Facial Emotion Recognition In Images and Videos

Fubao Wu Rui Pan

Abstract

Human expression recognition has important applications in mental analysis, behavior analysis etc. for humanization of artificial intelligence systems. It is challenging for a machine to detect and understand facial emotions. Previous semantic facial feature analysis techniques mostly suffer from high computation time and currently deep learning techniques using CNN improve a lot. State-of-the-art methods explore mainly human expression in static images recently and have average test accuracy around 60% in several images datasets. The more challenging goal is detecting facial emotions in videos. In this case, we found the test accuracy in many video datasets are less than 40%. For this purpose, we propose an emotion recognition system by using existing CNN models as features. We explore human recognition system to identify 7 types of emotions by using FER2013 dataset. Furthermore, we utilize recently proposed state-of-the-art preprocessing techniques(a modified VGG model) on facial images for manually crafting features and then use them as a CNN pretrained model features. In addition to the importance of feature extraction, we choose Long Short-Term Memory(LSTM) networks and using two datasets(JAFFE and AFEW) to train it. Finally, we use this model to test on image datasets and video datasets to compare the performance and get relative high accuracy with CNN pretrained model features.

1. Introduction

Humanization of artificial intelligence system is the important goal of artificial intelligence. Understanding the human mental states by facial emotion is an important task for completing this. Through the facial emotions, we can make artificial intelligence system better to tackle more complex tasks, such as mental analysis and user behavior analysis from their facial emotions. One specific application in medical scenes is that the medical robots can better interact with patients by understanding theirs facial expression to make predictions on the illness and mental states of patients, and recommend the corresponding activities and thus greatly reduce the work of doctors.

Most of conventional methods [9] focused on facial feature extractions of different expressions. For example, they try to get different deformation in wrinkles or deformations in eye, mouth, nose etc to identify features of emotions. Then they apply some common classifier like SVM and Nearest neighbor to classify the emotions. Recently, with the advent of deep learning, facial expression has been explored in deep learning techniques. Gudi [5] uses deep learning to recognize automatically the facial features like emotions, age and gender etc. Recently, publicly available models such as VGG_S [2] for emotional recognition and RNN models [3, 4] are applied. Also, Justus [10] advanced pre-processing algorithm for facial images and a transfer learning mechanism for Human-Computer Interaction application. Extensive work have already been done in the static images in which they basically use the recurrent neural network to train and test the static images. For example, the previous facial emotion competition on Kaggle have already gained lots of attentions and publications of some works. In this paper, we want to focus on the task of on the task of facial emotion recognition in images and videos using a existing pretrained model on CNN and the use LSTM to test how it behaves well on images and look at the testing result.

We propose to use transferring learning technique and then use Long short term memory (LSTM) to train on images to identify emotions, and then test the effect on images and videos. There are 7 common facial emotions: happy, angry, neutral, surprised, sad, fearful, disgusted. We utilized the preprocessing techniques [10] and then apply the model on modified Gudi/VGG_S as input to a LSTM network.an We will look at the CNN model effect after Gudi/VGG_S network and then examine the effect after using the LSTM model. We then apply the result of LSTM to identify emotions in a video scene.

2. Related Work

Recently, a various of neural network architectures have been utilized to tackle facial emotion recognition problem. Such as CNN, Deep Belief Networks, Very Deep Convolutional neural network and LSTM models.

Facial Action Coding System(FACS) is used to be a foundation of conventional and commercial facial analyzing. FACS's architecture is based on Active Appearance

Model which is using PCA directly on pre-processed pixels.Furthermore, it is encoded as the deviation of a face from the average face, The model is often used to classify emotions by using a single layered neural network.

A significant improvement was made by Le et al[12] Their results showed that the detector can be sensitive to other non-target high-level categories. In their study, the importance of pooling, rectification and contrast normalization was mentioned. Furthermore, Hinton and Srivastava[7] demonstrate further improvements in training by adding dropout layers. In 2013, Sermanet et al[11]. demonstrated a better solution to detection, localization and classification by using Deep Convolutional Neural Network. Baccouche et al[1]. offered the use of 3-D CNN in combination with Recurrent Neural Network for human-action classification in videos which is using spatial as well as temporal information to generate state-of-the-art results.

The accuracy of emotion recognition has highly increased by taking advantage of DNN. Han et al. built DNN model by feeding hand-crafted features and used it to detect speech emotions. Lim and Trigeorgis[14] chose a similar way to extract high-level features by using CNN. In order to capture the temporal structure, they used LSTM network to capture the contextual information in the data. Some researchers try to combine audio and visual modalities to detect emotion changes. Zhang et al[6]. used multimodal CNN for classifying emotions. That model is trained in two phases. First, two CNNs are pre-trained on large image datasets and fine-tuned to perform emotion recognition. The second phase a DNN was built that comprised of a number of fully-connected layers and the input is the extracted features of CNN. Panagiotis et al[15]. used a similar way to deal with this problem. They utilize CNN to extract features from a speech, while for the visual modality a deep residual network of 50 layers. Then, they used LSTM which is able to model the context and ensure the significant of feature extraction. The system is trained in an end-to-end fashion by taking advantage of the correlations of the each of streams. In our paper, we want to use LSTM network to train on existing CNN model to observe the effects.

CNN have recently enjoyed a great success in large-scale image and video recognition, more attempts have been made to improve the original architecture in recent years. In 2014, Sermanet et al. utilized smaller receptive window size and smaller stride of the first convolutional layer. In Howard's paper, he addressed another great opinion to change the depth in CNN architecture design. Based on this, Karen and Andrew[13] came up with a new idea to use Very Deep Convolutional Networks to improve the accuracy. They increase the depth through evaluation of networks and change the size of convolutional filters smaller to 3*3. This model performs well on several datasets and achieve state-of-the-art results. In our paper, we utilize the advantage of existing pretrained model as a feature extractor to train a LSTM network.

3. Problem Statement

Our problem includes two parts, one is identify emotion in images. The second part is to apply them on real time video scenes. The facial emotion recognition in images are trying to identify one of the 7 facial emotions: happy, angry, neutral, surprised, sad, fearful, disgusted. For images part, given a lots of facial images with different test data, how to train a neural network to effectively identify the features representing the facial emotions. Then we apply them to a series of unseen test images. For recognition in video, we try to utilize the existing result of facial detection in a video/movie, then try to identify the sequence of images with frame to identify the emotion with a several frames.

4. Methodology

Considering the success of CNN modeling on images tasks like images classification, segmentation, localization etc. We consider using a CNN network based on the work of Gudi[5] VGG-s[2]. The purpose of our project is recognizing 7 emotions, so this architecture is a great starting point and a good pre-trained model. Furthermore, we want to get a good result on videos' dataset, we choose to send the output of CNN network to a LSTM model which was proposed by Samira[3].

First of all, we use state-of-art preprocessing technique [10] the input images, the formal of input images of JAFFE is about 128*128 gray scale pixels arranged in a 2D matrix. For each image, we use face detection algorithm to normalize the face location and make sure the distance between two eyes are constant. In addition, we subtract the local mean pf pixel values from images and subtract the global mean of pixels at that location.

The original layer of this network contains an input layer of only $48 * 48$ input image data. This layer is followed by a convolutional layer with a kernel size of 5*5 having a stride of 1 both dimensions. After a local contrast layer and a max-pooling layer, two more convolutional layers are built to handle the data. The size of convolutional layers is 5*5 and 4*4. This model is finished with one fully connected layer with dropout function and one soft-max output layer. RELU was applied after all layers in this network.

Currently, the loss function we want to add one more max pooling layer to reduce the number of parameters. For normalization, we want to try batch normalization that we learned in the class.

Our purpose is trying to get a good accuracy about detect facial emotion in videos, so, we are trying to use LSTM to train videos' dataset.

For example, We will use several frames images span-

ning in 0.5 or 1 second in the video to analyze facial emotion. RNN can detect the event such as the presence of a particular expression, irrespective of the time, at which it occurs in a sequence. The input data of RNN is the output of CNN model we used before. RNN is a type of neural network which can transform a sequence inputs to a sequence of outputs. In our project, the architecture of CNN pre-trained model and later LSTM trained model is shown in figure 1.

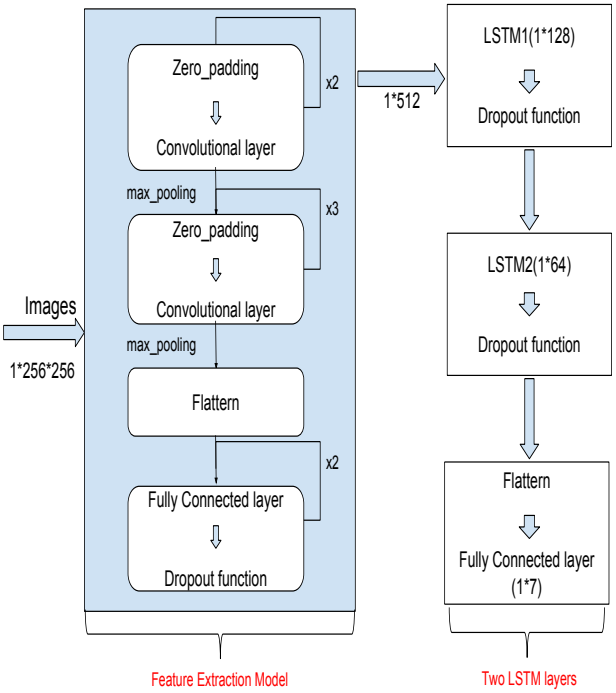


Figure 1: Architecture Design

In this paper, we build our model based on the architectures mentioned before in related work. Our model contains two parts:1. feature extraction model to detect. 2.LSTM to classify For the first part of model, we used AFER datasets to train it to set the parameters. We use this pre-trained model to extract facile feature to improve the accuracy. Since we want to capture contextual information in video data, we choose to design LSTM to classify emotions. The results with and without pre-trained model will be shown in the result section. The input to our Convolotional Neural Networks(pre-trained) is modified for inputing 256∗256 gray images. We preprocessed the original data by subtracting the mean value from each pixel. After preprocessing, the images are passed through a number of convolutional layers. Inspired by VGG model, we choose to use smaller filters with a small receptive field as 3*3 and the convolution stride is 1 pixel. We also tried 1*1 and 2*2 filters, but the result is not improved. 3*3 may be the smallest size to capture the notion of all directions. Although we use 'x2' or 'x3' to represent the round, number of filters is totally different in three parts. In the first round, we set 32, 64 and 128 layers in three rounds. In order to make the layers non-linear, we add RELU function after all hidden layers. Max-pooling is performed over a 2*2 pixel window with stride 2. We don't add normalization function in this model since the performance does not improve when adding it, but the time of memory consumption and computation time increase. For LSTM layers, we use two datasets(JAFFE and CK+2) to pre-train it. The input of LSTM is the output of feature extraction model. We design two LSTM layers here to improve the performance. Input shape is (1*1*512), after the first LSTM layer, the output is (1*1*128). Dropout function is added after every LSTM layer,. When this output go through the second LSTM layer, it becomes (1*1*64), not only dropout function, but also flatten function is added to control the dimension. Finally, the output is sent to the final Fully Connected layer to get a 1*7 outputs which means 7 different labels of emotions.

5. Experimental evaluations

5.1. Dataset

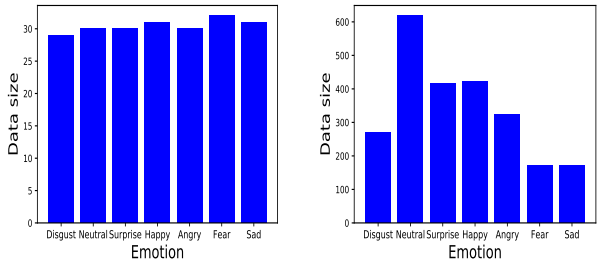
In this paper, we use the following datasets to evaluation **JAFFE:** JAFFE [] contains 219 images of 10 Japanese females with relative large dimensions. However, it has a limited number of samples, subjects and has been created in a lab controlled environment trained dataset. We use them for train to get a pretrained model with CNN.

CK+2: CK+ [8] is the current second version of the Cohn-Kanade AU-Coded Facial Expression Database for research in automatic facial image analysis and synthesis and for perceptual studies. It is publicly available and has large labeled facial emotions of images than JAFFE, and provides protocols and baseline results for facial feature tracking and action unit and emotion recognition. We use them for train and test. The expected result on images can achieve competitive model than the baseline method.

Video dataset Wild (AFEW) 5.0 dataset also has video data labeled: It contains short video clips extracted from Hollywood movies. The video clips present emotions with a high degree of variation, e.g. actor identity, age, pose and lighting conditions. The dataset contains 723 videos for training, 383 for validation and 539 test clips. Existing work have train accuracy about 80%, test accuracy 40%. We expect to get competitive result on this dataset. If time is allowed, we will use this dataset to train on our model. Currently plan is to use the video for only testing purpose.

Figure 2a denotes the dataset distributions of JAFFE dataset. It has small data size–231 labeled images but very well balanced distributions of 7 emotions.

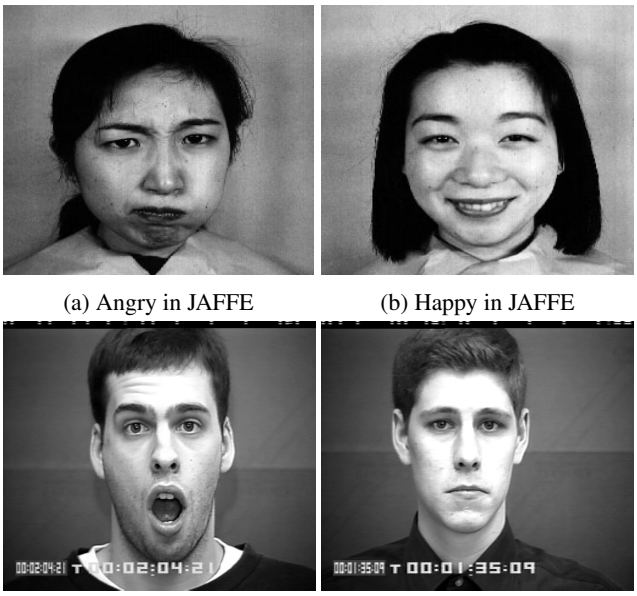
Figure 2b shows the dataset distribution of the CK+ dataset. This dataset has only very large dataset, we extracted the well-labeled 2746 images. The dataset is not very balanced, with large portion being "neutral", "happy" and "surprise" emotions.



(a) JAFFE (b) CK+

Figure 2: Data sub-class distribution

Figure 3 shows the data samples in the JAFFE and CK+ datasets. Figure 3(a), 3(b) are two JAFFE image data and their corresponding emotions. Figure 3(c), 3(d) are two CK image data and their corresponding emotions.



(a) Angry in JAFFE (b) Happy in JAFFE

(c) Surprise in CK+ (d) Sad in CK+

Figure 3: Dataset samples

5.2. Set up

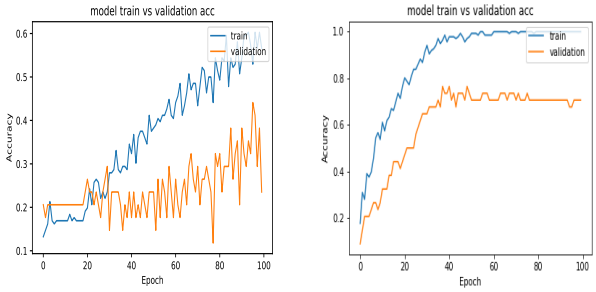
In this paper, we use precision, recall, f1-score, and accuracy to show the test result. We measure the JAFFE and CK+ images respectively. The training and validation data are 80% of the whole data of each database. Among them, validation is used to The test data is randomly kept from 20% of the whole dataset of each database.

We compared these results between LSTM model with and without the pretrained CNN model feature extractor. "without the pretrained CNN model" means the input are raw images for LSTM without the CNN features. "with the pretrained CNN model" means the input images first use CNN pretrained models as features and then train on the LSTM, which is our model architecture.

5.3. Qualitative results and comparisons

5.3.1 Results on JAFFE dataset

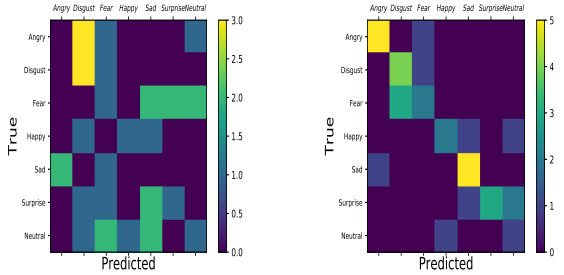
Figure 4a shows the training and validation accuracy on JAFFE dataset without the CNN features. The performance is as low as 40%. While using the CNN features, it has improved a lot with 70% validation accuracy.



(a) W/o CNN feature (b) W/ CNN feature

Figure 4: Train/validation accuracy on JAFFE

Figure 5a shows confusion matrix on the test data on JAFFE dataset. Without the CNN features, it classify few emotions like "disgust" and "happy" etc. With the pretrained model as the feature, the performance is significantly improved with each emotions, the test accuracy could achieve 80%.



(a) W/o CNN feature (b) W/ CNN feature

Figure 5: Confusion matrix on JAFFE

Table 1 shows the precision, recall and f1-score on test

data on JAFFE without pretrained CNN features. It has similar performance as the accuracy for confusion matrix. Table 2 shows the precision, recall and f1-score on test data on JAFFE with pretrained CNN features. It has better result compared with the model not using CNN feature, similar performance as the accuracy for confusion matrix.

Table 1: Measures w/o CNN on JAFFE dataset

Emotion/Measures	precision	recall	f1-score
Angry	0.00	0.00	0.00
Disgust	0.33	0.75	0.46
Fear	0.14	0.14	0.14
Happy	0.50	0.33	0.40
Sad	0.00	0.00	0.00
Surprise	0.33	0.20	0.25
Neutral	0.25	0.14	0.18
Overall	0.21	0.21	0.19

Table 2: Measures w/ CNN on JAFFE dataset

Emotion/Measures	precision	recall	f1-score
Angry	0.83	0.83	0.83
Disgust	0.57	0.80	0.67
Fear	0.50	0.40	0.44
Happy	0.67	0.50	0.57
Sad	0.71	0.83	0.77
Surprise	1.00	0.50	0.67
Neutral	0.25	0.50	0.33
Total	0.93	0.88	0.89

AS the results show in two tables, we get much better performance when we add pre-train feature extraction model. The test precision is 93% on test data.

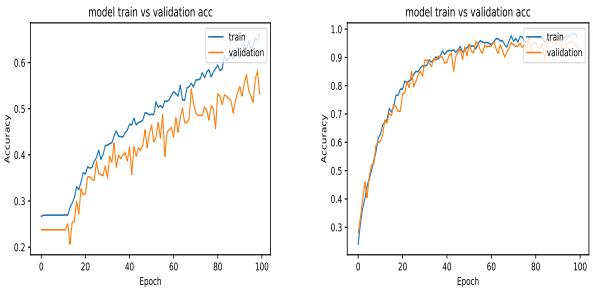
5.3.2 Results on CK+ dataset

Figure 6a shows the training and validation accuracy on CK+ dataset without CNN feature. Figure 6b shows the training and validation accuracy on CK+ dataset with CNN feature. It has greatly improved the performance with CNN feature. The training and validation accuracy could also close to 100% accuracy when using the CNN feature.

Figure 7a shows the confusion matrix on test data on CK+ dataset.

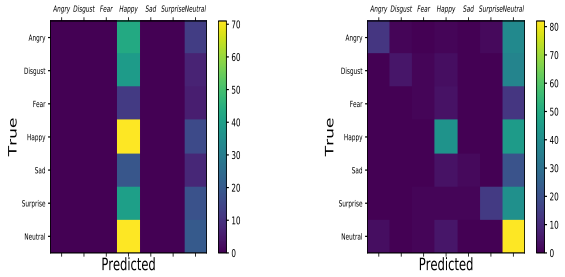
Table 3 shows the precision, recall and f1-score on test data on JAFFE

Table 4 shows the precision, recall and f1-score on test data on CK+. For CK+ dataset, since the size is much bigger than JAFFE, we get a much lower accuracy. Before we add feature extraction model, we get only 16% accuracy. When we use feature extraction model to capture face, the accuracy increase to 43% which is not bad.



(a) W/o CNN feature (b) W/ CNN feature

Figure 6: Train/validation accuracy on CK+



(a) W/O CNN feature (b) W/ CNN feature

Figure 7: Confusion matrix on CK+

Table 3: Measures without CNN on CK+ dataset

Emotion/Measures	precision	recall	f1-score
Angry	0.00	0.00	0.00
Disgust	0.26	0.13	0.18
Fear	0.00	0.00	0.00
Happy	0.00	0.00	0.00
Sad	0.00	0.00	0.00
Surprise	0.22	0.29	0.25
Neutral	0.29	0.68	0.41
Total	0.16	0.27	0.19

Table 4: Measures with CNN on CK+ dataset

Emotion/Measures	precision	recall	f1-score
Angry	0.32	0.21	0.25
Disgust	0.10	0.37	0.16
Fear	0.00	0.00	0.00
Happy	0.36	0.82	0.50
Sad	0.00	0.00	0.00
Surprise	0.30	0.12	0.17
Neutral	0.43	0.22	0.29
Overall	0.43	0.22	0.22

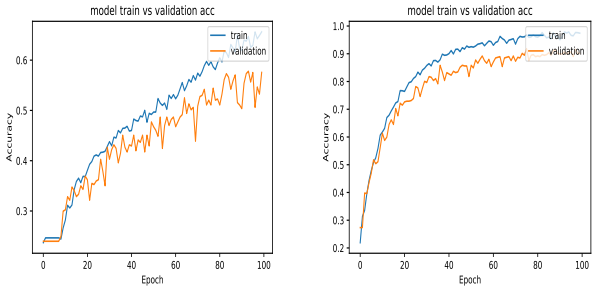
Table 5: Overall accuracy statistics

Accuracy	Validation	Test
JAFFE	0.71	0.83
CK+	0.95	0.39
total	0.91	0.44

5.4. Overall performances

. We test our overall performances on the combined two datasets, that is combining the JAFFE and CK+ dataset together as one dataset. We use the similar testing strategy for testing. The data is also divided into 70% training data, 10% validation data, 20% test data.

Figure **??** shows the training and validation accuracy on combined dataset without CNN feature. Figure 8b shows the training and validation accuracy on combined dataset with CNN feature. It has similar performance as before without combining and also has very good accuracy performance. The training and validation accuracy could also close to 100% accuracy when using the CNN feature.



(a) W/o CNN feature (b) W CNN feature

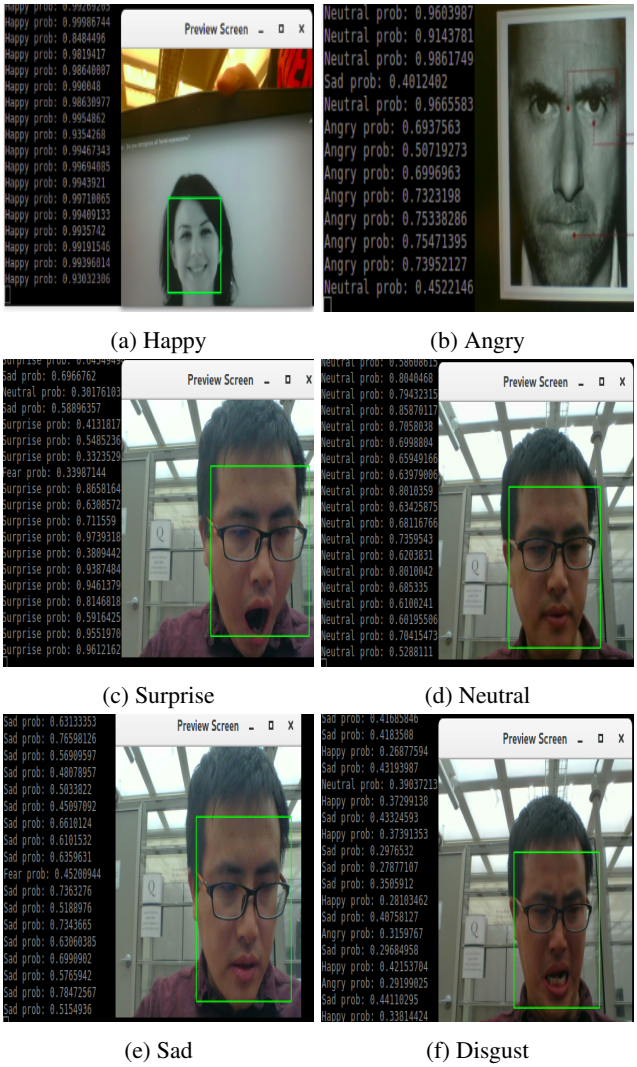
Figure 8: Train/validation accuracy on combined two dataset

Table 5 shows the overall performance on the JAFFE and CK+ dataset with validation and accuracy test. For JAFFE dataset, the model has a better test accuracy than validation accuracy. It is maybe because of the test data set is small. For CK+ large dataset with larger high-resolution, the model has a very high validation accuracy but generalize very bad, with an average accuracy only around 39%.

5.4.1 Results on video dataset

Finally, we want to test this model on videos to recognize dynamic emotions. We use camera on laptop to capture emotions in a video. This facial detect application will catch the face in video every 0.3 second, and we use our model to classify the emotion to see the accurate.

Figure 8 shows the result on the video. It shows the 6 sample and the detected emotions. For video data, the common emotions, "happy", "sad", "angry" are easy to detect. Basically if the emotions are more exaggerated, they are easy to be recognized as well. But for "disgust" emotion, it is identified as "angry", or "sad"



(a) Happy (b) Angry

(c) Surprise (d) Neutral

(e) Sad (f) Disgust

Figure 9: Test model on video dataset

6. Conclusion

In this paper, we proposed a combined model to detect and classify emotions from visual data. To consider the contextual information in the data LSTM network was used. To improve the accuracy to capture face, feature extraction model was added. Our experiments on the datasets show that our model is good at recognizing emotions on image, but not good enough on videos. From image datasets, we conclude that the more exaggerated emotion is, the easier for machine to recognize. Especially happy and angry can

be captured with high success rate. For video data, exaggerated expressions are easier to recognize as well. In general, facial emotion is pretty challenging for machine to recognize.

6.1. Possible Improvements

In our project, we use image data to train the model and use it to test video data which is not good enough. If we get a proper dataset, we can use sequential image data from videos to feed our model. For instance, if we have a number of frames for the same emotion but the results change. The result for first frame is happy, for second one is surprise and for third one is happy again.We can notice this fault and improve the model. For emotion classification, we can feed audio data to another designed audio CNN architecture. Combine this output with the visual information and feed to LSTM to get the results. The better idea to increase the accuracy in detecting video data, we can build a model to capture the key frames in the video. Key frame means the obvious frame to represent the emotion. By using this way, we can improve the performance of catching the correct emotion.

References

[1] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt. Sequential deep learning for human action recognition. In *International Workshop on Human Behavior Understanding*, pages 29–39. Springer, 2011.

[2] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*, 2014.

[3] S. Ebrahimi Kahou, V. Michalski, K. Konda, R. Memisevic, and C. Pal. Recurrent neural networks for emotion recognition in video. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 467–474. ACM, 2015.

[4] Y. Fan, X. Lu, D. Li, and Y. Liu. Video-based emotion recognition using cnn-rnn and c3d hybrid networks. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pages 445–450. ACM, 2016.

[5] A. Gudi. Recognizing semantic features in faces using deep learning. *arXiv preprint arXiv:1512.00743*, 2015.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[7] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

[8] T. Kanade, J. F. Cohn, and Y. Tian. Comprehensive database for facial expression analysis. In *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, pages 46–53. IEEE, 2000.

[9] J. Kumari, R. Rajesh, and K. Pooja. Facial expression recognition: A survey. *Procedia Computer Science*, 58:486–491, 2015.

[10] J. Schwan, E. Ghaleb, E. Hortal, and S. Asteriadis. High-performance and lightweight real-time deep face emotion recognition. In *Semantic and Social Media Adaptation and Personalization (SMAP), 2017 12th International Workshop on*, pages 76–79. IEEE, 2017.

[11] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv preprint arXiv:1312.6229*, 2013.

[12] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers. Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, 35(5):1285–1298, 2016.

[13] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[14] G. Trigeorgis, F. Ringeval, R. Brueckner, E. Marchi, M. A. Nicolaou, B. Schuller, and S. Zafeiriou. Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 5200–5204. IEEE, 2016.

[15] P. Tzirakis, G. Trigeorgis, M. A. Nicolaou, B. W. Schuller, and S. Zafeiriou. End-to-end multimodal emotion recognition using deep neural networks. *IEEE Journal of Selected Topics in Signal Processing*, 11(8):1301–1309, 2017.