The University of Hong Kong

Final Year Project

Facial Expressions Recognition and Synthesis Using Generative Adversarial Network

Intermediate Report

by

Lo Man Chun

Department of Computer Science

Supervised by

Dr. K. P. Chan

Department of Computer Science

Jan 30th, 2020

Abstract

Facial Expression Recognition is the process of identifying human emotion by classifying human faces images. It is a complicated image classification problem in the field of Computer Vision. There are many training algorithms used in Facial Expression Recognition such as Multiclass Support Vector Machines (SVM), Convolutional Neural Networks (CNN), Recurrent Neural Networks and Convolutional Long Short-Term Memory (ConvLSTM). Each method has its benefits and drawbacks and may be applicable to different situations. In this project, facial expression recognition will be done on same person in a video clip which contains frames of image. To assist the whole process, ConvLSTM, which is the combination of CNN and RNN, is used. ConvLSTM includes all benefits of CNN which can better distinguish two similar emotions, and the strength of RNN which can make the previous frame of image as an extra information. One of the biggest challenges in this project is that a there are insufficient sample size of high-quality public dataset in the Internet. In addition, Different head poses, and illumination make great differences in the model. In the project, we make use of Generative Adversarial Network, which consists of a generative network on the facial expression synthesis, to provide high-quality data to the discriminative network. Currently, the project is at the testing stage of the related model. Project details and current deliverable can be found through: i.cs.hku.hk/fyp/2019/fyp19005/.

Acknowledgments

Conducted by Lo Man Chun and Wang HanYan, this research project is supported by Department of Computer Science, The University of Hong Kong. We thank Dr. K. P. Chan for supervising us and providing valuable suggestions. The project would not be on the right track without his help.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

FER	Facial Expression Recognition.
SVM	Multiclass Support Vector Machines.
CNN	Convolutional Neural Networks.
RNN	Recurrent Neural Networks.
ConvLSTM	Convolutional Long Short-Term Memory.
GAN	Generative Adversarial Network.

1 Introduction

Facial Expression Recognition is a process of distinguishing human facial expressions through computer vision, which discovers the numerical patterns from the matrix of pixel values in image. Facial Expression Recognition is an image classification problem, which necessitates the labelling of the given image with a single facial expression from the following categories: happiness, sadness, anger, fear, disgust, surprise and neutral.

The existing FER systems performs well in the laboratory, which means high-quality and controlled datasets are used. In high-quality images, faces have similar head poses and illumination is constant. In addition, the facial expressions are acted out, which are more exaggerated than those of normal face. When some methods are applied using Cohn-Kanade dataset, which contains less than 500 high-quality facial expressions samples, the accuracy of all systems exceeds 90% [1]. However, the models are overfitting, meaning the training accuracy is higher than the validation accuracy, because they are less accurate when predicting the emotions of real images.

1.1 Background

1.1.1 Generative Adversarial Network

Generative Adversarial Network is a class of machine learning made by Ian Goodfellow and his teammates in 2014[2]. Two neutral networks can be trained by contesting each other without supervision, or with semi-supervision[3], fully supervision[4] and reinforcement learning[5].

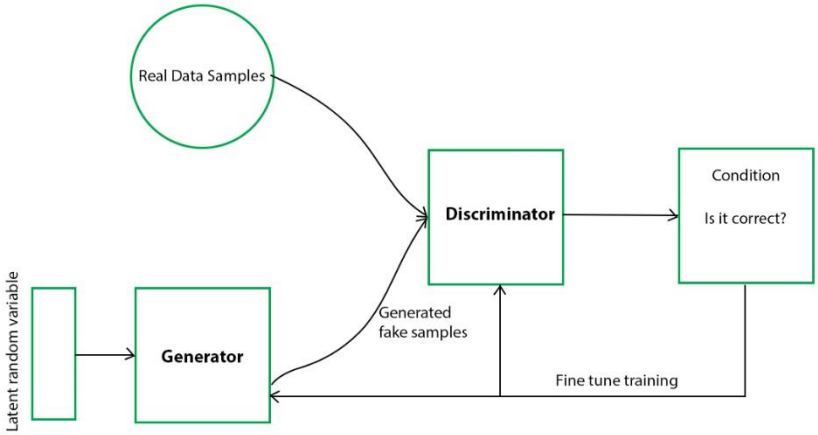

Figure 1.1: Flow chart of Generative Adversarial Network[6]

Fig. 1.1 shows the main components of GAN are the generator and the discriminator. The objective of generator is to create fake sample as real as possible, to fool the discriminator. The discriminator needs to distinguish whether the received samples are real or not. The real data samples and fake samples are randomly chosen and pass to the discriminator. The answer of discriminator is checked then a response is passed to both discriminator and generator, which allows them to improve their own algorithm.

Figure 1.2: Images from website ‘This person does not exist’[7]

GAN can be applied in various aspects, such as video games, fashion and science. It is commonly used to generate faces that do not exist. In the website ‘This person does not exist’ [7], images of human faces are randomly generated when you enter the website. In

Fig. 1.2, we can see that the faces are very realistic but only some details on the edge of the face and on the background are slightly strange, giving us evidence that some of them are made by GAN.

1.2 Motivation

Facial expression recognition has been active area of research for some decades, the first Automatic Facial Expression Recognition and Analysis was done in 2011[8], which is about AU detection and discrete emotion detection. Recently, many methods are invented such as CNN, RNN, which greatly increase the accuracy of emotion detection. Emotion detection is an ability that every person knows, but it is hard when we want to achieve it through computer. Some of the neural network are trying to act similar as human brain, which is impossible because human brains are too complicated. But through the invention of FER, we can know more about how people detect facial expression, for example, people detect the facial expression mainly by the shape such as eyebrow, month, face. This attract us to work on this topic.

1.3 Objective, Scope and Deliverables

Therefore, in this project, we propose ConvLSTM to be the training algorithm. ConvLSTM is a combination of CNN and RNN, come with their own benefits. CNN use kernel to find the pattern of smaller part of the image and RNN can make use of the previous frame and give extra information for FER system to judge the facial expression. It will better fit our project which need to distinguish the facial expression of a face continuously in a video clip.

The scope of this project includes:

i) Finding a pre-trained FER model and do testing

ii) Reviewing the algorithm used, for example the pre-processing part, the layers of CNN

iii) Selecting a good dataset which have constant illumination, head poses etc.

iv) Evaluating the result, adjusting the algorithm and running it again until the result is satisfied

By the end of this project, we will deliver:

i) The trained FER model with ConvLSTM as training algorithm;

ii) Video demonstration of our trained FER model;

iii) Detailed report on the project.

The project progress can be checked on the website: i.cs.hku.hk/fyp/2019/fyp19005/.

1.4 Significance and Contribution

Facial Expression Recognition can be widely applied in different situations to facilitates the user experiences. Our system can monitor user's dynamic facial expression. Therefore, it is useful for collecting business data or giving fast response according to user's emotions.

Some industries have already started to make use of FER to create new and exciting experience for their customers. For example, Riot is an emotionally responsive, live-action film using FER technology [9]. Webcam is used to capture audience's facial expressions during the movie, characters inside the movie will respond according to the emotions of audience. For example, when the audience feels agitated, the character responds defensively[9]. This allows the audience to involve in the movie and creates a new experience to them.

In the gaming industry, in video game testing phase, FER can be used to evaluate user experience. The facial expression of users will be recorded by the camera during they are testing the game. The data are critical for game designers to evaluate the game. A graph

of users' emotional experience can be drawn, which clearly show which part of the game need to be improved.

1.5 Report Outline

The report structure is mentioned as below: Section 2 is about the methodology applied in this project, which includes different training algorithms. In this section, the training algorithm we used is introduced and detail explanation of it is provided. We also discuss some training algorithms and stating why the training algorithm we have chosen is the most suitable in this situation. Section 3 is the current status of the project, covers what we have done and future plans. In this section, you can see our plans and schedules, which can see the progress clearly. Also, we will talk about the limitations and difficulties we faced in the progress, and the solutions we have thought. Section 4 gives a conclusion to the project progress report, which about the solution of the project and how the project can contribute to the world. This explains the importance of our project and our motivation on it.

2 Methodology

This section describes the methodology which planned to adopt in the projects. In October, research was done on related work and methodology was set up for this project. Below is the methodology we decided after considering different systems and algorithms.

2.1 Training Algorithms

Currently, there are many training algorithms which can be used for training FER model, each of them has its own benefits and drawbacks. In this project, we have chosen to use ConvLSTM (see section 2.2.3) as the training algorithm which is a combination of CNN (see section 2.2.1) and RNN (see section 2.2.2). This section will briefly introduce CNN, RNN and LSTM, which helps to explain the concept of ConvLSTM.

2.1.1 Convolutional Neural Networks (CNN)

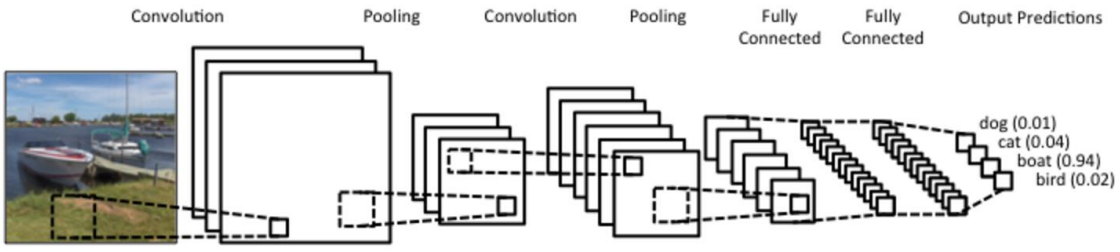

Figure 2.1: Concept map of Convolutional Neural Networks[15]

Convolutional Neural Network is one of the neural networks, which is commonly used in image classification. The structure of CNN contains many layers (see figure2.1) which can filter the image and extract the pattern of smaller part of image, such as the shape of eyebrow. There are mainly three types of layers, which are convolution layer, pooling layer and fully connected layer. The benefits of CNN is that it fits to the situation when we need to extract the detailed face characteristics such as the shape of eye brows, eyes,

smiling faces. But the limitation is that it is not suitable for face recognition in video, because the facial expressions keep changing in video and CNN can hardly recognize the not completed facial expression if we only consider one of the frames of the video.

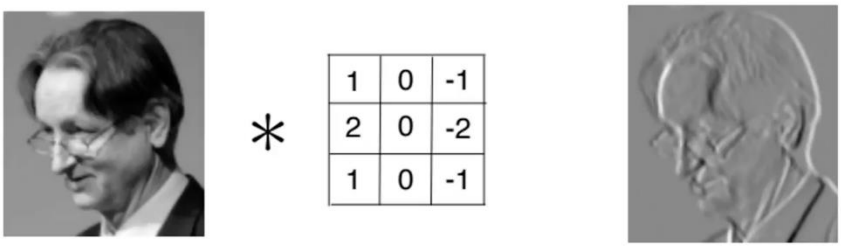

Figure 2.2: Convolutional layer[15]

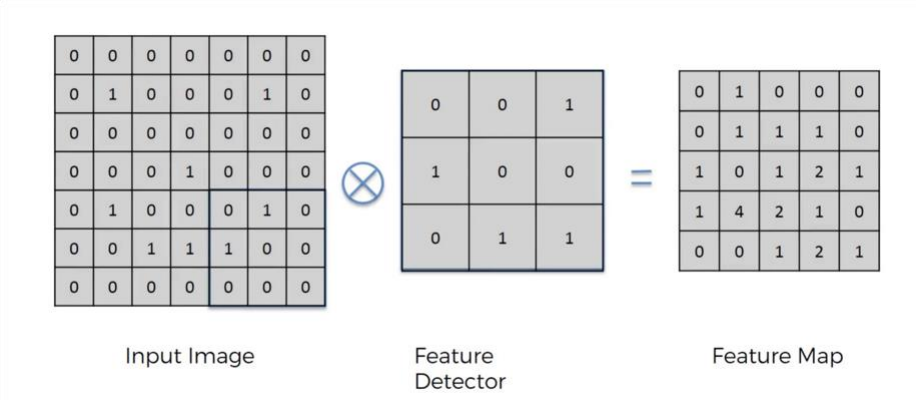

Figure 2.3: Extracting shape of object by kernel[15]

Convolutional layer is the core block in CNN. In convolutional layer, different kernels (feature detector) are computed with the input image's matrix of pixel value to extract different feature of image (see figure 2.2), such as the shape (see figure 2.3). After feature maps are created, they will be passed to the next layer, which is the pooling layer.

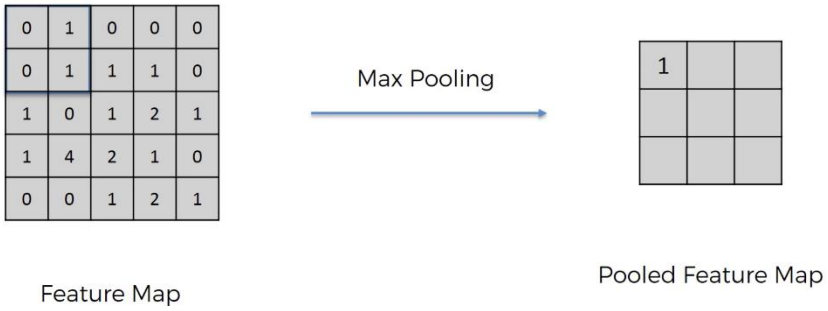

Figure 2.4: Max pooling[15]

Basically, pooling layer is a form of non-linear down-sampling. Max pooling (see figure 2.4) is commonly used to consider the input image into numbers of non-overlapping rectangles, then take the maximum value of each sub-region. Max pooling allows the system to make the same judgement if the whole image moves away by several pixels, and it has good anti-noise ability. However, there are some drawbacks when we only keep the maximum value of pixels. First, the location information of the pixels is lost, we only know the relative location of the pixels comparing to other pixels. Second, the appearance frequency of pixels is lost, even when a value appears several times, we only keep the maximum value, so we do not have the information of frequency of pixels when it may be useful in some cases.

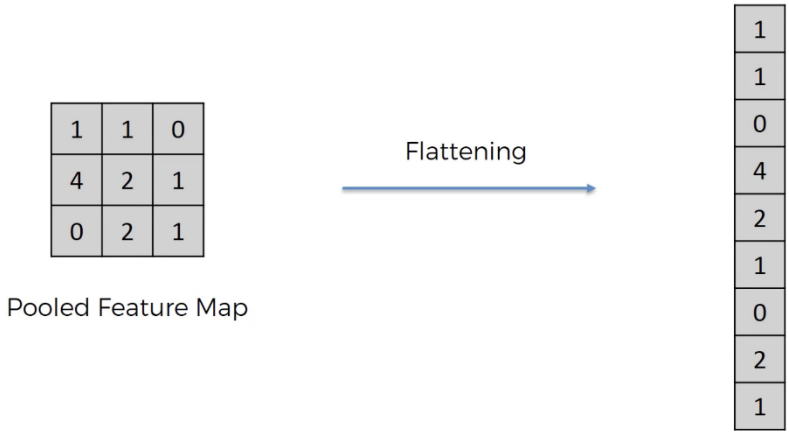

Figure 2.5: Flattening a pooled feature map[15]

Fully connected layer can generate flattened output of feature maps. High-level reasoning

is to be done in this layer. Those pooled feature maps will be flattened (see figure 2.5) and connected to the neural network (see figure 2.6). It is essential to change matrix results in convolutional layer and pooling layer into one-dimensional arrays so it can connect with the neural network we had. When the neural network cannot take matrix as input, we have to flatten the result before passing the data.

2.1.2 Recurrent Neural Networks (RNN)

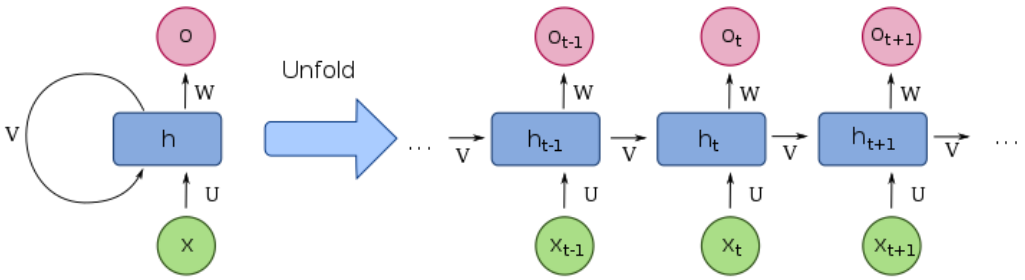

Figure 2.6: Diagram for a one-unit recurrent neural network (RNN). From bottom to top : input state, hidden state, output state. U, V, W are the weights of the network. Compressed diagram on the left and the unfold version of it on the right.

Recurrent Neural Networks classify images using dynamic temporal behavior, which can use the internal states (memory) to process the inputs. For example, in image classification of a video clip, if the previous frame of facial expression of actor is happiness, the next facial expression will be more likely to be happiness and smaller chance to be sadness. RNN is suitable for continuous action such as unsegmented, connected handwriting recognitions[12] and speech recognition[13]. In figure 2.6, we can see the hidden state h_t, which is the memory of RNN, is affected by h_{t-1}, x_t and those weights variables. The detailed formula is as below:

$$h_t = \mathrm{V} * h_{t-1} + Ux_t + b \qquad (1)$$

Here will talk more about the equation (1), h_t is the hidden layer vector, which is the essential factor in the equation, its value is affected by the previous h_{t-1}, x_t and those weights variables. A function is applied to h_t to get the output result. *V*, *U*, *W* and *b* are the

parameter matrices and vector, which are weight variables applied to different input value. We can adjust the value of those variables according to the importance of different input value. x_t is the input vector, which is the input received by the systems, such as the pixels value of the current frame.

Then, the output y_t is calculated by:

$$o_t = g(W * h_t) \quad (2)$$

In equation (2), g stands for the activation function to the hidden layer vector. It is a function used to define the output of a node from the hidden layer vector. In the equation, W is used as the weight variable and it is adjustable according to different situations.

There are many variations of RNN which have different numbers of inputs and outputs, such as one-to-one, one-to-many, many-to-one and many-to-many. Each of them is suitable for different situations, for example, one-to-many has one input and many outputs, which is suitable for image captioning – input a video and detect several objects inside the video. One-to-many RNN is also called 'sequence output'.

2.1.3 Long Short-Term Memory (LSTM)

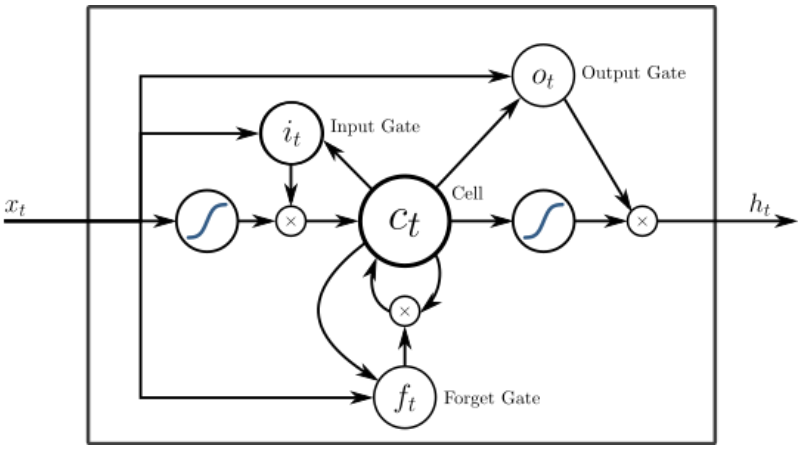

Figure 2.7: Structure of LSTM

Long Short-Term Memory is a variation of RNN, which adds three gates to the hidden layer. The hidden layer contains three gates, which are input gate, output gate and forget gate (see figure 2.7). The gates can control the flow of information both into and out of the cell and decide what information will keep in the long-term memory.

$$\boldsymbol{h_t = V * h_{t-1} + Ux_t + b}$$
$$\boldsymbol{h_t = (V^2 * h_{t-2} + V * Ux_{t-1} + V * b) + Ux_t + b}$$
……
$$\boldsymbol{h_t = (V^t * h_0 + \cdots) + Ux_t + b}$$

Figure 2.8: Iteration of h_t

Long Short-Term Memory was invented because RNN have some limitations. We can see the equation in Fig. 2.8, $\boldsymbol{h_t}$ is calculated according to the weight V. When V should be smaller than one, which means the closer factor have greater effect to the result. However, $\boldsymbol{V^t}$ is approaching to 0 when t become greater. When $\boldsymbol{V^t}$ = 0, it means the factor have no effect to the result anymore, the system 'forgets' the factor. However, in real life example, the past factor should still have effect on the judgement, so LSTM was invented.

Recently, some researchers try to apply LSTM in developing Artificial General Intelligence. In 2019, a deep LSTM core is used by AlphaStar, program of DeepMind, to excel a famous and complex video game StarCraft[14]. This was considered as a great improvement in developing Artificial General Intelligence.

2.1.4 Convolutional Long Short-Term Memory (ConvLSTM)

Convolutional Long Short-Term Memory is the combination of CNN and RNN. ConvLSTM have both CNN's strength on extracting patterns and RNN's strength on dynamic temporal behavior. The system uses convolutional layers to find features and LSTM to considers changes in different frames of video. The performance and accuracy will be better than using either CNN or RNN, especially in our project. We need to capture the feature in smaller image such as the shape of eyebrow, and also, we need to

consider the facial expression in the frames before.

2.2 Evaluation

After the testing of system, we will evaluate the accuracy of facial expression recognition system. We will adjust the algorithm inside the system to increase the accuracy. If the result is satisfactory when using high-quality dataset, we will try to run it with other public dataset which similar to the reality.

2.3 Summary

In this chapter, different training algorithms are discussed with simple introductions and compared with theirs benefits and drawbacks. ConvLSTM is found the most suitable for our purpose. Since we want to recognize facial expressions in video, ConvLSTM can consider the frames before and also the detail characteristics of facial expressions, combining the benefits from CNN and RNN, it should work better in video.

3 Current Status

This section is about the updated project schedule and the current status of this project. Section 3.1. presents the project progress; Section 3.2 presents the next step of the project;

3.1 Schedule and Progress

The project is running as planned. We can see the project schedule below(Table 3.1), we mark the status of task next to the task description. All of the tasks in September and October are either done or in progress.

In September, we met with our supervisor Dr K. P. Chan for the project scope and project plan and received advices from him. Then we dated Dr Chan to have a short meeting weekly so that he can always follow our progress and give suggestions to solve our difficulties. We successfully delivered the project plan and project website as discussed with Dr Chan.

In October, we conducted research on different training algorithm and find a pre-trained FER system for testing. We received the advice from Dr Chan about the public dataset and FER system in the Internet. Dataset CK+ is recommended by Dr Chan. We tested on a pre-trained FER system called Emopy, which is an open source system focusing in using public dataset.

In November, we read the source code of FER system, which made us know more about how the system recognize facial expression and that helping us in the project. We looked into the neural networks in the system.

In December and January, we tried to evaluate on the algorithm and used different datasets. Because we have found that the accuracy was lower than our expectation, so we

have figured out these two methods to improve it. We will keep doing it in February and try to improve the system.

Time period	Tasks
September	• Meet with supervisor (done) • Write a detailed project plan (done) • Develop project website (done)
October	• Research on different training algorithm (done) • Select public dataset from the Internet (done) • Test on existing FER system (done)
November	• Dig into the source code of FER system(done)
December - February	• Evaluation of the algorithm to improve accuracy (in progress) • Try with different datasets (in progress)
March - April	• Write a final report • Preparation for final presentation

Table 3.1: Project Schedule

3.2 Limitation and difficulties

According to the schedule plan (Table 3.1), evaluation of the algorithm should be started. However, the accuracy of the FER system was less than 50% when I do the testing with the dataset provided. I thought this related to the quality and quantity of the provided dataset, so CK+ dataset was adopted, which is known as one of the high-quality free datasets in the Internet. After that, the testing accuracy remained lower than my

expectation. Now there are two possible methods to solve this, one way is to evaluate the training algorithm I am using or choose another training algorithm, another way is to find another dataset. However, there are probably no better datasets with continuous frames in the Internet, I may choose a dataset with only the climax image, and this make us need to change the training algorithm too because ConvLSTM only apply to continuous images.

3.3 Next Step

After we finish the debug of FER system, we will continue evaluating the source code of the FER system, as planned in December, to increase the accuracy by changing the algorithm or using different datasets. Researches will be done for finding better solution of it.

In the January, we will start to prepare for the first presentation to the department. Clear and attractive PowerPoint will be done for the explanation of my progress and the topic.

4 Conclusion

Facial expression recognition has been developed for over 3 decades. Many powerful algorithms are introduced such as CNN, RNN, ConvLSTM, to increase the accuracy. In addition, different algorithm focuses on different problem, for example CNN focus on extracting local data from an image, RNN focus on using previous results to help the recognition on next image, ConvLSTM combines both advantages of CNN and RNN. However, the quality of dataset is still one of the greatest challenges in developing a well-trained neural network. We hope that through the Generative Adversarial Network, we can generate more high-quality samples for the FER system, which can increase the accuracy of system. Also, the high-quality samples usually differ from the images we have in the real life, we hope that the contest of two networks, the work on pre-processing function, the training algorithm can solve the problem. Most of the systems we can found in the Internet have significance difference between training accuracy and validation accuracy, but not many of them have tried combining GAN with FER system. The development of FER system is now at the bottleneck, many people want to add FER in their system for data collection or immediate respond according to user emotion, but the accuracy is not enough for them to bring their ideas into reality. It is believed that when the FER technology is mature, many creative and amazing applications with FER will be developed and bring a different and new experience to human.

References

[1] Kulkarni, K., Corneanu, C., Ofodile, I., Escalera, S., Bar, X., Hyniewska, S., Allik, J., Anbarjafari, G. (2018). Automatic recognition of facial displays of unfelt emotions.

[2] Goodfellow, Ian; Pouget-Abadie, Jean; Mirza, Mehdi; Xu, Bing; Warde-Farley, David; Ozair, Sherjil; Courville, Aaron; Bengio, Yoshua (2014). Generative Adversarial Networks (PDF). Proceedings of the International Conference on Neural Information Processing Systems (NIPS 2014). pp. 2672–2680.

[3] Salimans, Tim; Goodfellow, Ian; Zaremba, Wojciech; Cheung, Vicki; Radford, Alec; Chen, Xi (2016).Improved Techniques for Training GANs

[4] Isola, Phillip; Zhu, Jun-Yan; Zhou, Tinghui; Efros, Alexei (2017). Image-to-Image Translation with Conditional Adversarial Nets". Computer Vision and Pattern Recognition

[5] Ho, Jonathon; Ermon, Stefano (2016). Generative Adversarial Imitation Learning

[6] Rahul_Roy. (2019). Generative Adversarial Network (GAN). Retrieved from https://www.geeksforgeeks.org/generative-adversarial-network-gan

[7] This Person Does Not Exist. (n.d.). Retrieved from https://thispersondoesnotexist.com/.

[8] Valstar, Michel & Jiang, Bihan & Mehu, Marc & Pantic, Maja & Scherer, Klaus. (2011). The first facial expression recognition and analysis challenge.

[9] RIOT. (n.d.). Retrieved from https://thoughtworksarts.io/projects/riot/

[10] Thoughtworksarts. (2019). EmoPy. Retrieved from https://github.com/thoughtworksarts/EmoPy.

[11] Perez, A. (2018). Recognizing human facial expressions with machine learning. Retrieved October 22, 2019, from https://www.thoughtworks.com/insights/articles/recognizing-human-facial-expressions-machine-learning.

[12] Graves, A.; Liwicki, M.; Fernandez, S.; Bertolami, R.; Bunke, H.; Schmidhuber, J. (2009). A Novel Connectionist System for Improved Unconstrained Handwriting Recognition

[13] Sak, Hasim; Senior, Andrew; Beaufays, Francoise (2014).Long Short-Term Memory recurrent neural network architectures for large scale acoustic modelin

[14] Stanford, Stacy (2019). DeepMind's AI, AlphaStar Showcases Significant Progress Towards AGI

[15] James, Y. (2018). [資料分析&機器學習] 第5.1講: 卷積神經網絡介紹(Convolutional Neural Network). Retrieved from https://medium.com/jameslearningnote/資料分析-機器學習-第5-1講-卷積神經網絡介紹-convolutional-neural-network-4f8249d65d4f.