation of the frame is essential. Also, some related works have done in recent years. These works usually are using CNN combined with long short-term memory (LSTM) [153], which are used to extract features and understand temporal sequence features, respectively. In picture classifying tasks, CNN is considered the best performance, and LSTM is used to understand the temporal feature of the video sequence. The video analysis system that includes these two structures achieve a better performance than traditional methods. Some related works proved such models' ability. Some excellent studies using such models are as follows, Kahou et al. [157] using simple RNN (Recurrent neural network) with a CNN framework. In the 2015 Emotion Recognition in the Wild (EmotiW) Challenge [158], the results in the paper show that the joint of RNN improves the performance of emotion expression recognition. Kim et al. [159] divide emotion recognition into two parts. The first part using a Deep learning neural network to extract frame features. Then the second part using first part features to train an LSTM structure for temporal information understanding. Graves et al. [160] set bi-LSTM and unidirectional LSTM structure to extract the temporal features. The bi-LSTM extract temporal feature from forward and backward, which performs better than single order.

### 3.3 Our approach

From the previous discussion, the relation between frames in a video proved critical, while useless information in the frames still a problem. For example, only face area in the frames is regarded as the useful part, which influences emotion recognition. So, a pre-processing step in videos could add before the emotion classification to increase the performance.



Figure 64. Difference dataset sample with high FPS and high-quality camera [161]

Video also is an image sequence. If the camera set in a fix position, then most of the background will be no different. So, if we subtract two frames in the video, the result will show the difference between these two frames. This pre-processing method could create a perfect image dataset only if the FPS(frames per second )are high with a high quality video. Figure.64 shows a subtract sample with high FPS and quality video. The background turns to black with slight pepper noise. The details and the shape of the shoe are obvious after the video's subtract operation. After the pre-processing, a new subtract image dataset created. Use the difference of the video set to analysis video in the real world and videos is the novelty of this paper. While CNN is still needed to extract images features and other operations to reduce the useless part of the image. The pre-processing includes face detection, face alignment.

### 3.4 Model structure

This section presents the details about the whole emotion recognition system. Figure.65 shows the whole system sketch, which includes pre-processing and test progress. The video dataset of RAVDESS [162, 163] used in this paper. First, the video decompressed into single images. Face detection and face alignment processing technique used before subtracting operation for focusing on face. Gap and stride value need to set in subtract operation. The gap value is the distance between two frames in a single subtract operation. If the gap is too small, then two frames will have

less difference, and the mean value of the subtract image will close to zero, which means subtract method losing too many features and cannot be accepted. If the gap is too big, then there is a weak relationship between two frames, which also means the goal of the method fell and will not lead to better performance. Also, the stride needs to set carefully, which has the same meaning as a stride in a convolution neural network. The value of strides controls the subtract image database size. After pre-processing, CNN structures are used to in extracting features and classification task. AlexNet [164], GoogleNet [165], ResNet [166] structures are used and compared.

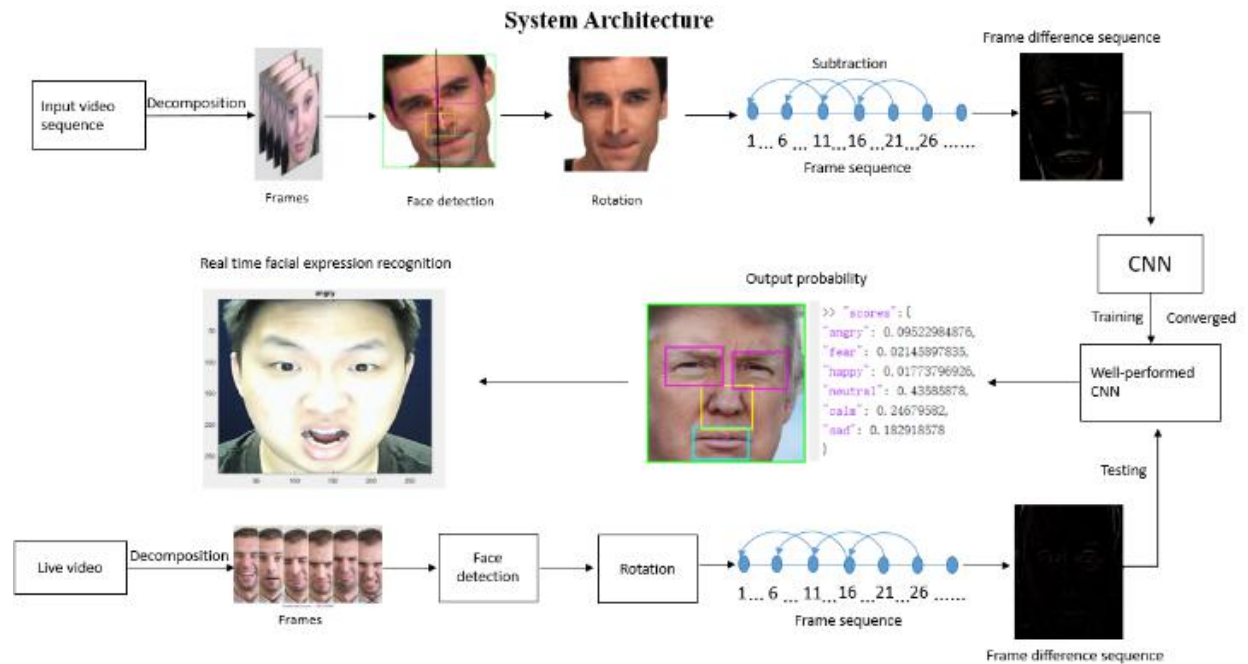


Figure 65. Sketch of new pre-processing system[162]

### 3.5 Face Detection

Haar features are used to detect the face in the picture in this method. In order to detect faces from a picture, haar-like needs several steps. First, haar-like features need to set a threshold to decide the face part. Figure.66 shows some haar filters [167], which used to extract features; each filter detects different types of edge. The second step is face detection. In the face detection algorithm, every part of the original picture will be divided into two classes, a face or not a face. After face detection, the face part will stay, while the place which considered not a face will remove. Figure.67 shows the process of decision. Also, eyes, nose, mouth, can be detected at the same time. A test of the face detection algorithm shown in figure 68. In figure 68, nose, eyes, and mouth with the face are detected after face detection.



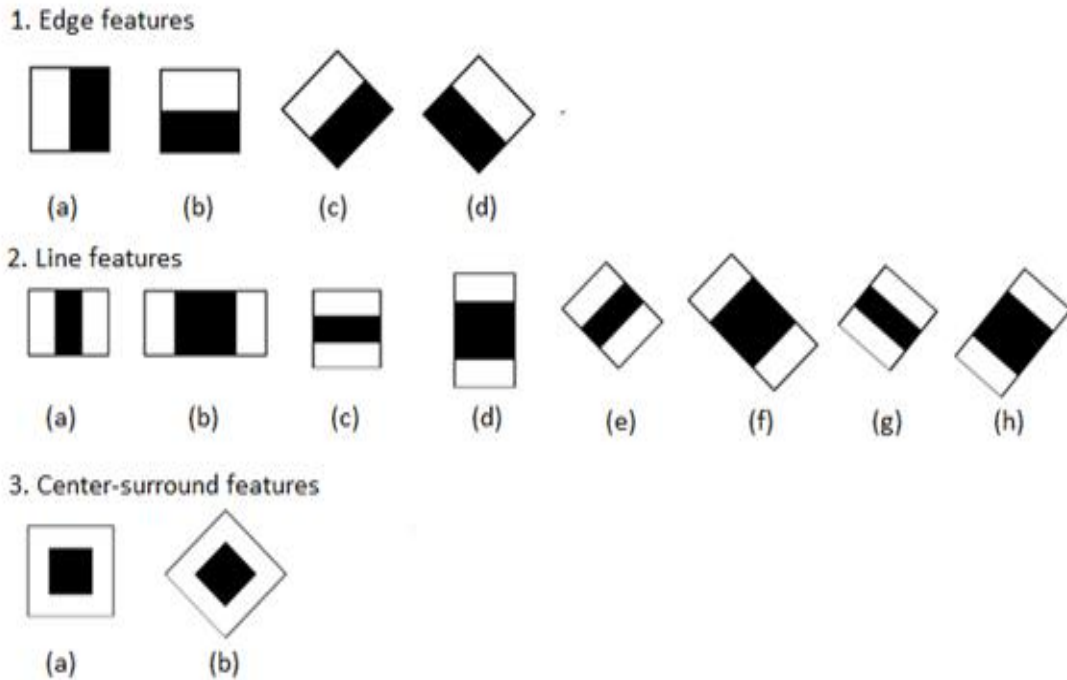


Figure 66. Haar filters sample [77]

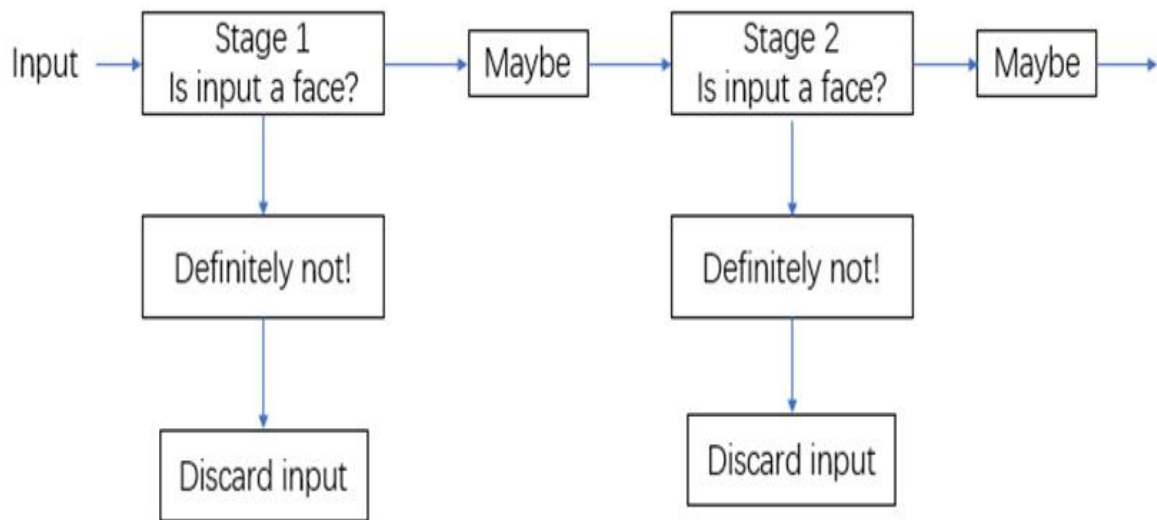


Figure 67. Haar-like decision diagram[157]

When testing a video, most of the background disappeared. Face detection and face alignment are used before the subtract operation. The face detection and face alignment algorithms have an impact on the final expression recognition. When the face is detected, the face of the face is wrong, and the feature of the subtracted image is also wrong. As a result, the final expression classification result is biased or wrong.

### 3.6 Face Alignment

Face alignment is another essential pre-processing. Some volunteers have some significant head movements when they talk or sing. Large movements can confuse features, which will lead to the result after the subtract is sensitive to the edge of the face rather than the difference of the face. In order to avoid this situation, an idea was present to force the face in the image straight by other parts detection results. As shown in figure 69, Face detection also can detect other parts of the face. Figure 68 shows the result of solving the rotated face problem. The steps for finding out the angle are as follows,

- (1) Finding angle:
- (2) Find the middle point of two eyes' boxes.
- (3) Find the middle point of two points in step 1.
- (4) Find the middle point of the mouse box.
- (5) Connect the points found in step 2 and 3.
- (6) Calculate the angle between the line in step 4 and the vertical line.

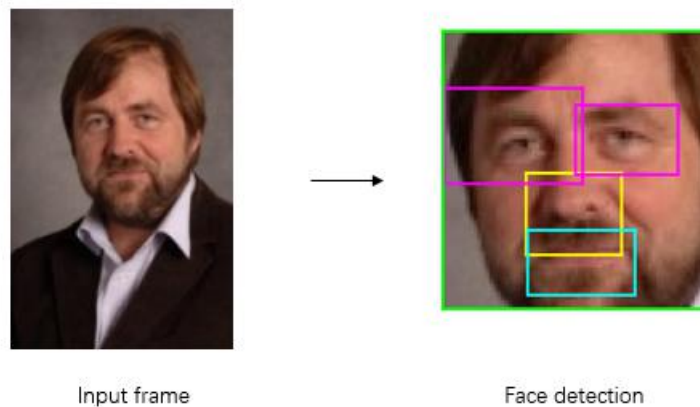


Figure 68. Face detection  
Face alignment

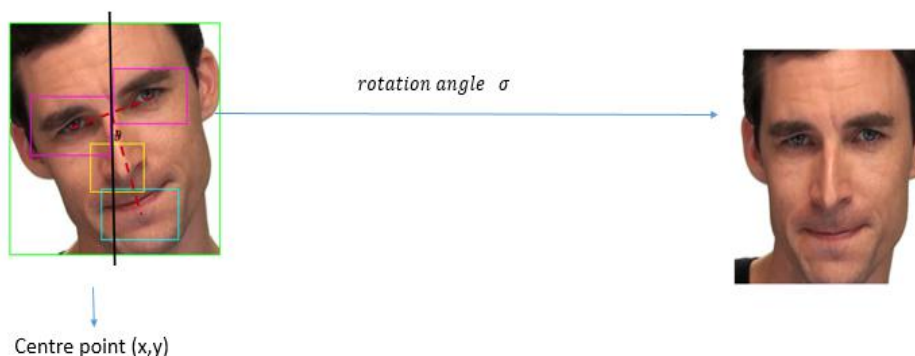


Figure 69. Example of a rotated face[157]

If the rotation angle of the face has been calculated, a positive face can be obtained, and a better result will be achieved.

### 3.7 Result and analysis

A new pre-processing method and a CNN based classifier consist of this new emotion recognition system. Alex-net, google-net, and different deep Res-net structure are chosen and used as the CNN part. The video database used to test is RAVDESS. This database includes six emotions, which include happy, angry, neutral, calm, fear, and sadness. The size of the original video frames is 1280\*720, which is too big. After face detection and alignment, images are resized to a fixed size. This dataset has 24 volunteers with 12 females and 12 males, each of them singing some sentences with emotion in about 3s video. This video set has six classes, which are neutral, calm, happy, angry, fear, and sad. The value of the gap is set as 5, and the value of the stride is set as 4. An image database is created, and the count of each class is shown in table 5.

Label	Count
Angry	5786
Calm	6644
Fear	5710
Happy	6021
Neutral	3024
Sad	6483

Table 5 Details of database[157]

The database has 33668 images. This database is used to train the CNN structure and trained in Matlab. The ratio of the training set, test set, and the validation set is 8:1:1. Different deep of ResNet is built and trained. ResNet-101, ResNet-51, ResNet-10, ResNet-8 and ResNet-4 are tested. After all, ResNet-4 is chosen in all ResNet structure, which has the best result of all ResNet models and is regarded as well trained.

Table 6 shows some important parameters which play a vital role in training the neural networks. Single GPU (NVIDIA GeForce GTX 1070) is the hardware chosen.

Settings	Learning Rate	Max Epochs	Learning Rate Factor	Learning Rate period	Max Bitch Size	Environment
Alex-Net Base	0.001	15	0.7	2	128	Single GPU
Google-Net Base	0.0001	10	0.2	2	8	Single GPU
ResNet Base	0.0005	6	0.2	1	32	Single GPU

Table 6 The different parameter values chosen for training the networks[157]

Table 7 shows the final result of each CNN model. Two of the classifiers that are thought to achieve a little better accuracy rate than this new method which is Biqiao Z et al. model which achieved an accuracy rate of 83.15% on Acoustic+Visual dataset and Frank A. Russo et al. model based on 247 raters classifier achieved 80% accuracy rate with Acoustic+Visual dataset. However, our model only trained with visual data type and has the best accuracy rate of 79.74% using the Alex-net structure.

Authors	Data type	classifier	Accuracy
Biqiao Z et al	Acoustic+ Visual	Shared models	83.15%
Tuanbo G et al	Acoustic	Global feature SVM	79.40%
Frank A. Russo et al	Acoustic+Visual	247 raters	80%
Frank A. Russo et al	Visual	247 raters	75%
Frank A. Russo et al	Acoustic	247 raters	60%
<b>Our model</b>	<b>Visual</b>	<b>CNN</b>	<b>79.74%</b>

Table 7 Accuracy Comparison[157]

In figure 70 the blue line and the red lines shos the training accuracy and the loss of the cross-entropy in training set respectively. The blakc point is the check point and the black line linked by check points is validation accuracy and loss. In the Alex-Net training process, we found after 4 to 5 epochs iteration, the validation loss didn't change. So we stop here and get a 79.74% accuracy.

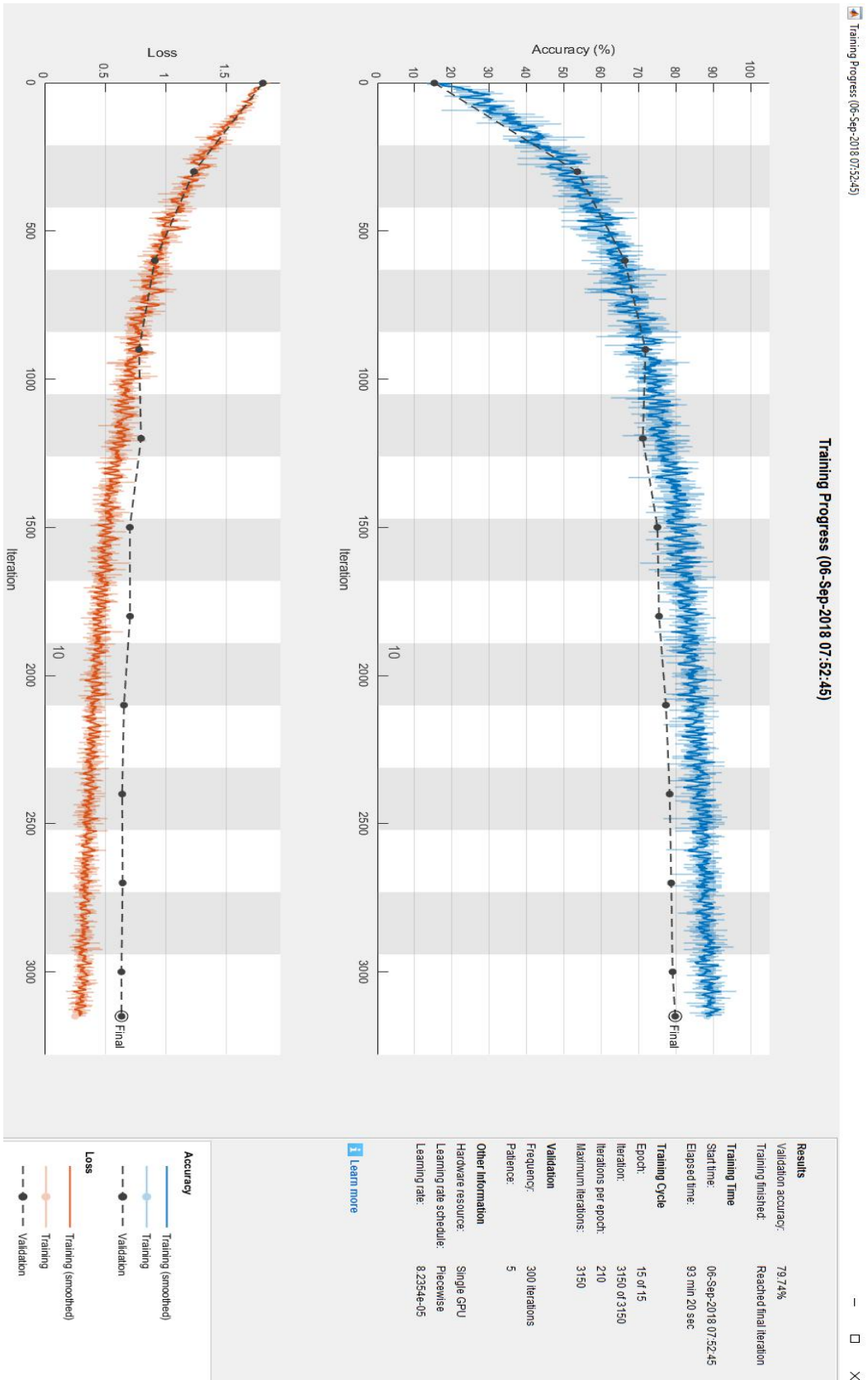


Figure 70. Alex-net training process[157]

The GoogleNet is very complicated with over one hundred convolution layers, so it is very hard to train. Finally, the classify accuracy is 62.89% shown in figure 71.

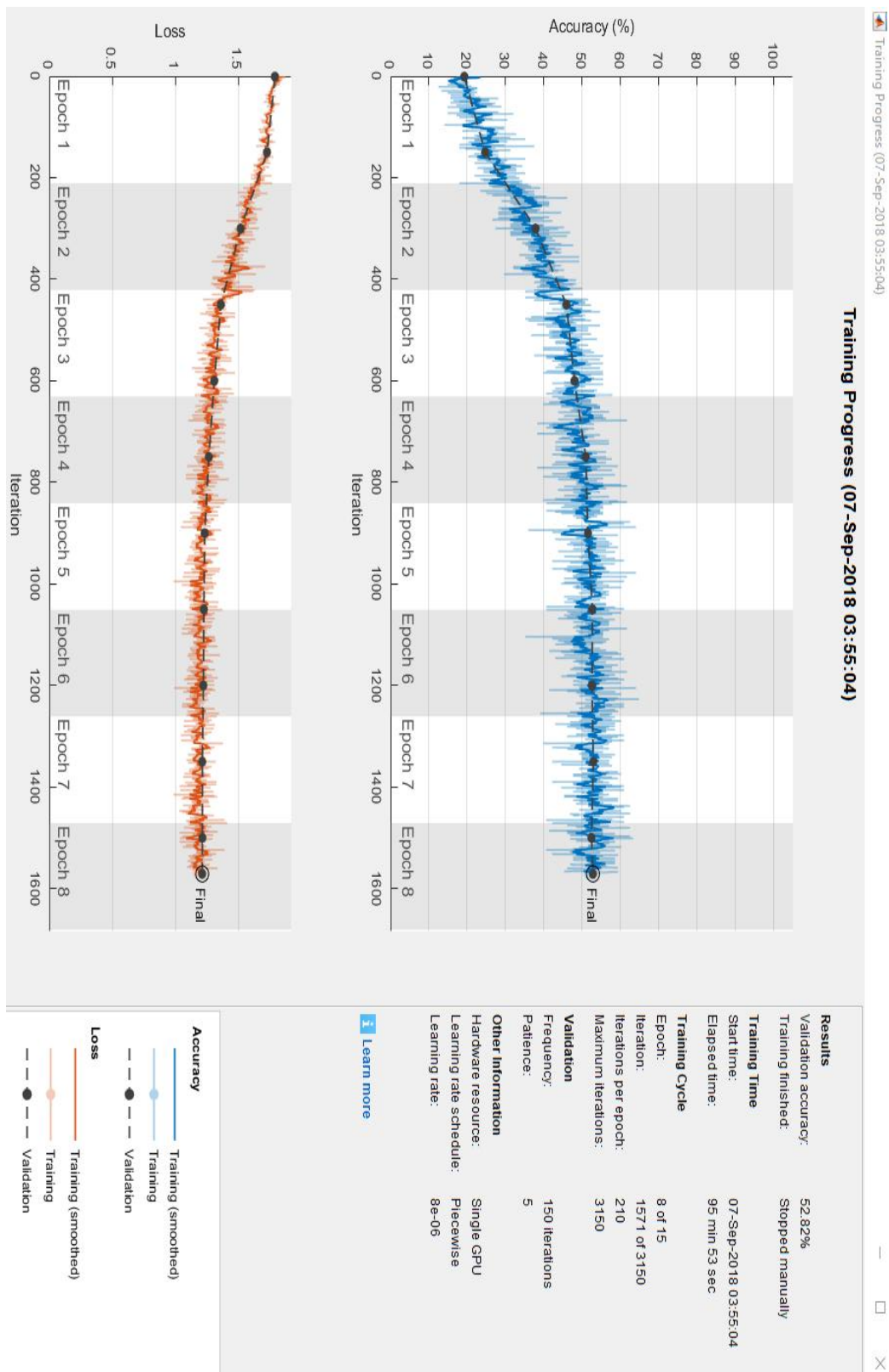


Figure 71 GoogleNet structure training process[157]

The Resnet structure is well trained as shown in the figure 72. The loss and learning rate behaviour show the model fits the dataset well, but the accuracy is 75.89% at the end. The accuracy of this model is not higher than Alex-net, a deeper structure might improve the accuracy of classification.

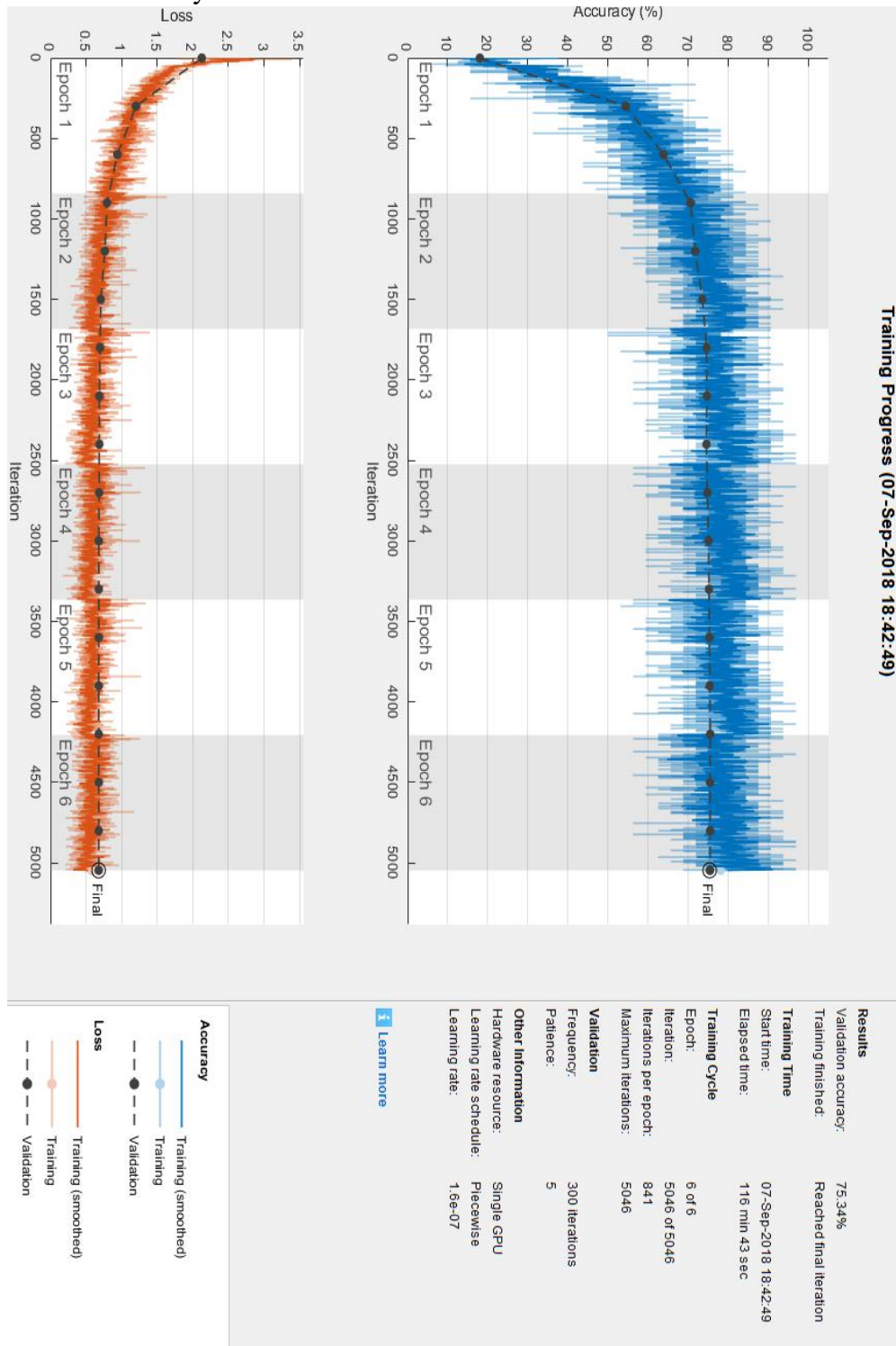


Figure 72 Res-Net structure training process[157]

As can be clearly seen from Table 8, the overall structure based on AlexNet and the ResNet-based structure show the best results, but GoogleNet layers number is more than 100, while the number of layers in AlexNet and ResNet-4 are far more less than 100. Deeper ResNet structures have also been tested, but the best performance is ResNet-4. In general, deep convolution structures have ability extract features from images and analysis or understand features. However, the pre-processing operation reduces many unimportant features and causes the most static pixels to darken. The new image dataset created in this method is not complicated because most meaningless points become zero. In some aspects, the pre-processing operation achieve its goal, which is help CNN analysis the features. Figure 73 shows some of the actual tests for this model.

CNN Structure	Accuracy (%)
Alex net structure	<b>79.74</b>
Google net structure	62.89
ResNet-4	75.89

Table 8 Results of three structures[157]



Figure 73 (a) Real test (camera vision: angry sad fear neutral happy)

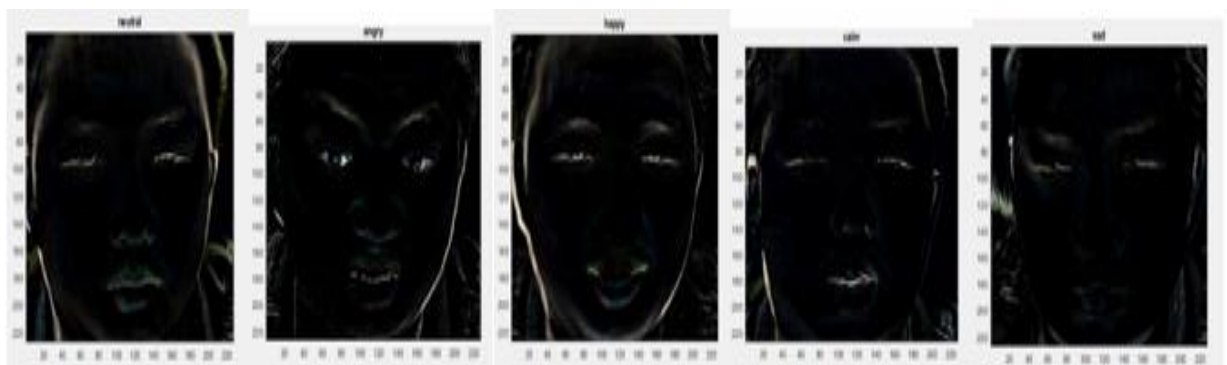


Figure.73 (b) Real test (difference vision: neutral angry happy clam sad)[157]



### **3.8 Conclusions**

The proposed pre-processing method can produce an accuracy of 79.78%, which is 4.78% higher than the accuracy of human judgment.[157] This method can analyze people's emotions, but the insufficient number of frames per second will affect the results and timeliness of the analysis. Problems in face detection and face alignment techniques should be avoided. Therefore, better face recognition and alignment techniques should be tested or used.

## **Chapter 4 : Conclusion and Future work**

### **4.1 Conclusion**

This paper proposes a new video pre-processing method, and uses this method to construct a new set of video facial expression recognition models. Many video processing methods has exist and the relationship between frames is considered to be the key point of video analysis technology. But the connection between video frames has never been effective used in deep learning. This essay believes that there is a pattern of facial movements when people generate expressions, and this pattern can be used for the feature construction of human expression recognition.

The video pre-processing method implemented in Chapter 3 uses the difference map between the vibration frames and put the difference map into the convolutional neural network for feature extraction and learning. The test results base on Matlab show that the video processing speed and accuracy are improved. The video data becomes a differential feature picture data set after processing. In the new differential image data set, people can clearly see the characteristics of subtle expression changes. In addition to the pre-processed video method in the implementation process, it also involves methods such as face detection, face correction, and convolutional neural network classifiers. This video analysis method can be used to analyze video or real-time video streams. There are also some problems in the implementation of this method that need to be improved.

- The result of face detection plays a decisive role in the final result. If the results of face detection are not ideal, the final analysis will be very unreliable.
- The quality of the video used needs to be high quality, and the subtle facial differences need to be captured during the difference process. A low quality video material will be misleading for video analysis.
- At present, the analysis of face occlusion and incomplete facial features will be unreliable.

This report also introduces some key points of images and videos, as well as the basic

---

knowledge of deep learning, including convolutional neural networks and recurrent neural networks. At the same time, the classification network, object detection network and face detection model in deep learning are analyzed. The problems encountered in training neural networks in practical experiments are analyzed. Finally, we show some commonly used data-sets in learning.

In short, this article introduces some basic knowledge of image processing, video processing, neural networks, model training. A new video processing scheme is proposed, and the scheme is implemented and verified to be used in video analysis problems. Finally published the paper “*Human emotion recognition in video using subtraction pre-processing*” in conference ICMLC2019 and “*Dense convolutional networks for efficient video analysis*” in conference ICCAR 2019.

## 4.2 Future work

Although the expression recognition algorithm proposed in this paper has proven to be effective, the accuracy still needs to be improved. In the experiments, the face detection is still not good and the background of video in the database is very simple. The expression recognition system still has some difficulties in the real life, which has complex background. In addition, the method based on the difference between frames, there are still other processing methods to find the relation features between frames can be developed. Based on the research of video analysis technology, some directions of expression recognition are proposed.

- For complex and ever-changing backgrounds, 3D reconstruction technology can be used to simulate the 3D shape of human faces to remove background interference. 3D reconstruction of human faces can provide richer data for convolutional neural networks to find subtle changes in people.
- The quality of the data used in the video expression recognition method implemented in Chapter 3 of this essay is not high enough, some features are loss when processing the video. If the number of frames is very high, the result will better.
- In expression analysis, speech information can be used. The dataset used in this essay does not contain speech information. Voice information used as a feature of video facial expression analysis is believed greatly improve the result.

---

## Reference

- [1]J. H. Bappy, A. K. Roy-Chowdhury, J. Bunk, L. Nataraj, and B. Manjunath. Exploiting spatial structure for localizing manipulated image regions. In ICCV, 2017.
- [2]T. J. De Carvalho, C. Riess, E. Angelopoulou, H. Pedrini, and A. de Rezende Rocha. Exposing digital image forgeries by illumination color classification. TIFS, 2013.
- [3]R. Girshick. Fast r-cnn. In ICCV, 2015.
- [4]W. Gao et al., "The CAS-PEAL large-scale Chinese face database and baseline evaluations," *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, vol. 38, no. 1, pp. 149-161, 2008.
- [5]S. H. Yoo, D. Matsumoto, and J. A. LeRoux, "The influence of emotion recognition and emotion regulation on intercultural adjustment," *International Journal of Intercultural Relations*, vol. 30, pp. 345-363, 2006.
- [6]P. Ekman, "Emotions revealed: Recognizing faces and feelings to improve communication and emotional life," 2003.
- [7]A. Poursaberi, H. A. Noubari, M. Gavrilova, and S. N. Yanushkevich, Gauss–Laguerre wavelet textural feature fusion with geometrical information for facial expression identification," *EURASIP Journal on Image and Video Processing*, vol. 2012, no. 1, pp. 1-13, 2012.
- [8]M. M. Ibrahim, "Video processing analysis for non-invasive fatigue detection and quantification," University of Strathclyde, 2014.
- [9]Y. LeCun, F.J. Huang, and L. Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–97. IEEE, 2004.
- [10]A. Berg, J. Deng, and L. Fei-Fei. Large scale visual recognition challenge 2010. [www.image-net.org/challenges](http://www.image-net.org/challenges). 2010.
- [12]D. Cires,an, U. Meier, and J. Schmidhuber. Multi-column deep neural networks for image classification. *Arxiv preprint arXiv:1202.2745*, 2012.
- [13]A. Krizhevsky and G.E. Hinton. Using very deep autoencoders for content-based image retrieval. In *ESANN*, 2011.
- [14]Y. LeCun, K. Kavukcuoglu, and C. Farabet. Convolutional networks and applications in vision. In *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pages 253–256. IEEE, 2010.
- [15]T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. Metric Learning for Large Scale Image Classifi-cation: Generalizing to New Classes at Near-Zero Cost. In *ECCV - European Conference on Computer Vision, Florence, Italy, October 2012*.
- [16]D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov. Scalable object detection using deep neural networks. In *CVPR*, 2014
- [17]A. Krizhevsky, I. Sutskever, and G. Hinton. Im-agenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25*, pages 1106–1114, 2012.

- 
- [18]C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In C. J. C. Burges, L. Bottou, Z. Ghahramani, and K. Q. Weinberger, editors, NIPS, pages 2553–2561, 2013.
- [19]J. Dean, G. Corrado, R. Monga, K. Chen, M. Devin, M. Mao, M. Ranzato, A. Senior, P. Tucker, K. Yang, Q. V. Le, and A. Y. Ng. Large scale distributed deep networks. In P. Bartlett, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, NIPS, pages 1232–1240. 2012.
- [20]S. Arora, A. Bhaskara, R. Ge, and T. Ma. Provable bounds for learning some deep representations. CoRR, abs/1310.6343, 2013.
- [21]U. V. C. Atiyeh, C. Aykanat, and B. Ucar. On two-dimensional sparse matrix partitioning: Models, methods, and a recipe. SIAM J. Sci. Comput., 32(2):656–683, Feb. 2010.
- [22]P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. CoRR, abs/1312.6229, 2013.
- [23]J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In CVPR, 2015.
- [24]T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick. Microsoft COCO: Common objects in context. In ECCV. 2014.
- [25]K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In ICCV, 2015.
- [26]K. He and J. Sun. Convolutional neural networks at constrained time cost. In CVPR, 2015.
- [27]S. Gidaris and N. Komodakis. Object detection via a multi-region & semantic segmentation-aware cnn model. In ICCV, 2015.
- [28]F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In CVPR, 2007.
- [29]A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio. Fitnets: Hints for thin deep nets. In ICLR, 2015.
- [30]A. M. Saxe, J. L. McClelland, and S. Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. arXiv:1312.6120, 2013.
- [31]C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In CVPR, 2015.
- [32]M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional neural networks. In ECCV, 2014.
- [33]I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. Technical report, arXiv preprint arXiv:1409.3215, 2014.
- [34]J. Martens and I. Sutskever. Learning recurrent neural networks with Hessian-free optimization. In Proc. ICML’2011. ACM, 2011.
- [35]C. Gulcehre, K. Cho, R. Pascanu, and Y. Bengio. Learned-norm pooling for deep feedforward and recurrent neural networks. In Machine Learning and Knowledge Discovery in Databases, pages 530–546. Springer, 2014.
- [35]Y. Bengio, N. Boulanger-Lewandowski, and R. Pascanu. Advances in optimizing recurrent networks. In Proc. ICASSP 38, 2013.

- 
- [36]D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. Technical report, arXiv preprint arXiv:1409.0473, 2014.
- [36]A. Graves. Supervised Sequence Labelling with Recurrent Neural Networks. Studies in Computational Intelligence. Springer, 2012.
- [37]I. Almajai, S. Cox, R. Harvey, and Y. Lan. Improved speaker independent lip reading using speaker adaptive training and deep neural networks. In IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 2722–2726, 2016.
- [38]S. Gergen, S. Zeiler, A. H. Abdelaziz, R. Nickel, and D. Kolossa. Dynamic stream weighting for turbo-decoding-based audiovisual ASR. In Interspeech, pp. 2135–2139, 2016.
- [39]A. Graves and N. Jaitly. Towards end-to-end speech recognition with recurrent neural networks. In International Conference on Machine Learning, pp. 1764–1772, 2014.
- [40]O. Koller, H. Ney, and R. Bowden. Deep learning of mouth shapes for sign language. In ICCV Workshop on Assistive Computer Vision and Robotics, pp. 85–91, 2015.
- [41]J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng. Multimodal deep learning. In International Conference on Machine Learning, pp. 689–696, 2011.
- [42]S. Sadanand and J. Corso. Action bank: A high-level representation of activity in video. In CVPR, 2012.
- [43]G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler. Convolutional learning of spatio-temporal features. In ECCV, pages 140–153. Springer, 2010.
- [44]N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev. Panda: Pose aligned networks for deep attribute modeling. In CVPR, 2014.
- [45]Q. P. X. Peng, Y. Qiao and Q. Wang. Large margin dimensionality reduction for action similarity labeling. IEEE Signal Processing Letter, 2014.
- [46]C. Theriault, N. Thome, and M. Cord. Dynamic scene classification: Learning motion descriptors with slow features analysis. In CVPR, 2013.
- [47]H. Wang and C. Schmid. Action recognition with improved trajectories. In ICCV, 2013.
- [48]B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In NIPS, 2014.
- [49]I. Laptev and T. Lindeberg. Space-time interest points. In ICCV, 2003.
- [50]Q. V. Le, W. Y. Zou, S. Y. Yeung, and A. Y. Ng. Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In CVPR, 2011.
- [51]Y. LeCun and Y. Bengio. Convolutional networks for images, speech, and time-series. Brain Theory and Neural Networks, 1995.
- [52]J. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. In CVPR, 2015.
- [53]A. Jain, J. Tompson, Y. LeCun, and C. Bregler. Modeep: A deep learning

- framework using motion features for human pose estimation. In ACCV, 2014.
- [54]V. Jain, B. Bollmann, M. Richardson, D. Berger, M. Helmstaedter, Briggman, W. Denk, J. Bowden, J. Mendenhall, W. Abraham, Harris, N. Kasthuri, K. Hayworth, R. Schalek, J. Tapia, J. Lichtman, and H. Seung. Boundary learning by optimization with topological constraints. In CVPR, 2010.
- [55]S. Ji, W. Xu, M. Yang, and K. Yu. 3d convolutional neural networks for human action recognition. IEEE TPAMI, 35(1):221–231, 2013.
- [56]O. Kliper-Gross, Y. Gurovich, T. Hassner, and L. Wolf. Motion interchange patterns for action recognition in unconstrained videos. In ECCV, 2012.
- [57]O. Kliper-Gross, T. Hassner, and L. Wolf. The action similarity labeling challenge. TPAMI, 2012.
- [58]O. Kliper-Gross, T. Hassner, and L. Wolf. The one shot similarity metric learning for action recognition. In Workshop on SIMBAD, 2011.
- [59]D. B. Kris M. Kitani, Brian D. Ziebart and M. Hebert. Activity forecasting. In ECCV, 2012.
- [60]P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In Proc. ICCV VS-PETS, 2005.
- [61]J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. CoRR, abs/1411.4389, 2014.
- [62]J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In ICML, 2013.
- [63]C. Feichtenhofer, A. Pinz, and R. P. Wildes. Spacetime forests with complementary features for dynamic scene recognition. In BMVC, 2013.
- [64]Y. L. Jonathan J. Tompson, Arjun Jain and Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In Proceedings of NIPS, 2014.
- [65]S. R. Kaiming He, Xiangyu Zhang and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In Proceedings of ECCV, 2014.
- [66]A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In Proceedings of NIPS, 2012.
- [67]S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In Proceedings of CVPR, 2006.
- [68]P. M. R. Martin Koestinger, Paul Wohlhart and B. Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In Proceedings of IEEE International Workshop on Benchmarking Facial Image Analysis Technologies, 2011.
- [69]Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In Proceedings of CVPR, 2014.
- [70]A. Torralba, K. Murphy, and W. Freeman. Sharing visual features for multiclass and multiview object detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007.
- [71]J. Uijlings, K. van de Sande, T. Gevers, and A. Smeulders. Selective search for

- 
- object recognition. *International Journal of Computer Vision*, 2013.
- [72]R. Vaillant, C. Monroq, and Y. LeCun. An original approach for the localisation of objects in images. In *Proceedings of International Conference on Artificial Neural Networks*, 1993.
- [73]R. Vaillant, C. Monroq, and Y. LeCun. Original approach for the localisation of objects in images. *IEE Proc on Vision, Image, and Signal Processing*, 1994.
- [74]M. Viola and P. Viola. Fast multi-view face detection. In *Proceedings of CVPR*, 2003.
- [75]P. Viola and M. J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 2004.
- [76]P. Ekman and W. V. Friesen, "Facial action coding system: A technique for the measurement of facial movement. Palo Alto," CA: Consulting Psychologists Press. Ellsworth, PC, & Smith, CA (1988). From appraisal to emotion: Differences among unpleasant feelings. *Motivation and Emotion*, vol. 12, pp. 271-302, 1978.
- [77]W. Lirong, W. Xiaoli, and X. Jing, "Lip Detection and Tracking Using Variance Based Haar-Like Features and Kalman filter," in *Frontier of Computer Science and Technology (FCST)*, 2010 Fifth International Conference on, 2010, pp. 608-612.
- [78]P. Gang, S. Lin, W. Zhaohui, and L. Shihong, "Eyeblink-based Anti-Spoofing in Face Recognition from a Generic Webcam," in *Computer Vision*, 2007. *ICCV 2007. IEEE 11th International Conference on*, 2007, pp. 1-8.
- [79]B. Lu, Y. Sun, and L. Wang, "Dynamic Face Fatigue Detection Based on Feature-Lever Fusion," in *Multimedia Communications (Mediacom)*, 2010 International Conference on, 2010, pp. 123-126.
- [80]B. Fasel and J. Luetttin, "Automatic facial expression analysis: a survey," *Pattern Recognition*, vol. 36, pp. 259-275, 2003.
- [81]P. I. Wilson and J. Fernandez, "Facial feature detection using Haar classifiers," *Journal of Computing Sciences in Colleges*, vol. 21, pp. 127-133, 2006.
- [82]P. Minh-Tri, V. D. D. Hoang, and C. Tat-Jen, "Detection with multi-exit asymmetric boosting," in *Computer Vision and Pattern Recognition*, 2008. *CVPR 2008. IEEE Conference on*, 2008, pp. 1-8.
- [83]L. Liying and G. Weiwei, "The Face Detection Algorithm Combined Skin Color Segmentation and PCA," in *Information Engineering and Computer Science*, 2009. *ICIECS 2009. International Conference on*, 2009, pp. 1-3.
- [84]M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces," *Pattern Analysis and Machine Intelligence*, *IEEE Transactions on*, vol. 12, pp. 103-108, 1990.
- [85]B. Scassellati, "Eye finding via face detection for a foveated active vision system," in *Proceedings of the National Conference on Artificial Intelligence*, 1998, pp. 969-976.
- [86] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *European Conference on Computer Vision (ECCV)*, 2014.
- [87]K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning*

- 
- Representations (ICLR), 2015.
- [88]J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
- [89]S. Ren, K. He, R. Girshick, and J. Sun, “FasterR-CNN: Towards real-time object detection with region proposal networks,” in Neural Information Processing Systems (NIPS), 2015.
- [90]T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft COCO: Common Objects in Context,” in European Conference on Computer Vision (ECCV), 2014.
- [91]J. Johnson, A. Karpathy, and L. Fei-Fei, “Densecap: Fully convolutional localization networks for dense captioning,” arXiv:1511.07571, 2015.
- [92] J. Carreira and C. Sminchisescu, “CPMC: Automatic object segmentation using constrained parametric min-cuts,” IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2012.
- [93]P. Arbelaez, J. Pont-Tuset, J. T. Barron, F. Marques, and J. Malik, “Multiscale combinatorial grouping,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
- [94]D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, “Scalable object detection using deep neural networks,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014.
- [95]J. Dai, K. He, and J. Sun, “Convolutional feature masking for joint object and stuff segmentation,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
- [96]M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional neural networks,” in European Conference on Computer Vision (ECCV), 2014.
- [97]C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, and A. Rabinovich, “Going deeper with convolutions,” in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
- [98]A. Krizhevsky, I. Sutskever, and G. Hinton, “Imagenet classification with deep convolutional neural networks,” in Neural Information Processing Systems (NIPS), 2012.
- [99]Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” arXiv:1408.5093, 2014.
- [100]Zeiler, M., Taylor, G., and Fergus, R. Adaptive deconvolutional networks for mid and high level feature learning. In ICCV, 2011.
- [101]Hinton, G. E., Osindero, S., and The, Y. A fast learning algorithm for deep belief nets. *Neural Computation*, 18:1527-1554, 2006.
- [102]Jarrett, K., Kavukcuoglu, K., Ranzato, M., and Le-Cun, Y. What is the best multi-stage architecture for object recognition? In ICCV, 2009.
- [103]Le, Q. V., Ngiam, J., Chen, Z., Chia, D., Koh, P., and Ng, A. Y. Tiled convolutional neural networks. In NIPS, 2010.
- [104]Sande, K., Uijlings, J., Snoek, C., and Smeulders, A. Hybrid coding for



- selective search. In PASCAL VOC Classification Challenge 2012, 2012.
- [105] Yan, S., Dong, J., Chen, Q., Song, Z., Pan, Y., Xia, W., Huang, Z., Hua, Y., and Shen, S. Generalized hierarchical matching for sub-category aware object classification. In PASCAL VOC Classification Challenge 2012, 2012.
- [106] Erhan, D., Bengio, Y., Courville, A., and Vincent, P. Visualizing higher-layer features of a deep network. In Technical report, University of Montreal, 2009.
- [107] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. ImageNet large scale visual recognition challenge. CoRR, abs/1409.0575, 2014.
- [108] Zeiler, M. D. and Fergus, R. Visualizing and understanding convolutional networks. CoRR, abs/1311.2901, 2013. Published in Proc. ECCV, 2014.
- [109] Wei, Y., Xia, W., Huang, J., Ni, B., Dong, J., Zhao, Y., and Yan, S. CNN: Single-label to multi-label. CoRR, abs/1406.5726, 2014.
- [110] Oquab, M., Bottou, L., Laptev, I., and Sivic, J. Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks. In Proc. CVPR, 2014.
- [111] Perronnin, F., Sánchez, J., and Mensink, T. Improving the Fisher kernel for large-scale image classification. In Proc. ECCV, 2010.
- [112] X. Xiong, and F. Torre, “Supervised descent method and its applications to face alignment,” in IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 532-539.
- [113] Z. Zhang, P. Luo, C. C. Loy, and X. Tang, “Facial landmark detection by deep multi-task learning,” in European Conference on Computer Vision, 2014, pp. 94-108.
- [114] Z. Liu, P. Luo, X. Wang, and X. Tang, “Deep learning face attributes in the wild,” in IEEE International Conference on Computer Vision, 2015, pp. 3730-3738.
- [115] S. Yang, P. Luo, C. C. Loy, and X. Tang, “WIDER FACE: A Face Detection Benchmark”. arXiv preprint arXiv:1511.06523.
- [116] V. Jain, and E. G. Learned-Miller, “FDDB: A benchmark for face detection in unconstrained settings,” Technical Report UMCS-2010-009, University of Massachusetts, Amherst, 2010.
- [117] B. Yang, J. Yan, Z. Lei, and S. Z. Li, “Convolutional channel features,” in IEEE International Conference on Computer Vision, 2015, pp. 82-90.
- [118] S. Yang, P. Luo, C. C. Loy, and X. Tang, “From facial parts responses to face detection: A deep learning approach,” in IEEE International Conference on Computer Vision, 2015, pp. 3676-3684.
- [119] X. P. Burgos-Artizzu, P. Perona, and P. Dollar, “Robust face landmark estimation under occlusion,” in IEEE International Conference on Computer Vision, 2013, pp. 1513-1520.
- [120] X. Cao, Y. Wei, F. Wen, and J. Sun, “Face alignment by explicit shape regression,” International Journal of Computer Vision, vol 107, no. 2, pp. 177-190, 2012.
- [121] T. F. Cootes, G. J. Edwards, and C. J. Taylor, “Active appearance models,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 6, pp. 681-685, 2001.

- 
- [122]X. Yu, J. Huang, S. Zhang, W. Yan, and D. Metaxas, "Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model," in IEEE International Conference on Computer Vision, 2013, pp. 1944-1951.
- [123]P. Viola and M. J. Jones, "Robust real-time face detection. International journal of computer vision," vol. 57, no. 2, pp. 137-154, 2004
- [124]M. T. Pham, Y. Gao, V. D. D. Hoang, and T. J. Cham, "Fast polygonal integration and its application in extending haar-like features to improve object detection," in IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 942-949.
- [125]Q. Zhu, M. C. Yeh, K. T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients," in IEEE Computer Conference on Computer Vision and Pattern Recognition, 2006, pp. 1491-1498.
- [126]G. Ghiasi, and C. C. Fowlkes, "Occlusion Coherence: Detecting and Localizing Occluded Faces," arXiv preprint arXiv:1506.08347.
- [127]S. S. Farfade, M. J. Saberian, and L. J. Li, "Multi-view face detection using deep convolutional neural networks," in ACM on International Conference on Multimedia Retrieval, 2015, pp. 643-650.
- [128]M. Castrillón, O. Déniz, C. Guerra, and M. Hernández, "ENCARA2: Real-time detection of multiple faces at different resolutions in video streams," *Journal of Visual Communication and Image Representation*, vol. 18, no. 2, pp. 130-140, 2007.
- [129]J. Zhao and G. Kearney, "Classifying facial emotions by backpropagation neural networks with fuzzy inputs," in International Conference on Neural Information Processing, 1996, vol. 1, pp. 454-457.
- [130]R. Katratwar and P. Ghonge, "Emotion Analysis by Facial Feature Detection."
- [131]C. L. Lisetti and D. E. Rumelhart, "Facial Expression Recognition Using a Neural Network," in FLAIRS Conference, 1998, pp. 328-332.
- [132]L. Ma and K. Khorasani, "Facial expression recognition using constructive feedforward neural networks," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 3, pp. 1588-1595, 2004.
- [133]M. Dahmane and J. Meunier, "Emotion recognition using dynamic grid-based HoG features," in Automatic Face & Gesture Recognition and Workshops (FG2011), 2011 IEEE International Conference on, 2011, pp. 884-888: IEEE.
- [134]C. Orrite, A. Gañán, and G. Rogez, "Hog-based decision tree for facial expression classification," in Iberian Conference on Pattern Recognition and Image Analysis, 2009, pp. 176-183: Springer.
- [135]N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005, vol. 1, pp. 886-893: IEEE.
- [136]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, 2007.
- [137]G. Littlewort, M. S. Bartlett, I. Fasel, J. Susskind, and J. Movellan, "Dynamics of facial expression extracted automatically from video," *Image and Vision Computing*, vol. 24, no. 6, pp. 615-625, 2006.

- 
- [138]H.-B. Deng, L.-W. Jin, L.-X. Zhen, and J.-C. Huang, "A new facial expression recognition method based on local gabor filter bank and pca plus lda," *International Journal of Information Technology*, vol. 11, no. 11, pp. 86-96, 2005
- [139]D. Gabor, "Theory of communication. Part 1: The analysis of information," *Journal of the Institution of Electrical Engineers-Part III: Radio and Communication Engineering*, vol. 93, no. 26, pp. 429-441, 1946.
- [140]D. Datcu and L. Rothkrantz, "Facial expression recognition in still pictures and videos using active appearance models: a comparison approach," in *Proceedings of the 2007 international conference on Computer systems and technologies*, 2007, p. 112: ACM.
- [141]M. Suk and B. Prabhakaran, "Real-time mobile facial expression recognition system-a case study," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 132-137.
- [142]Y. Zhang and Q. Ji, "Active and dynamic information fusion for facial expression understanding from image sequences," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 27, no. 5, pp. 699-714, 2005.
- [143]H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 20, no. 1, pp. 23-38, 1998.
- [144]T. Sakai, M. Nagao, and S. Fujibayashi, "Line extraction and pattern detection in a photograph," *Pattern recognition*, vol. 1, no. 3, pp. 233-248, 1969.
- [145]B. Scassellati, "Eye finding via face detection for a foveated active vision system," in *AAAI/IAAI*, 1998, pp. 969-976.
- [146]W. Gao et al., "The CAS-PEAL large-scale Chinese face database and baseline evaluations," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 38, no. 1, pp. 149-161, 2008.
- [147]S. H. Yoo, D. Matsumoto, and J. A. LeRoux, "The influence of emotion recognition and emotion regulation on intercultural adjustment," *International Journal of Intercultural Relations*, vol. 30, pp. 345-363, 2006.
- [148]P. Ekman, "Emotions revealed: Recognizing faces and feelings to improve communication and emotional life," 2003.
- [150]A. Poursaberi, H. A. Noubari, M. Gavrilova, and S. N. Yanushkevich, "Gauss-Laguerre wavelet textural feature fusion with geometrical information for facial expression identification," *EURASIP Journal on Image and Video Processing*, vol. 2012, no. 1, pp. 1-13, 2012.
- [151]M. M. Ibrahim, "Video processing analysis for non-invasive fatigue detection and quantification," *University of Strathclyde*, 2014.
- [152]B. H. Mohammad Mahoor, "Facial Expression Recognition Using Enhanced Deep 3D Convolutional Neural Networks," pp. 30-40, 2017.
- [153]J. Donahue et al., "Long-term recurrent convolutional networks for visual recognition and description," pp. 2625-2634, 2015.
- [154]W.-S. Chu, F. De la Torre, and J. F. Cohn, "Learning spatial and temporal cues for multi-label facial action unit detection," *12th IEEE International Conference on Automatic Face & Gesture Recognition*, pp. 25-32, 2017

- 
- [155]A. Graves, C. Mayer, M. Wimmer, J. Schmidhuber, and B. Radig, "Facial expression recognition with recurrent neural networks," Proceedings of the International Workshop on Cognition for Technical Systems, 2008.
- [156]B. H. Mohammad Mahoor, "Facial Expression Recognition Using Enhanced Deep 3D Convolutional Neural Networks," pp. 30-40, 2017.
- [157]J. Donahue et al., "Long-term recurrent convolutional networks for visual recognition and description," pp. 2625-2634, 2015.
- [158]W.-S. Chu, F. De la Torre, and J. F. Cohn, "Learning spatial and temporal cues for multi-label facial action unit detection," 12th IEEE International Conference on Automatic Face & Gesture Recognition, pp. 25-32, 2017
- [159]A. Graves, C. Mayer, M. Wimmer, J. Schmidhuber, and B. Radig, "Facial expression recognition with recurrent neural networks," Proceedings of the International Workshop on Cognition for Technical Systems, 2008.
- [160]B. C. Ko, "A Brief Review of Facial Emotion Recognition Based on Visual Information," sensors, vol. 18, no. 2, p. 401, 2018.
- [161]S. Ebrahimi Kahou, V. Michalski, K. Konda, R. Memisevic, and C. Pal, "Recurrent neural networks for emotion recognition in video," Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, pp. 467-474: ACM, 2015.
- [162]H.-W. Ng, V. D. Nguyen, V. Vonikakis, and S. Winkler, "Deep learning for emotion recognition on small datasets using transfer learning," Proceedings of the 2015 ACM on international conference on multimodal interaction, pp. 443-449: ACM, 2015.
- [163]D. H. Kim, W. Baddar, J. Jang, and Y. M. Ro, "Multi-objective based spatio-temporal feature representation learning robust to expression intensity variations for facial expression recognition," IEEE Transactions on Affective Computing, 2017.
- [164]A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," Neural Networks, vol. 18, no. 5-6, pp. 602-610, 2005.
- [165]M. Cannici, M. Ciccone, A. Romanoni, and M. Matteucci, "Event-based Convolutional Networks for Object Detection in Neuromorphic Cameras," arXiv preprint arXiv:1805.07931, 2018.
- [166]S. R. Livingstone and F. A. Russo, "The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English," PloS one, vol. 13, no. 5, p. e0196391, 2018.
- [167]J. Jeon et al., "A Real-time Facial Expression Recognizer using Deep Neural Network," Proceedings of the 10th International Conference on Ubiquitous Information Management and Communication, p. 94: ACM, 2016
- [168]A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," Advances in neural information processing systems, pp. 1097-1105, 2012.
- [169]C. Szegedy et al., "Going deeper with convolutions," Proceedings of the IEEE

conference on computer vision and pattern recognition, pp. 1-9, 2015.

[170]K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016.

[171]H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," IEEE Transactions on pattern analysis and machine intelligence, vol. 20, no. 1, pp. 23-38, 1998.