Real-Time Human Emotions Recognition System Using Deep Learning and Image Processing

1st Kanisha Liyanage
dept. of Computing and Information Systems
Wayamba University of Sri Lanka
Colombo, Sri Lanka
email: kanishaliyanage97@gmail.com

2nd Nethmini Bandara
dept. of Computing and Information Systems
Wayamba University of Sri Lanka
Kandy, Sri Lanka
email: nethminibandara19@gmail.com

3rd Tharinda Vidanagama
dept. of Computing and Information Systems
Wayamba University of Sri Lanka
Kuliyapitiya, Sri Lanka
email: tharinda@wyb.ac.lk

***Abstract*—Human emotion recognition has attracted the interest of many problem solvers in the field of artificial intelligence. The emotions on a human face say so much about our thought process and give a glimpse of what's going on inside the mind. Real-time emotion recognition is to acquaint the machine with a human-like ability to recognize and analyze human emotions. This research paper proposes and implements an application to recognize human facial emotion in real-time and output an emoji according to the detected facial expression and then send a message accordingly. Deep learning in combination with Image Processing was used in order to achieve the required outcome of this research. To recognize the facial expression several models were trained using the FER-2013 data set which contains over 30000 images under 7 facial expressions. The models were able to report an accuracy of around 65% using this dataset. The models were tested using the test data set and the results were validated. Using the best model with the highest accuracy the work was extended using OpenCV and python to recognize real-time facial expressions fed in using a web camera and then give out the relevant emoji.**

Keywords—Convolutional Neural Network, Deep Learning, Facial Emotion Detection, OpenCV.

I. INTRODUCTION

Facial expressions are natural and influential signals to understand human emotional states and intentions. Also, as a part of communication human shows their emotions using facial expressions. Today, everything is getting automated through computers. It is very easy for a human to recognize emotions and facial expressions but for machines, emotion recognition is a challenging task. Facial Expression Recognition (FER) has become a very popular research subject in the field of computer vision. Recognition of facial expressions can be used in various fields, more expressively in security. We can accomplish much by accurately predicting the facial expressions of humans. The sources to do this are widely available but it needs appropriate modifications. Dealing with a such huge amount of data could be very time-consuming using traditional feature-based methods. This is the reason why researchers prefer deep learning techniques, especially Convolutional Neural Networks (CNN) for the classification of images. Many researchers have put their hands on finding the best fit to achieve an accurate output such that the results can be made useful in real-world applications. In recent years many techniques and mechanisms are used for emotion classification but developing an automated system to accomplish this task has been proven difficult.

This research paper attempts to develop an application that uses the context of the human face. To develop the application Deep Learning techniques were used and image classification techniques to build the model. In the process of building the model, various datasets were used and most of the datasets were not upright. Finally, two appropriate datasets based on the FER-2013 dataset were considered in this paper. The FER-2013 dataset is one of the most popular datasets of human facial emotions designed by Goodfellow et al. The dataset contains around 35,887 well-structured 48x48 pixel grayscale images. Figure 1 shows a set of sample images from the FER-2013 dataset. The dataset is a collection of seven different emotions labeled as follows 0: Angry, 1: Disgust, 2: Fear, 3: Happy, 4: Sad, 5: Surprise, 6: Neutral. Figure 2 shows one sample image from each class. In terms of numbers, the dataset has 28,710 examples as the training set. The test set contains 3590 examples. Figure 3 and Figure 4 show the distribution of images in 7 classes in the training dataset and test dataset respectively.

Fig. 1. Sample of the FER-2013 dataset

Fig. 2. Sample image from each class

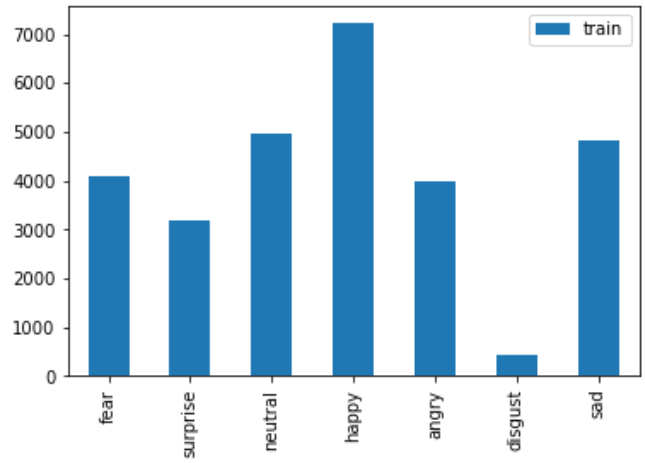

Fig. 3. Training set distribution.

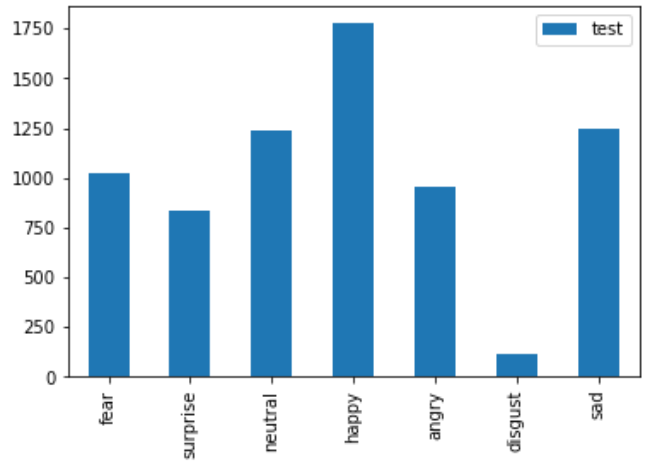

Fig. 4. Validation set distribution

The remainder of this paper is organized as follows: the next chapter contains the literature review. Section 3 and section 4 describes the methodology and the Technology respectively. The results obtained are discussed in section 5 and the paper is concluded in section 6. Finally, section 7 provides a glimpse of future works.

II. Literature

Facial expression is the mutual indicator for all humans to convey their emotions. There are many efforts to make automatic facial expression analysis tools [1] as it has application in many fields such as robotics, medicine, driving assist systems, and lie detector. Since the twentieth century, Ekman et al. [2] defined seven basic emotions, regardless of philosophy in which a human grows with the seven expressions (anger, fear, happy, sad, neutral, disgust, and surprise). Especially Convolutional Neural Networks. Convolutional Neural Networks come under the subdomain of Machine Learning which is Deep Learning. Algorithms under Deep Learning process information the same way the human brain does. In CNN, the input image data will be subjected to a set of convolution operations such as filtration and max pooling. Then, the resulting data which will be of lesser dimension compared to the original image data will be subjected to fully connected layers to predict the output. The most stated reason why CNN is preferable for image classification is as mentioned by performing the convolution operations, the dimensionality of the data shrinks significantly large. Hence, the number of parameters to be learned decreases, and the network complexity decreases which leads to fewer chances of overfitting. In recent years, many papers have been published that use deep learning for facial emotion recognition. [3] [4] [5] One such research done by Yu and Zhang [3] used a five-layer collective CNN to achieve a 0.612 accuracy. They have pre-trained their models on the FER-2013 dataset and then finetuned the model on the Static Facial Expressions in the Wild 2.0 (SFEW) dataset. Mollahosseini et al. [6] proposed a network consisting of two convolutional layers each followed by max pooling and then four inception layers. They used this network on seven different datasets including the FER-2013 dataset. They have also compared the accuracies of their proposed network with an AlexNet network trained on the same datasets. They found that their architecture had better performance on the MMI and FER-2013 datasets with comparable performances on the remaining five datasets. The FER-2013 dataset, in particular, succeeded to reach an accuracy of 0.664. Ming Li et al. propose a neural network model to overcome two shortcomings in still image-based FERs which are the inter-variability of emotions between subjects and the misclassification of emotions. The model consists of two convolutional neural networks - the first is trained with facial expression databases whereas the second is a DeepID network used for learning identity features. Most other works in the same field attempted to solve the facial emotion recognition problem by different methods like adjusting the number of layers, combining datasets, and preprocessing the dataset. In this paper, a single dataset, FER-2013 was chosen over a combination of different datasets and then experiments were conducted with different models to find the highest accuracy that each model could reach.

III. Methodology

The proposed system is building a real-time application for recognizing human facial emotions. This system is under the user context which is one type of context in mobile computing. The user context concerned in this paper is the user's emotional situation. Additionally, the proposed system contains features such as emoji suggestions according to the recognized facial expression and recommend task to the user by recognizing the user's real-time mood according to the facial expression. Also, this application plays music according to the emotion predicted.

IV. Technology

The main technology used for building the application is Deep Learning. In deep learning, Convolutional Neural Networks (CNNs) were used as the main technique because the problem is based on images and video feeds. CNNs are used for image classification. Then Python OpenCV was used for image processing for the rest of our application to get outputs.

V. Experiment and Results

A. *Model trained with Original Dataset*

The models considered were trained in different CNN architectures. In the VGG19-V2 architecture model, in 5 epochs only a 41.70% validation accuracy was obtained. In the MobileNet-V2 architecture model, 38.78% validation accuracy was gained. Then the model was trained with a normal CNN model with six hidden layers. With 30 epochs, this model gained a validation accuracy of 60.27%. In that

same model with 150 epochs, 59.77% validation accuracy was obtained. Finally, a model with 5 hidden layers was built and ran with 30 epochs with a 0.001 learning rate. This model achieved a training accuracy of 79.77% and a validation accuracy of 64.54% using approximately 3.2 million parameters. Figure 5 shows the model summary.

Figure 6 shows the model accuracy and Figure 7 is showing the model loss plot. According to both plots, high accuracy is achieved on the training set but accuracy on the validation set is stuck at 66% also no overfitting can be seen in the dataset hence it can be concluded that the inefficiency may be due to the unbalanced dataset.

Finally, the confusion matrices of the training set and the test set were plotted. Figure 8 and Figure 9 show matrices respectively.

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 48, 48, 32)	320
conv2d_1 (Conv2D)	(None, 48, 48, 64)	18496
batch_normalization (BatchNo	(None, 48, 48, 64)	256
max_pooling2d (MaxPooling2D)	(None, 24, 24, 64)	0
dropout (Dropout)	(None, 24, 24, 64)	0
conv2d_2 (Conv2D)	(None, 24, 24, 128)	73856
conv2d_3 (Conv2D)	(None, 22, 22, 256)	295168
batch_normalization_1 (Batch	(None, 22, 22, 256)	1024
max_pooling2d_1 (MaxPooling2	(None, 11, 11, 256)	0
dropout_1 (Dropout)	(None, 11, 11, 256)	0
flatten (Flatten)	(None, 30976)	0
dense (Dense)	(None, 1024)	31720448
dropout_2 (Dropout)	(None, 1024)	0
dense_1 (Dense)	(None, 7)	7175

Total params: 32,116,743
Trainable params: 32,116,103
Non-trainable params: 640

Fig. 5. Final model summary

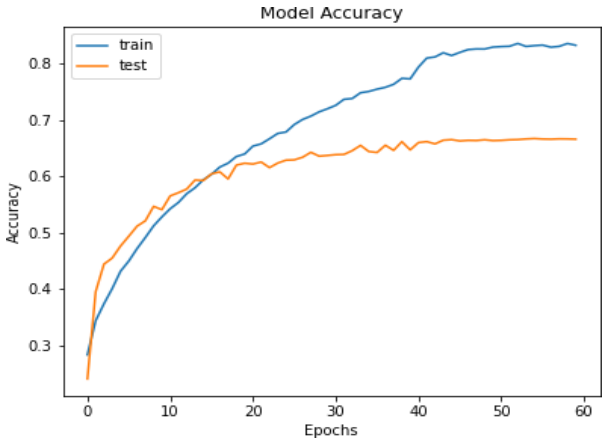

Fig. 6. Accuracy plot

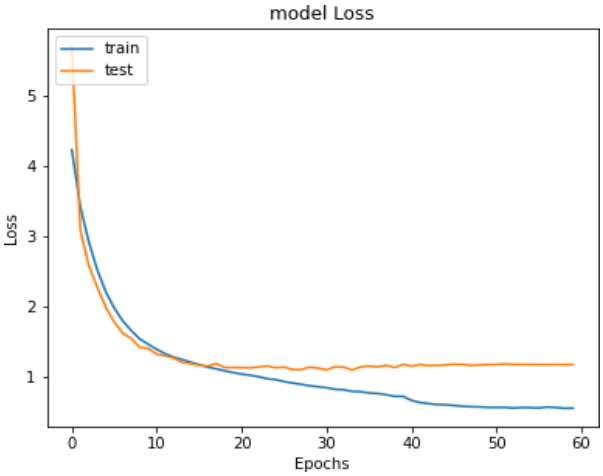

Fig. 7. Loss plot

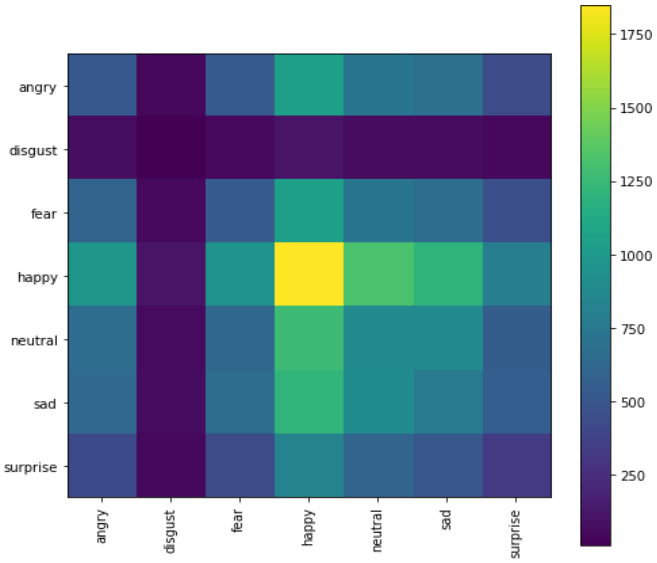

Fig. 8. Confusion matrix of the training set

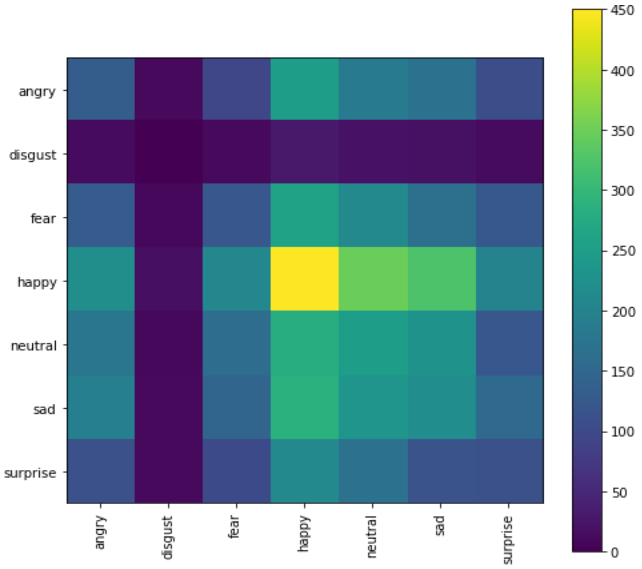

Fig. 9. Confusion matrix of the test set

All the above models were trained on the original dataset. Next, in order to achieve better accuracy more models were trained by preprocessing the dataset.

B. Model trained with 'disgust' images removed dataset

First, the 'disgust' emotion data folder was removed and a model was trained with the remaining. The model consists of 5 hidden layers, ran with 60 epochs. This model achieved a training accuracy of 69.80% and a validation accuracy of 65.24%. Figure 10 and Figure 11 show the loss plot and the accuracy plot. Figure 12 and Figure 13 show the confusion matrices for the training set and the test set.

C. Model trained with 'happy' images reduced dataset

The next model was trained by reducing the number of images in the happy folder approximately to the number of images in other folders. This was applied to the new dataset created by removing the disgust folder. To achieve this the happy images in the training set were reduced by 2000 images and the test set by 500 images. This model was also trained with 5 hidden layers with 60 epochs. The model gained a training accuracy of 67.53% and a validation accuracy of 63.04%. Figure 14 and Figure 15 show the loss plot and the accuracy plot for the model trained by removing the disgusting image folder. This model was not much better than the disgust-removed model but better that the model trained with the original dataset.

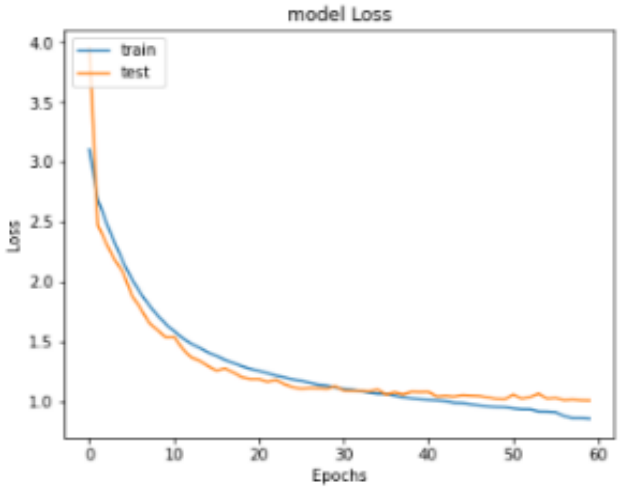

Fig. 10. Loss plot

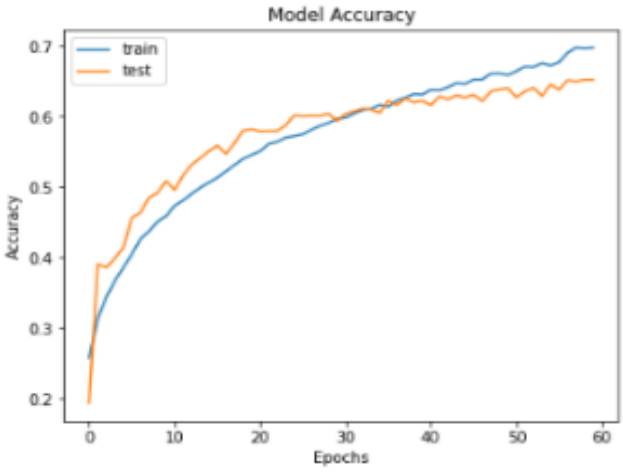

Fig. 11. Accuracy plot

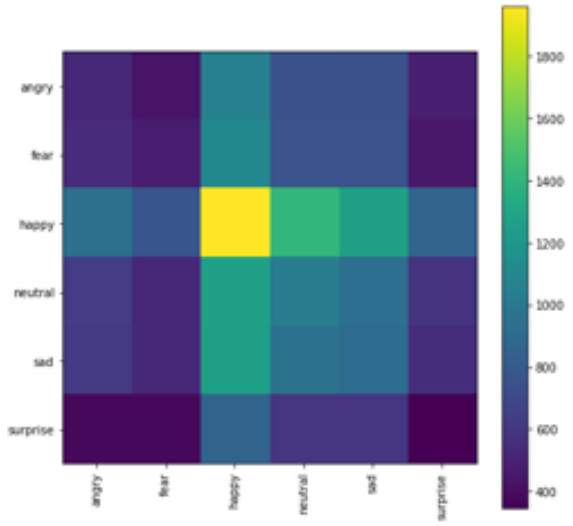

Fig. 12. Confusion matrix of the training set

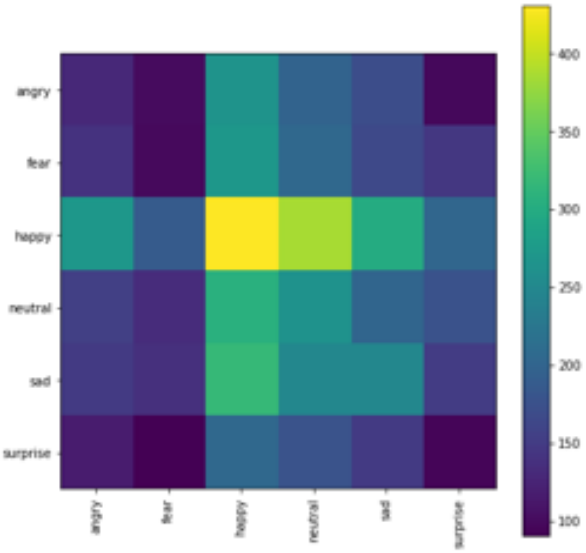

Fig. 13. Confusion matrix of the test set

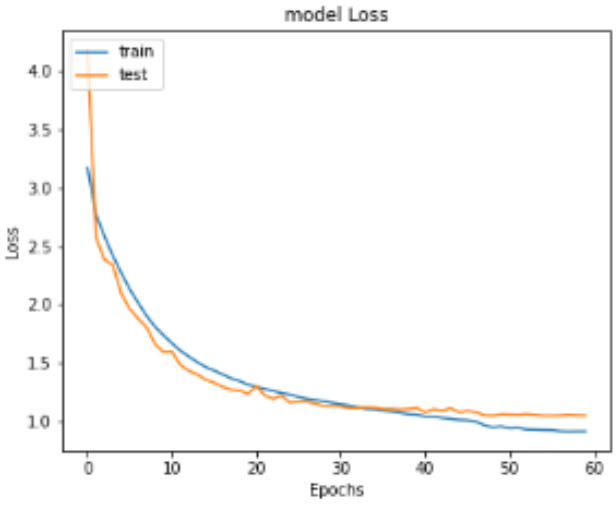

Fig. 14. Loss plot

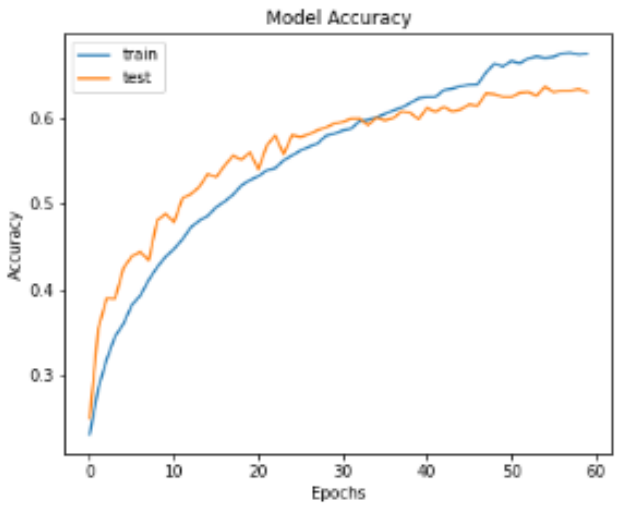

Fig. 15. Accuracy plot

D. *Model trained after preprocessing and balancing dataset*

With the results obtained by adjustments to the dataset, the final decision to train the model was first clean the dataset by removing the outliers. The second step was to balance the data set. Accordingly, one emotion folder in the training set was set to have 4500 images and the test set have 1500 images. Some folders like happy already had images of more than 4500 and they were brought down to the considered number by manually deleting the images. The folders that had images less than the set value were brought up to the required number by image augmentation.

The final model developed was with 5 hidden layers and ran with 60 epochs. This model achieved a training accuracy of 85.65% and a validation accuracy of 66.22% using approximately 4.8 million parameters. Though the accuracy is high; this model is highly overfitted.

E. *Emotion prediction application*

The model trained with the 'disgust' removed dataset was used for the emotion-predicting application developed using Python OpenCV & Python Custom Tkinter GUI toolkit (Figure 16). The current mood of the user, emoji relevant to the mood, and a message relevant to the mood are displayed on the screen when the user gives input using a camera live feed. Another option provided in this application was playing music relevant to the mood. This application was tested in real-time feeding inputs using people with different moods (Figure 17).

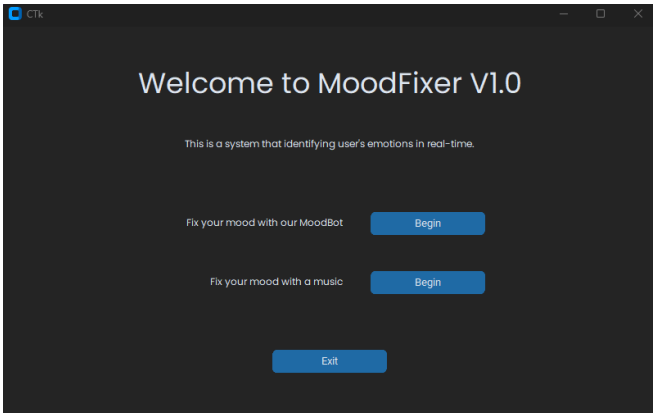

Fig. 16. Application interface

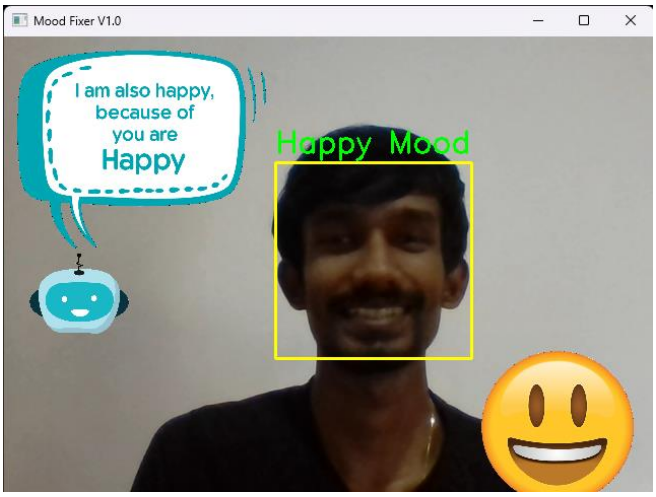

Fig. 17. The application predicts the mood correctly, show the relevant emoji and the message

VI. Conclusion

A general building design for creating real-time CNN was proposed and tested. In this application, facial emotions are recognized using a trained model. In the process of training the model, the aim was to achieve better accuracy for the model. To make the results and the recognition process of the facial expression become more precise different methods were tried and tested. When checked with the real-time camera live feed the model could recognize neutral, happy, and surprise easily and it hardly recognizes others. Though the application gives these results based on the mood taken from the camera the user can be in a different state in real, like he will laugh but actually he is sad, the application takes the input laughing. In conclusion, the model was developed and trained to recognize facial expressions and then according to the recognized facial emotion give out the relevant emoji and send a message to the user accordingly.

VII. Future Works

Machine learning models are nowadays used in numerous applications due to convenience. This model can be used to build an emoji suggestion chat application identifying the user's facial context. So that the user does not have to go through all the emotions to find the suitable one. In the near future users will be able to send their current mood with just one click using facial emotions recognition emoji suggestions. Also, this application can be used for VR and AR technologies.

References

[1] Zafar B, Ashraf R, Ali N, Iqbal M, Sajid M, Dar S, Ratyal N (2018) A novel discriminating and relative global spatial image representation with applications in CBIR. Appl Sci 8(11):2242.

[2] Ekman P, Friesen WV (1971) Constants across cultures in the face and emotion. J Personal Soc Psychol 17(2):124.

[3] Zhiding Yu & Cha Zhang. (2015). Image-based Static Facial Expression Recognition with Multiple Deep Network Learning. 435-442.

[4] Raghuvanshi, A., & Choksi, V. (2016). Facial Expression Recognition with Convolutional Neural Networks.

[5] Ebrahimi Kahou, S., Michalski, V., Konda, K., Memisevic, R., and Pal, C. (2015). Recurrent neural networks for emotion recognition in video. 467-474.

[6] A. Mollahosseini, D. Chan, and M. H. Mahoor. (2016). Going deeper into facial expression recognition using deep neural networks. 2016

IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Placid, NY, 1-10.

[7] Ashley. (2020). *An Overview on Convolutional Neural Networks*. Medium, https://medium.com/swlh/an-overview-on-convolutional-neural-networks-ea48e76fb186.

[8] Rohit Thakur. (2019). *Step-by-step VGG16 implementation in Keras for beginners*. Towards data science, https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c.

[9] Great Learning. (2021). *Everything you need to know about VGG16*. Medium, https://medium.com/@mygreatlearning/everything-you-need-to-know-about-vgg16-7315defb5918.