# REAL TIME HUMAN FACE EXPRESSION RECOGNITION

Priyanka Rasal,
Kareena More,
Tejas Pagar,
Adil Tamboli

*Research Scholar, Department of Computer Engineering, Sapkal College of Engineering Nashik, Maharashtra, India*
*Research Scholar, Department of Computer Engineering, Sapkal College of Engineering Nashik, Maharashtra, India*
*Research Scholar, Department of Computer Engineering, Sapkal College of Engineering Nashik, Maharashtra, India*
*Research Scholar, Department of Computer Engineering, Sapkal College of Engineering Nashik, Maharashtra, India*

**ABSTRACT**

*This study proposes a system for facial expression recognition using Convolutional Neural Networks (CNNs) trained on the FER2013 dataset. The objective is to accurately classify human emotions from facial expressions, including anger, disgust, fear, happiness, sadness, surprise, and neutral. The CNN model is trained on FER2013, a dataset that contains over 35,000 labeled images of human faces displaying one of the seven emotions. The CNN model extracts features from the input image using convolutional and pooling layers, followed by fully connected layers for emotion prediction. The model is trained by adjusting the weights to minimize the difference between the predicted and true emotions. The system is evaluated using a separate set of data called the validation set. The validation accuracy of the proposed system was found to be 60%.*

*Facial expression recognition using CNNs has numerous applications, such as in human computer interaction, virtual reality, and security systems. However, challenges remain in handling variations in lighting, pose, and occlusion, which can affect the accuracy of the system. The proposed system contributes to the field of computer vision by offering a viable approach for emotion recognition and has the potential to be further developed for real-world applications.*

**Keyword: -** *Facial expression recognition, Convolutional Neural Networks, FER2013 dataset, Emotion recognition, Image classification, Human-computer interaction, Computer vision.*

## 1. INTRODUCTION

Real-time face expression recognition using CNN (Convolutional Neural Network) is a popular research area in computer vision. It involves detecting and recognizing human facial expressions in real-time, which has a wide range of applications such as in human-computer interaction, video surveillance, and healthcare.

CNN is a deep learning architecture that has become popular due to its ability to achieve high accuracy in image classification tasks. CNNs can learn hierarchical representations of images, which makes them ideal for recognizing complex patterns in images, such as human faces.

To perform real-time face expression recognition using CNN, a CNN model is trained on a large dataset of labeled facial expressions. The CNN model is then used to detect and recognize facial expressions in real-time video streams. The recognition process involves detecting the facial landmarks and extracting the relevant features from the facial images, which are then fed into the CNN model for classification.

There are different approaches proposed for real-time face expression recognition using CNN, such as the use of facial landmarks, facial action units, and facial geometry. The performance of these approaches has been evaluated on various datasets, such as the CK+ dataset, the JAFFE dataset, and the EmoReact dataset.

However, real-time face expression recognition using CNN faces several challenges, such as occlusions, variations in lighting conditions, and facial expressions that occur in rapid succession. Researchers have proposed various techniques to overcome these challenges, including the use of data augmentation, feature normalization, and real-time data processing.

In conclusion, real-time face expression recognition using CNN is an important area of research in computer vision that has shown promising results. It has the potential to improve the accuracy and speed of facial expression recognition in various applications.

## 2. PREVIOUS WORK

Chung-Lin and Yu-Ming proposed Point Distribution Model (PDM) approach to analyze facial expression based on the facial feature extraction. PDM approach analysis the statistics of the coordinates of the classified or labeled points over the training set. The proposed approach is performed by using 180 images from 15 volunteers, each volunteer demonstrates six expressions, and then 12 images are chosen from each volunteer. Action Parameters (AP) Classifier is performed in order to classify and match the extracted features from facial images. The proposed approach achieved overall accuracy of 84.41 [1].

Support Vector Machine (SVM) algorithm is used by Philipp and Rana to classify Cohn- Kanade (CK) facial expression and live video in order to identify the emotions universally recognized which are (e.g. for the basic emotions of ‘anger’, ‘disgust’, ‘fear’, ‘joy’, ‘sorrow’ or ‘surprise’) supplied during training. In this study, SVM achieved 87.90 of recognition performance [2].

Different feature extraction techniques were presented separately in the study performed by Tommaso, Caifeng, Vincent and Ralph to extract features from facial images; These techniques are Histogram of Oriented Gradients (HOG), Local Binary Patterns (LBP), and Local Ternary Patterns (LTP). The used techniques were applied with various parameters of facial expression recognition. The extracted features were classified by Support Vector Machine (SVM). The Cohn-Kanade (CK) database was used, which was created from 100 persons, their ages from 18 to 30 years. 310 images were selected 7 from CK database for these experiments. LBP achieved the best accuracy of recognition that reached 92.9 , HOG achieved 92.7 , and finally; LTP achieved 91.7 [3].

Kharat and Dudul investigated three various techniques for feature extraction from facial expressions for emotion recognition on six universally recognized basic emotions, namely angry, disgust, fear, happy, sad and surprise along with neutral one. These techniques are Discrete Cosine Transform (DCT), Fast Fourier Transform (FFT) and Singular Value Decomposition (SVD). Support Vector Machine (SVM) classifier is used to classify the extracted facial features. The study is performed using JAFFE database. This database contains 219 images. DCT+SVM method achieved recognition rate of 94.29, and FFT+SVM method achieved 94.29, and SVD+SVM method achieved 92.86 [4].

Murugappan, Nagarajan and Yaacob extracted features from facial images by using Discrete Wavelet Transform (DWT) approach. They collected 460 image from a series of videos and used them in their experiments. After extracting the features from images, they used two various classifiers to classify these features K-Nearest Neighbors (KNN) algorithm and Linear Discriminant Analysis (LDA) algorithm. LDA provides fast evaluations for input samples by
calculating the distance between a new sample and training samples in each class weighed by their variability matrices. LDA tries to find an optimal hyper plane to six classes of emotions (fear, neutral, happy, disgust, surprise and sad). Recognition rate of 83.26 were achieved with KNN classifier and 75.21 with LDA classifier.

## 3. DATASET PREPARATION

The dataset from a Kaggle Facial Expression Recognition Challenge (FER2013) is used for the training and testing.

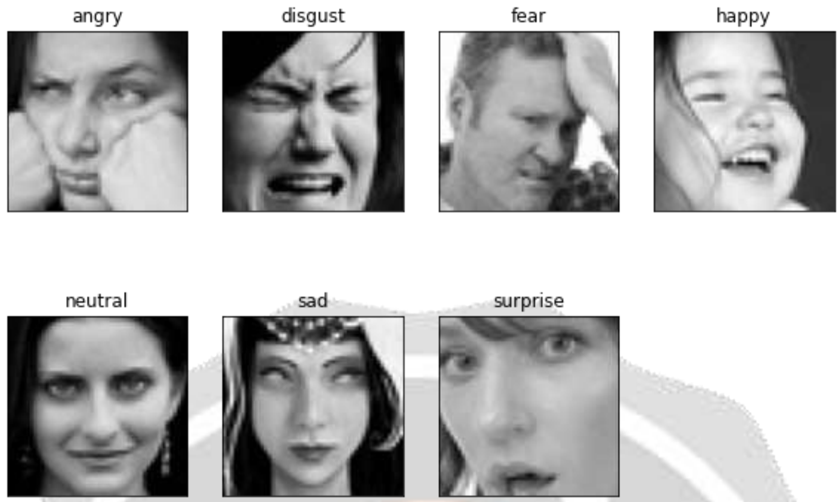



**Fig 1: Dataset Sample Image**

It comprises pre-cropped, 48-by-48-pixel grayscale images of faces each labeled with one of the 7 emotion classes: anger, disgust, fear, happiness, sadness, surprise, and neutral. Dataset has training set of 35887 facial images with facial expression labels. The dataset has class imbalance issue, since some classes have large number of examples while some has few. The dataset is balanced using oversampling, by increasing numbers in minority classes.

The balanced dataset contains 40263 images, from which 29263 images are used for training, 6000 images are used for testing, and 5000 images are used for validation.

## 4. METHODOLOGY

### 4.1 System Workflow

The facial expression recognition system is implemented using convolutional neural network. System Architecture diagram is as follows:

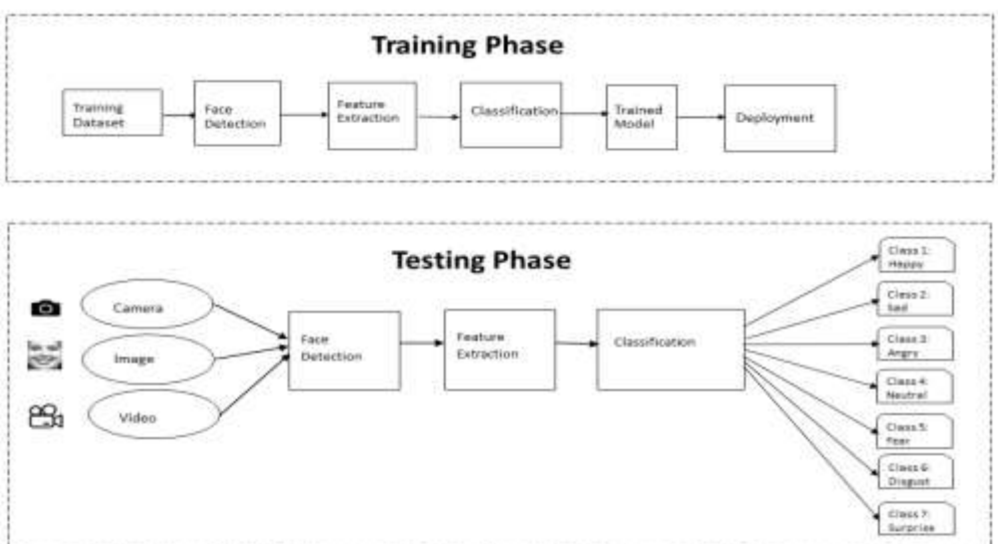



**Fig 2: System Architecture Diagram**

During training, the system receives a training data comprising grayscale images of faces with their respective expression label and learns a set of weights for the network. The training step took as input an image with a face. Thereafter, an intensity normalization is applied to the image. Normalized images are used to train the Convolutional Network.
To ensure that the training performance is not affected by the order of presentation of the examples, a validation dataset is used to choose the final best set of weights out of a set of trainings performed with samples presented in different orders. The output of the training step is a set of weights that achieve the best result with the training data. During test, the system received a grayscale image of a face from test dataset and output the predicted expression by using the final network weights learned during training. Its output is a single number that represents one of the seven basic expressions.

## 5. RESULTS

### 5.1 Model 1

First CNN model is compiled using optimizer 'RMSprop', loss function 'categorical-cross entropy' and matrices 'accuracy'. Concept of 'Early Stopping' is used with the patience of 2. Early stopping is a technique used in machine learning to prevent overfitting of a model. Overfitting occurs when a model learns the training data too well and fails to generalize to new, unseen data. Model is trained with 10 epochs. During first epoch, training accuracy was 26% and validation accuracy was also 26%. During each epoch it was increasing and finally on the sixth cycle it gives training accuracy was 65% and validation accuracy was also 44%.
Graphical representation of loss and accuracy of both training and testing data is shown in Fig 3.

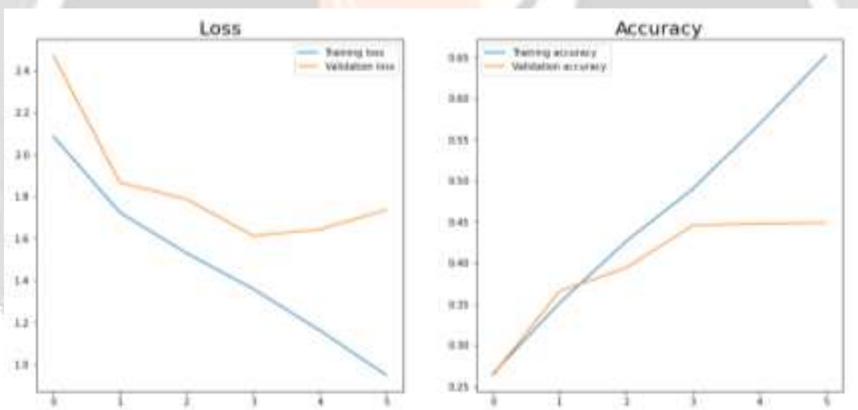



**Fig 3: CNN Model 1 accuracy vs loss**


| | precision | recall | f1-score | support |
|---|---|---|---|---|
| angry | 0.36 | 0.41 | 0.38 | 960 |
| disgust | 0.41 | 0.39 | 0.40 | 111 |
| fear | 0.33 | 0.36 | 0.34 | 1018 |
| happy | 0.67 | 0.68 | 0.68 | 1825 |
| neutral | 0.50 | 0.31 | 0.38 | 1216 |
| sad | 0.31 | 0.47 | 0.37 | 1139 |
| surprise | 0.72 | 0.40 | 0.51 | 797 |
| accuracy | | | 0.46 | 7066 |
| macro avg | 0.47 | 0.43 | 0.44 | 7066 |
| weighted avg | 0.49 | 0.46 | 0.47 | 7066 |

**Fig 4: model 1 confusion matrix**

**Fig 5: CNN model 1 Classification report**

As shown in Fig 4, CNN model 1 has not performed very well for some classes. But for some of them, it performed pretty good. For class 3, it has predicted 1244 images correctly which was impressive but at the same time for class 0 and class 5, performance was very poor.

Finally, in detail classification report is shown in Fig 5 in which precision, recall, f1-score, accuracy etc. for all the classes is represented clearly.

At the end of results of CNN model 1, it could be concluded that our model is not trained well enough and improvements could be made. That’s why we are moving to model 1 with different parameters.

**5.2 Model 2**

The second CNN model is compiled using optimizer ’Adam’ with the learning rate 0.001 which has a massive impact than RMSprop which we use in last model, loss function ‘Categorical Cross-entropy’ and matrices ‘accuracy’ which remains same. At the end of model training i.e. on epoch 22, training accuracy was 70% and validation accuracy was 60% which was the massive improvement as compared to model 1.

Graphical representation of loss and accuracy of both training and testing data using model 2 is shown in Figure 6. There was a steep increase in accuracy while steep decrease in loss. This model is performing very good as compared to model 1.

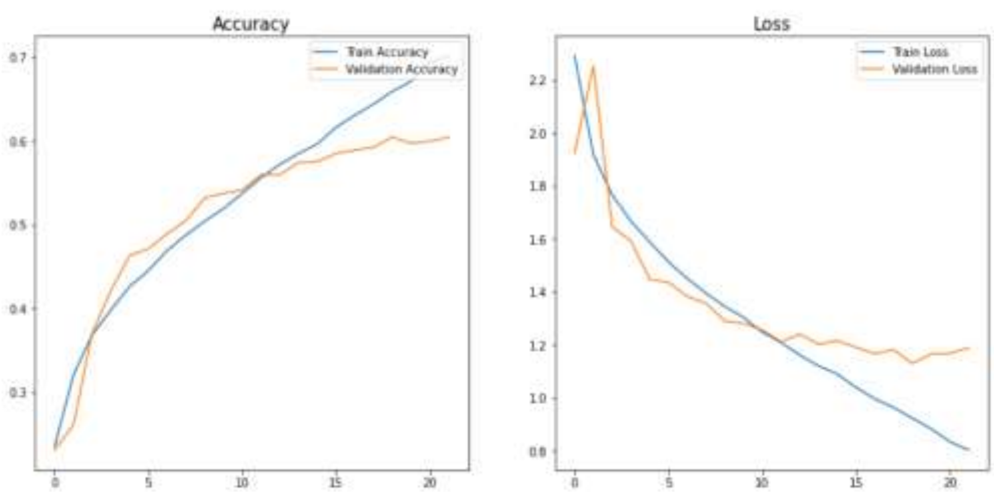
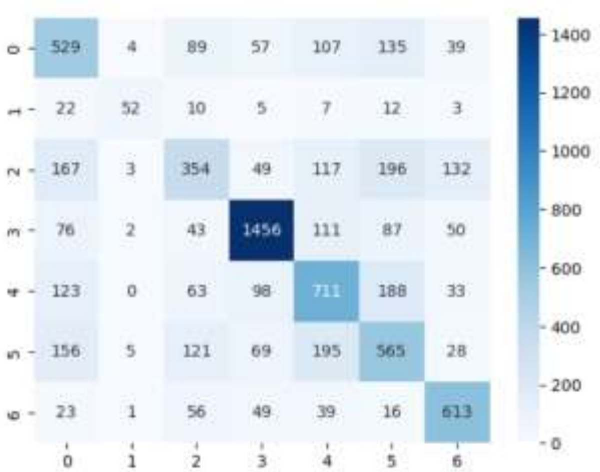



**Fig 6: CNN Model 2 accuracy v loss**




|  | precision | recall | f1-score | support |
|---|---|---|---|---|
| angry | 0.48 | 0.55 | 0.51 | 960 |
| disgust | 0.78 | 0.47 | 0.58 | 111 |
| fear | 0.48 | 0.35 | 0.40 | 1018 |
| happy | 0.82 | 0.80 | 0.81 | 1825 |
| neutral | 0.55 | 0.58 | 0.57 | 1216 |
| sad | 0.47 | 0.50 | 0.48 | 1139 |
| surprise | 0.68 | 0.77 | 0.72 | 797 |
| accuracy |  |  | 0.61 | 7066 |
| macro avg | 0.61 | 0.57 | 0.58 | 7066 |
| weighted avg | 0.61 | 0.61 | 0.60 | 7066 |

**Fig 7: CNN model 2 confusion matrix** **Fig 8: CNN model 2 Classification Report**

As shown in Fig 7, CNN model 2 has performed very well for some classes. But for some of them, it performed a bit less. For class 3, it has predicted 1456 images correctly but at the same time for class 0 and class 5, there were many wrong predictions.

Finally, in detail classification report is shown in Figure ?? in which precision, recall, f1-score, accuracy etc. for all the classes is represented clearly.

After watching model 2's performance, result was decent and far better than model 1. That's why we decided to go with model 2 for the further operations.

**5.3 Final results on new data**

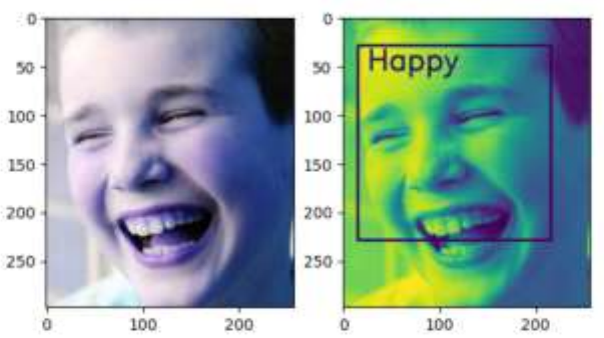



Fig 9: Result 1


Fig 10: Result 2

## 6. CONCLUSIONS

In conclusion, the project aimed to develop a facial expression recognition system using Convolutional Neural Networks (CNN) and evaluate its performance on the FER2013 dataset. The model achieved an accuracy of 60%, which is a moderate level of performance. Despite the limitations in model accuracy, the project successfully demonstrated the feasibility of using CNNs for facial expression recognition. The model's accuracy can be improved in future work by exploring different network architectures, hyperparameters, and data augmentation techniques. The FER2013 dataset used in the project has limitations in terms of sample size and diversity of facial expressions. Therefore, future work can involve using larger and more diverse datasets to train and evaluate the model's performance. Overall, the project contributes to the growing body of research on facial expression recognition and can have practical applications in areas such as human-computer interaction, emotion analysis, and mental health assessment, research work.

## 7. REFERENCES

[1] C.-L. Huang and Y.-M. Huang, "Facial expression recognition using model-based feature extraction and action parameters classification," Journal of visual communication and image representation, vol. 8, no. 3, pp. 278–290, 1997.

[2] G. M. S. NAJAH and G. S¸eng¨ul, "Emotion estimation from facial images," Ph.D. dis-sertation, PhD dissertation, Atilim University, 2017.

[3] K.-C. Liu, C.-C. Hsu, W.-Y. Wang, and H.-H. Chiang, "Real-time facial expression recognition based on cnn," in 2019 International Conference on System Science and Engineering (ICSSE). IEEE, 2019, pp. 120–123.

[4] G. Kharat and S. Dudul, “Human emotion recognition system using optimally designed svm with different facial feature extraction techniques,” WSEAS Transactions on Computers, vol. 7, no. 6, pp. 650–659, 2008.

[5] M. Murugappan, N. Ramachandran, Y. Sazali et al., “Classification of human emotion from eeg using discrete wavelet transform,” Journal of biomedical science and engineer-ing, vol. 3, no. 04, p. 390, 2010