iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 8 **Issue:** VI **Month of publication:** June 2020

DOI: http://doi.org/10.22214/ijraset.2020.6165

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue VI June 2020- Available at www.ijraset.com

Live Emotion Recognition System using CNN

Suyog Gadage [1], Vishal Parandwal [2], Suyash Gorde [3], Rohit Surve [4], Amol Kamble[5]
[1, 2, 3, 4]*B. E. Students,* [5]*Assistant Professor, Deptt. of Computer Engineering, Modern Education Society's College of Engineering Pune, India.*

Abstract: ***Nowadays, Facial Expression Recognition (FER) or Emotion Recognition (ER) is significant for Security Systems, Human-Computer Interaction, Lie Detection, Monitoring at ATMs, etc. It's a system that recognizes the seven basic human emotions (Happy, Angry, Sad, Fear, Disgust, Surprise and Neutral). Real-Time Emotion Recognition implementation may become complex in Image Classification. Use of Deep Learning is important in Image Classification. In Deep Learning, Convolutional Neural Networks (CNNs) can help to reduce difficulties of Emotion Recognition. But Traditional CNN-based methods, i.e. Shallow CNN (SHCNN), leads to less accuracy of the CNN model. In this paper, we propose a Deep Convolutional Neural Network (DCNN) architecture to extract features of ER tasks. The pre-processing technique is used to extract only relevant features from the dataset. The dataset is used FER2013 which comprises 48-by-48-pixel grayscale images of human faces, each labelled with one of 7 emotion categories, an image set if 35,887 examples. For training, and testing dataset, the split ratio is- 80:20. Proposed DCNN comprises 4 Convolution layers, 2 Max-Pooling layers and 2 Fully Connected layers, 'Adam' as an optimizer and 'binary cross-entropy' as a loss function. The result of final recognition is calculated using sigmoid classification. Accuracy of SHCNN is 65-68% however, our proposed method accuracy is 90.95% for FER2013 dataset. For convenience for the user, we develop Desktop application and Web application to use our Emotion Recognition system.***
Keywords: Facial Expression Recognition, Deep Learning, Convolutional Neural Networks (CNNs), Image Classification, Sigmoid Classification, Optimizer, Loss Function.

I. INTRODUCTION

Facial Expression Recognition (Emotion Recognition) system aims to recognize or predict the seven basic emotions (e.g., happiness, sadness, anger, surprise, disgust, and fear) from human facial images, as shown in Fig. 1. ER system evokes appreciable attention because of its potential application in human-abnormal behaviour detection, computer interfaces, autonomous driving, health management, and other similar tasks.

Convolutional Neural Networks (CNNs) is one of the most popular architectures of Deep Learning that is used to solve the ER problem. CNN consists of four most important layers:

1) *Convolutional Layer:* It is the first layer of the CNN that extracts the feature from an input image. It is the mathematical operation that takes two input images such as image matrix and filter or kernel.
2) *ReLU (Rectified Linear Unit) Layer:* It is a transform function only activates a node or pixel input above a certain value otherwise value of pixel will be zero.
3) *Pooling Layer:* In this layer, the shrinkage of the image matrix into a smaller size is done. The window/image matrix size varies.
4) *Fully Connected Layer:* This is the final layer in CNN that stack up all the layers of CNN. Flattening of image matrix into a vector and feed it into a fully connected layer like a neural network.

There are many ways to improve ER accuracy. For example, before the final step, the average of current recognition result (i.e. passing from various CNN layers) and previous recognition result is calculated and it is considered as final recognition result [1]. Considering static and micro-expression simultaneously [3]. The Multimodal system that uses 2D+3D Facial expressions [4]. Combining features of CNN and LSTM [5]. All of these methods focus on architecture to improve the accuracy of CNN.

In this paper, we proposed the method to focus on the pre-processing of the FER2013 dataset and improving the architecture of traditional CNN. The pre-processing procedure is extracted relevant features of the FER2013 dataset i.e. not to extract the missing or any irrelevant image pixel value. Our proposed method is Deep Convolutional Neural Network (DCNN), which is denser than Shallow CNN (SHCNN). The number of convolutional, max-pooling and fully connected layers are more than SHCNN.

The method is divided into three steps:

a) Pre-processing of the dataset FER2013.
b) Face Detection.
c) The Deep CNN (DCNN) architecture.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 8 Issue VI June 2020- Available at www.ijraset.com

In this way, Emotion Recognition system can be implemented through the DCNN and integrating webcam with ER system.

Fig. 1

II. EMOTION RECOGNITION SYSTEM

A. Pre-Processing

Pre-processing of the FER2013 dataset is just extracting the valuable or relevant images (pixel values). This dataset contains some missing values, irrelevant values, etc. this can reduce accuracy for any model. Thus, pre-processing is the first step for increased accuracy for the proposed method.

B. Face Detection

For face detection, we used Logitech C270 HD webcam with Open CV face program and the parameters from the face training set haarcascade_frontalface_default.xml. The detection was programmed in Python to achieve a detection rate of about 10 images per second. This step was followed by grayscale conversion and resizing for each image to immediately and quickly input the captured images into the trained framework.

C. Proposed Architecture

1) *Deep Convolutional Neural Network (DCNN):* We used the dataset FER2013 that comprises seven basic emotions (e.g. Happiness, Sadness, Anger, Surprise, Disgust, and Fear). There are 28709 48×48 grayscale images in the training set, 7178 48×48 grayscale images in the validation set, and 7178 48×48 grayscale images in the test set. The training set was inputted into the DCNN architecture, with parameters of Python, Keras and TensorFlow for training the architecture. As shown in Fig. 2, the architecture is divided into an input layer, four layers of convolution, two layers of pooling, and two fully connected layers for obtaining the predicted results of 7 emotions. The first layer of convolution uses 64 5×5 convolution kernels, while the second convolution layer, the third convolution layer and fourth convolution layer use the same size of the kernel, i.e. 128, 512, and 512 3×3 convolution kernel. The ReLU layer that applying the ReLU activation function such as max (0, x) is used at each layer after each convolution. Max-pooling layer of 2×2 is embedded into each layer after ReLU layer. As the Fully Connected layer is Flattening of image matrix that is the output of previous three layers, the three layers (i.e. Are in matrix form) are converted into a vector and feed into first fully connected layer which containing 256 neurons then output if first fully connected layer is fed into a second fully connected layer which contains 512 neurons. The DCNN layers are utilized for processing images, the last layer comprises the sigmoid function. The sigmoid function is usually used in the output layer of binary classification, where the result is 0 or 1, as the value for the sigmoid function lies between 0 and 1 only so, the result can be predicted easily to be 1 if the value is greater than 0.5 and 0 otherwise.
2) *Optimizers:* Adam can be looked at as a combination of RMSprop and Stochastic Gradient Descent with momentum. It uses the squared gradients to scale the learning rate like RMSprop and it takes advantage of momentum by using moving average of the gradient instead of gradient itself like SGD with momentum. Adam is an adaptive learning rate method, which means, it computes individual learning rates for different parameters. Its name is derived from adaptive moment estimation, and the reason it's called that is because Adam uses estimations of first and second moments of gradient to adapt the learning rate for each weight of the neural network.

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$$
$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$$
$$w_t = w_{t-1} - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

..... (1)

3) *Loss Function:* The goal of machine learning and deep learning is to reduce the difference between the predicted output and the actual output. This is also called as a Cost function (C) or Loss function. We have used binary cross-entropy loss function (BCE). BCE loss is used for the binary classification tasks. BCE Loss creates a criterion that measures the Binary Cross Entropy between the target and the output. If we use the BCE Loss function, we need to have a sigmoid layer in our network.

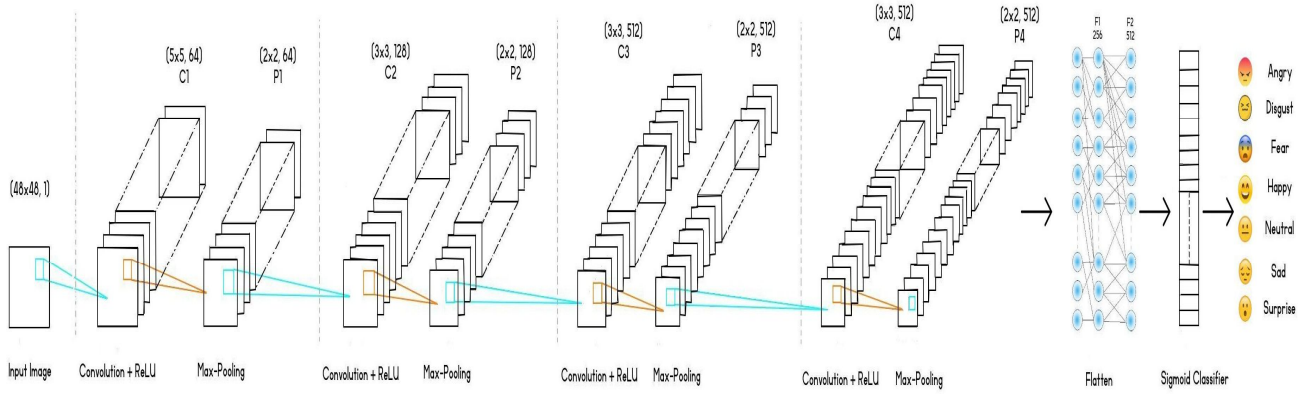


Fig. 2

4) *Design Method:* Design method contains the design for training the model with the dataset FER2013 get from Kaggle. Training, and Testing images are split into 80:20 ratio i.e. 80% of dataset images are for training, and rest 20% for testing. The training model is given in Fig. 3 which contains blocks of defining dataset pre-processing, DCNN architecture to train the model, optimizers and loss functions, and training the model over this model with the dataset and finally saving the model.

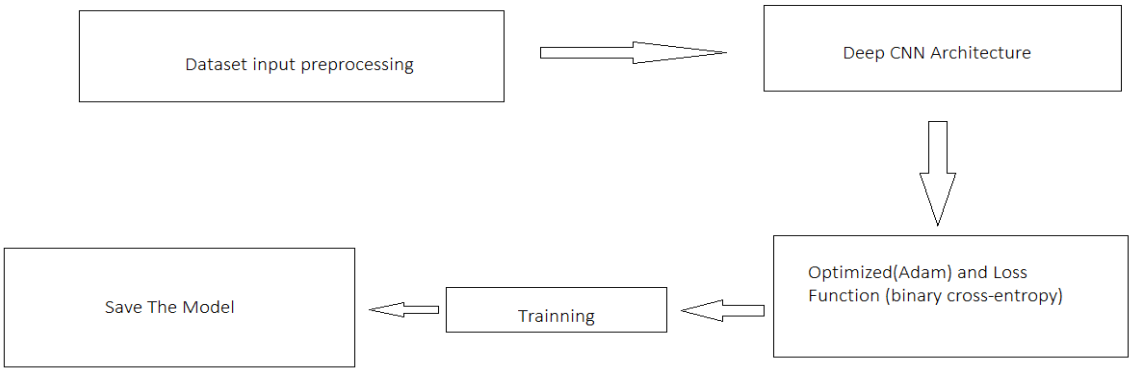


Fig. 3

After the model is trained over the train data, it has to be tested. This is done according to Fig. 4 for testing the model. In this process, the saves model is loaded and the testing images (from dataset or images given by the user) pass through the model and finally predicts the output of the model i.e. It is happy, sad, anger, surprise, disgust, or fear.

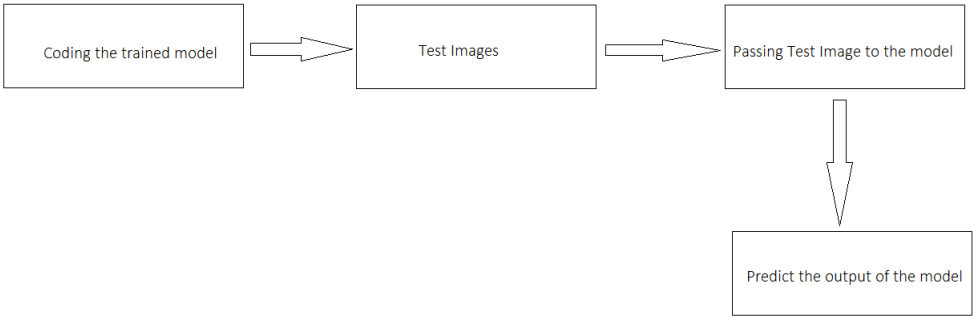


Fig. 4

III.EVALUATIONS AND RESULTS

A. Dataset

1) *FER2013:* The FER2013 dataset consists of 35887 grayscale face images of size 48×48 used for Kaggle challenge. Each image is labelled with one of the seven kinds of expressions (angry, disgust, fear, happy, sad, surprise and neutral).

2) *Emotion labels in the dataset*

0 – Anger: 4593 images
1 – Disgust: 547 images
2 – Fear: 5121 images
3 – Happy: 8989 images
4 – Sad: 6077 images
5 – Surprise: 4002 images
6 – Neutral: 6198 images

B. Evaluations

The proposed model for Emotion Recognition System is implemented by DCNN architecture. The main packages used for project interpreter Python 3.7 are Keras 2.2.4 which is an open-source neural network library written in Python. It is capable of running on top of TensorFlow 2.1.0 which is a free and open-source software library for dataflow and differentiable programming across a range of tasks.

The model contains all the layers of DCNN architecture including their Output shapes with filters used and the parameters of that layer. The parameters like biases and weights are Total params: 4,478,727, Trainable params: 4,474,759 and Non-trainable params: 3,968. The model is trained over train data of 28709 48×48 grayscale images, and tested over 7178 48×48 grayscale images, which gives test accuracy of **90.95%** after testing for 50 Epochs and Batch size 128.

Adam is an adaptive learning rate method, which means, it computes individual learning rates for different parameters. Its name is derived from adaptive moment estimation, and the reason it's called that is because Adam uses estimations of first and second moments of gradient to adapt the learning rate for each weight of the neural network.

The designed model is given below:

Table 1
Designed Model: "sequential_1"

Layer (type)	Output Shape	Param #
Conv2d_1 (Conv2D)	(None, 48, 48, 64)	640
Batch_normalization_1 (Batch)	(None, 48, 48, 64)	256
Activation_1 (Activation)	(None, 48, 48, 64)	0
Max_pooling2d_1 (MaxPooling2	(None, 24, 24, 64)	0
Dropout_1 (Dropout)	(None, 24, 24, 64)	0
Conv2d_2 (Conv2D)	(None, 24, 24, 128)	204928
Batch_normalization_2 (Batch	(None, 24, 24, 128)	512
Activation_2 (Activation)	(None, 24, 24, 128)	0
Max_pooling2d_2 (MaxPooling2	(None, 12, 12, 128)	0
Dropout_2 (Dropout)	(None, 12, 12, 128)	0
Conv2d_3 (Conv2D)	(None, 12, 12, 512)	590336
Batch_normalization_3 (Batch	(None, 12, 12, 512)	2048
Activation_3 (Activation)	(None, 12, 12, 512)	0
Max_pooling2d_3 (MaxPooling2	(None, 6, 6, 512)	0
Dropout_3 (Dropout)	(None, 6, 6, 512)	0
Conv2d_4 (Conv2D)	(None, 6, 6, 512)	2359808
Batch_normalization_4 (Batch	(None, 6, 6, 512)	2048
Activation_4 (Activation)	(None, 6, 6, 512)	0
Max_pooling2d_4 (MaxPooling2	(None, 3, 3, 512)	0
Dropout_4 (Dropout)	(None, 3, 3, 512)	0

Flatten_1 (Flatten)	(None, 4608)	0
Dense_1 (Dense)	(None, 256)	1179904
Batch_normalization_5 (Batch	(None, 256)	1024
Activation_5 (Activation)	(None, 256)	0
Dropout_5 (Dropout)	(None, 256)	0
Dense_2 (Dense)	(None, 512)	131584
Batch_normalization_6 (Batch	(None, 512)	2048
Activation_6 (Activation)	(None, 512)	0
Dropout_6 (Dropout)	(None, 512)	0
Dense_3 (Dense)	(None, 7)	3591
Total params: 4,478,727		
Trainable params: 4,474,759		
Non-trainable params: 3,968		

C. Results

DCNN evaluation accuracy for test images (from dataset or user) is: 90.95%

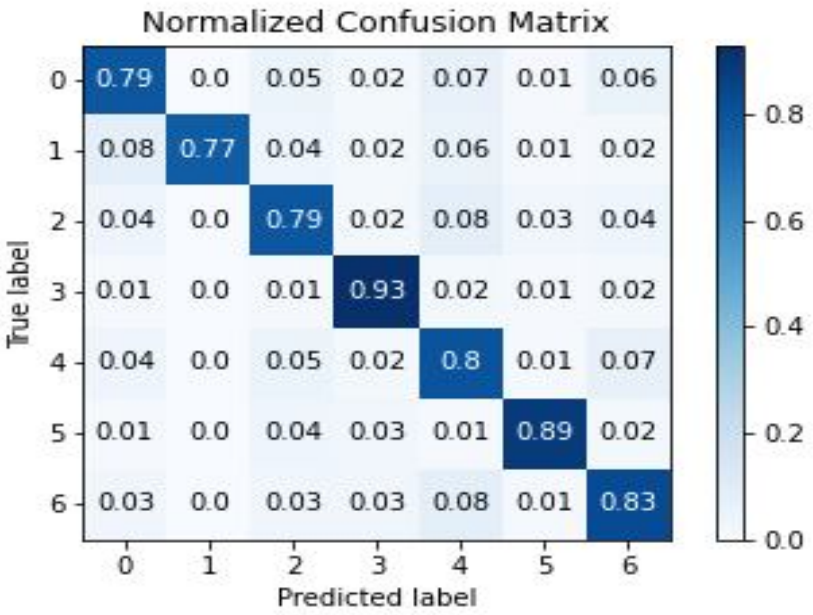
Comparison of various algorithms of CNN is shown in Table 2 below.

Table 2

Method	Accuracy
CNN-LSTM	58.62%
LSTM	63.79%
HCNN	69.10%
FRR-CNN	70.63%
OUR PROPOSED METHOD	90.95%

As shown in Table above, various algorithms and their accuracy are mentioned. The combination of CNN-LSTM [5], Traditional CNN (SHCNN) [3], and FRR-CNN [2] deals with CNN architecture. But the proposed method improves CNN architecture and deals with pre-processing of dataset FER2013.

The recognition result is provided for the FER2013 dataset is represented by a confusion matrix as shown in Fig. 5.



Put Fig. 5.

Fig. 6 illustrates emotions recognized by the proposed method, these are some successful recognition even when the subjects are partially occluded by a notebook as shown in Fig. 6 (e). We can conclude those emotions, such as ‘‘happy”, “sad”, ‘‘surprise’’, and “neutral” are easy to recognize. This conclusion is in line with the results that are indicated by the confusion matrix, as illustrated in Fig. 5.

Fig. 6(a)

Fig. 6(b)

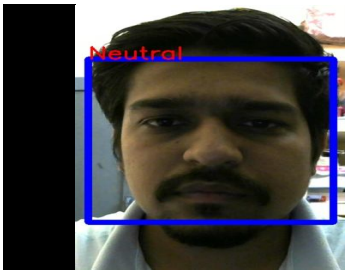


Fig. 6(c)

Fig. 6(d).

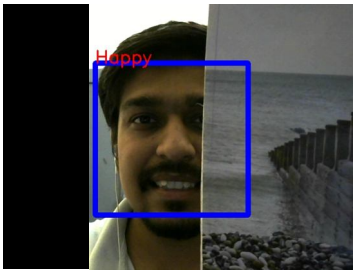


Fig. 6(e).

Fig. 6(f).

Fig. 7 illustrates some cases of failed recognition of facial expressions, which are represented as unknown (nothing) or a wrong label. Fig. 7 (a) is the condition where the proposed method sometimes cannot recognize emotion. Sometimes the proposed method predicts wrong for example, emotion as shown in Fig. 7 (b) predicts “sad” but the emotion is “angry”.

Fig. 7 (a)

Fig. 7 (b)

We developed a Desktop and Web Application for user convenience. Fig. 8 and Fig. 9 shows some snapshots of both applications.

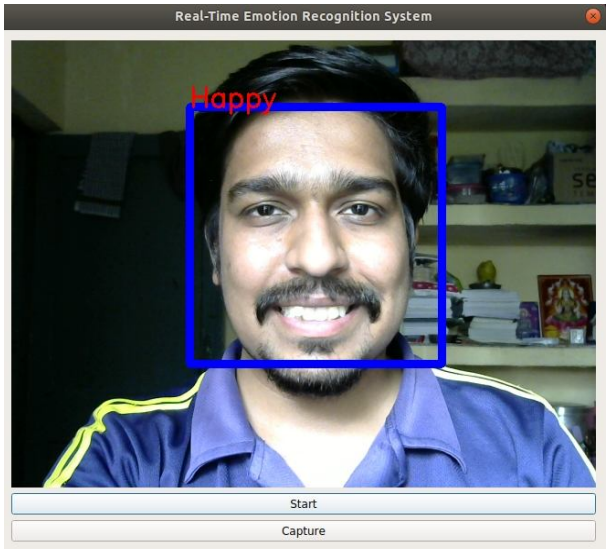


Fig.8

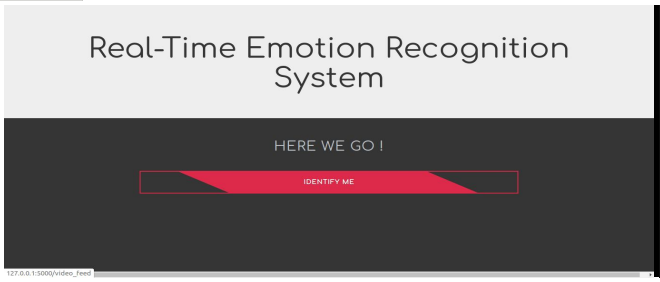


Fig. 9(a).

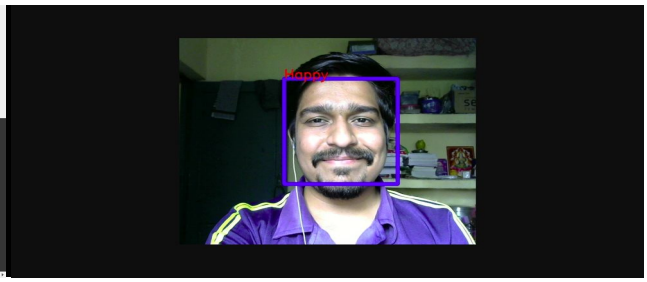


Fig. 9(b).

As shown in Fig. 8, desktop Application is implemented in PyQt5 Python Framework that is widely used to build desktop applications. As shown in Fig 9 (a) and Fig. 9 (b), web Application is implemented in Flask, a web-based framework that is a light-weight framework is used to build web applications

IV. CONCLUSIONS

In this paper, the proposed method is to pre-processing of the FER2013 dataset (extracts relevant features) and to improve the architecture of traditional CNN. The Evaluation accuracy goes to 90.95% and it is way more than other CNN architecture as shown in Table 2. Emotion recognition technology has come a long way in the last twenty years. Today, machines can automatically verify identity information for secure transactions, for surveillance and security tasks, and access control to buildings etc. These applications usually work in controlled environments and recognition algorithms can take advantage of the environmental constraints to obtain high recognition accuracy. However, next-generation emotion recognition systems are going to have a widespread application in smart environments [4] – where computers and machines are more like helpful assistants [5]. To achieve this goal a DCNN is proposed to avoid potential errors in real-time emotion recognition based on the traditional convolutional neural network. Using a camera with high frame rates, the influence of noise from the environments can be reduced. Moreover, because of the DCNN method, emotion recognition results become more robust from frame to frame. As a result, the accuracy of emotion recognition is improved. Experimental results have shown that the proposed Emotion recognition system is more reliable than the traditional CNN approach.

V. ACKNOWLEDGMENT

This work was supported by Prof. A. S. Kamble (Department of Computer Engineering) of Modern Education Society's College of Engineering (MESCOE). Furthermore, we indebted to our Head of Department Dr. (Mrs) N. F. Shaikh, whose constant encouragement and motivation inspired us to do our best.

REFERENCES

[1] Keng-Cheng Liu, Chen-Chien Hsu, Wei-Yen Wang, and Hsin-Han Chiang, "Real-Time Facial Expression Recognition Based on CNN", International Conference on System Science and Engineering (ICSSE), 5th September 2019.

[2] Siyue Xie and Haifeng Hu, "Facial expression recognition with FRR-CNN", IEEE, 16th February 2017.

[3] Si Miao, Haoyu Xu, Zhenqi Han, and Yongxin Zhu, "Recognizing Facial Expressions Using a Shallow Convolutional Neural Network", IEEE Access, 5th June 2019.

[4] Huibin Li, Jian Sun, Zongben Xu, and Liming Chen, "Multimodal 2D+3D Facial Expression Recognition with Deep Fusion Convolutional Neural Network", IEEE, 8th June 2017.

[5] Tzuu-Hseng S. Li, Ping-Huan Kuo, Ting-Nan Tsai, and Po-Chien Luan, "CNN and LSTM Based Facial Expression Analysis Model for a Humanoid Robot", IEEE Access, 11th July 2019.

[6] Biao Yang, Jinmeng Cao, Rongrong Ni, and Yuyu Zhang, "Facial Expression Recognition Using Weighted Mixture Deep Neural Network Based on Double-Channel Facial Images", IEEE Access, 15th December 2017.

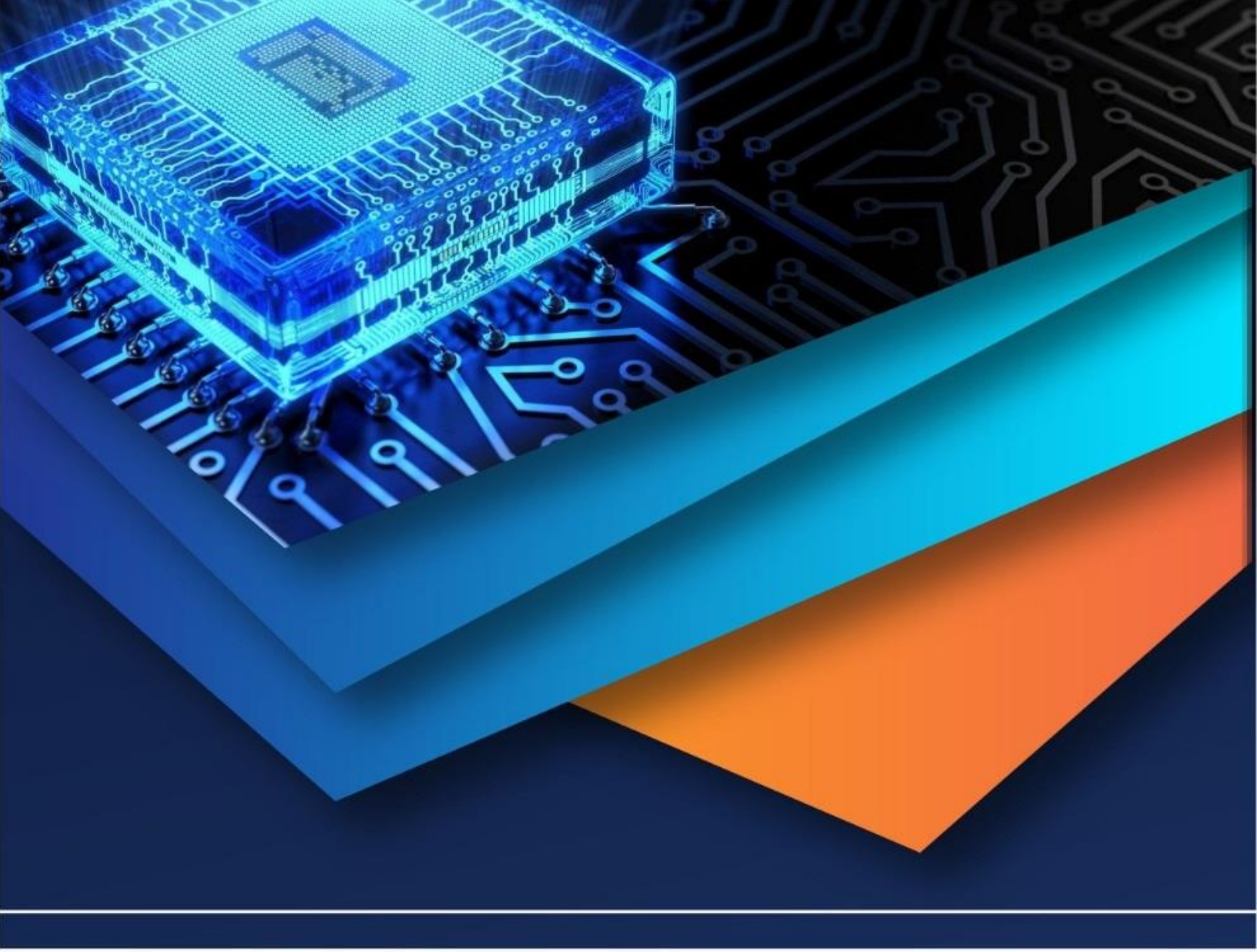
[7] Hongli Zhang, Alireza Jolfaei and Mamoun Alazab, "A Face Emotion Recognition Method Using Convolutional Neural Network and Image Edge Computing", IEEE Access, 28th October 2019.

[8] T.H. Kim, C. Yu, and S.W. Lee, "Facial expression recognition using feature additive pooling and progressive fine-tuning of CNN", IEEE, 11th May 2018.

[9] Yong Li, Jiabei Zeng, Shiguang Shan, and Xilin Chen, "Occlusion aware facial expression recognition using CNN with attention mechanism", IEEE Transactions on Image Processing, 8th August 2018.

[10] Luz Santamaria-Granados, Mario Munoz-Organero, Gustavo Ramirez-González, Enas Abdulhay, and N. Arunkumar, "Using Deep Convolutional Neural Network for Emotion Detection on a Physiological Signals Dataset (AMIGOS)", IEEE Access, 23rd November 2018.

[11] Vishal Dilip Parandwal, Suyog Udayshankar Gadage, Suyash Bajirao Gorde, Rohit Vasant Surve, Prof. A. S. Kamble, "A literature survey on live emotion recognition system using cnn", International Journal of Creative Research Thoughts (IJCRT), ISSN:2320-2882, Volume.8, Issue 4, pp.1627-1632, April 2020.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY