ABSTRACT
Over the past assorted decades, one of the most agile spheres of research is the emotion recognition. The grounds of this study is to come forward with finely grained real-time learning model of emotion recognition comprising of phases like feature extraction, subset feature and emotion classifier. Haar Cascade technique is being used for identifying the input figure to identify the characteristic values. This work also aims to classify human emotions like fear, neutral, angry, surprise, happy and sad by using two classifiers; Long Short Term Memory (LSTM) and Convolutional Neural Network (CNN) by deploying a finely grained real time emotion recognition algorithm using virtual markers. Initially, Haar Cascade was used for eyes and face discernment, then Neighborhood Difference Features (NDF) were extracted and virtual markers are being positioned on designated places on identified face using VGG16 approach. The facets are validated using cross-validation and forwarded to CNN and LSTM classifiers. Experimental results of loss function, confusion matrix, classification report of training and testing had proved that the model proposed give consistent output with the real time facial expressions.

KEY WORDS: DEEP LEARNING (DL); CONVOLUTIONAL NEURAL NETWORKS (CNN); INTERNET OF THINGS (IOT); FACIAL LANDMARK; VGG16; LONG SHORT TERM MEMORY (LSTM).

INTRODUCTION
The recognition and classification of human emotion has recently been a hot topic, worldwide. The emotional and mental state of a human can be identified by the expressions on his face. While communication, a human expresses numerous emotions which vary in complexity and intensity thus may have different meaning. Broadly, the main categories of human emotions are angry, fear, happy, neutral, sad, surprise and disgust. Moreover, the mixed emotions can also be expressed by the permutation of these basic emotions. Recognizing facial emotions can bridge the communication gap among human and machine.

The introduction of the Internet of Things (IOT), smart environment at various places like hospital, cities, hotels and smart cities has laid much importance on automatic emotion detection. Presently, there are number of Natural Language Processing (NLP) based Intelligent Personal Assistants (IPA's) which are in use worldwide such as Alexa, Siri, Cortana etc. Facial Emotion Recognition (FER) has number of application areas such as identification of anti-social elements, detecting the drowsiness of drivers in real time, in helping the medical practitioners to analyze the synthetic human expressions and identification of mental disorder, video surveillance, security systems, auto face capturing and many more.

FER is a challenging issue to be solved in the computer vision community, thus it is very essential to deploy a system which categorizes human emotion with high accuracy and minimum loss in feature extraction. The pipeline of our proposed framework is depicted in Figure
1, which explains the part of the training and testing phase and how the facial expressions are categorized into different emotions. The continuing part of the paper comprises of section 2 which discuss the recent related work. Proposed approach along with methodology has been presented in Section 3. Section 4 explains the experimental setup done for evaluating the model followed by result analysis and lastly section 5 concludes with the work done.

PROPOSED METHODOLOGY

A finely-grained facial emotion recognition system has been proposed where the emotional state of the user is being monitored in real-time. Each image goes through the preprocessing steps of the module and after completion of all the steps it gives recognized emotional state of the user as the output. The proposed framework comprises of robust features which filters irrelevant and redundant information from the identified face. It is also expandable to classify complex emotions. The proposed model can be divided into three modules, listed below:

- Haar Cascade Face Detection
- Neighborhood Difference Features Extracted
- VGG16 Emotion State Classification

PROPOSED METHODOLOGY

A finely-grained facial emotion recognition system has been proposed where the emotional state of the user is being monitored in real-time. Each image goes through the preprocessing steps of the module and after completion of all the steps it gives recognized emotional state of the user as the output. The proposed framework comprises of robust features which filters irrelevant and redundant information from the identified face. It is also expandable to classify complex emotions. The proposed model can be divided into three modules, listed below:

- Haar Cascade Face Detection
- Neighborhood Difference Features Extracted
- VGG16 Emotion State Classification

In the first step, categorical data containing the labeled images of seven basic human emotions is being collected and for that purpose we have used FER2013 dataset for training and testing the model. Secondly, we have pre-processed the data by balancing the unbalanced data, rescaling, resizing the images, normalizing, and adding data- argumentation; that is the process of adding modified copies of already existing data like randomly rotating images at a particular angle, cropping them, shear shifting and flipping images. The pre-processed data is then split into two parts testing and training part. The training part is of 75% ratio and the testing part is in the ratio of 25%. Then we have performed Data Standardization by converting the structure of disparate data into a data format common to all images in the dataset. After splitting the dataset and performing data standardization we have developed Convolutional Neural Network model.

Figure 2 shows the self descriptive flow diagram for real time facial expression recognition. For detecting the emotion on the face of a person in real-time, the real time images as input has been taken from the live camera and their coordinates are extracted. Further, for face detection Haar-Cascade technique is being used. On successful face detection, the image is converted into grayscale. A rectangular frame is created over the face of the person using the Region of Interest (ROI) extracted coordinates. The processed image is fed to the model for classification of emotions. The image in the ROI is then rescaled and resized using cv2 in accordance to the shape and size acceptable by the CNN model. The model then categorizes the image into one of the seven emotion categories and the emotion having maximum probability is stored in result and the result is displayed on the label.

Algorithm 1: Real Time Facial Emotion Detection

Input: face_image
Output: Emotion

Begin
Step 1: model = model_from_json()
Step 2: model.load_weights()
Step 3: input image from the camera
Step 4: detect face in image Haar-Cascade.detectMultiScale()
if face.image.sum() != 0:
go to step 5.
else:
go to step 3.
Step 5: image = image_to_grayscale()
Step 6: create ROI for (a,b,c,d) in face:
where a, b, c and d are the face coordinates
draw_rect ((a,b),(a+c, b+d))
image_resize()
Step 7: model.predict() output
Step 8: result = max(prediction)
Step 9: putText(prediction_label)
Step 10: end
**Proposed Algorithm:** A finely grained algorithm for real-time facial emotion detection has been explained in this section. The inputs are the various expressions expressed on the face. As an output, this algorithm will classify the emotion on the input image.

According to this algorithm, the real-time images are taken as input from the camera. Using Haar-Cascade, if the frame identifies any image then that image is converted to grayscale for training else the frame re-identifies for an image. After converting an image into grayscale, create a rectangular ROI box which is created around the face on camera frame. The obtained image is then passed to the model for making the predictions on seven different emotions. The model then produce the probability values of all the identified emotions with maximum probability among the seven different classes will be displayed as the output on the label.

**Model Definition:** Convolution Neural Network: The mathematical operation is used on two functions, namely \( f \) and \( g \) and produces a third function \((f * g)\) which explains the procedure of modifying the shape using other functions known as convolution. It is mainly used in digital signal processing, functional analysis, matrices, image and signal processing. It is a sliding window function applied onto matrix. The convolution is applied to images for transformation; represented in matrix as shown in Figure 3. A digital image is stored as matrix of numbers where all the numbers signifies the brightness of a single pixel; ranging from 0 to 255. According to the RGB channel, the color picture comprises of three matrices associated with three color channels i.e. red, green and blue. In the images, the convolutional neural network is used to look for different patterns using mathematical multiplication. Two matrices are taken as image matrixes are multiplied to obtain a signifying feature. A filter or a kernel corresponds to a feature that is being looked in an image like horizontal or vertical edge detection filters, shape detection, color detection etc. This kernel is applied to image and subsequent feature map values.

A digital image is stored as matrix of numbers where all the numbers signifies the brightness of a single pixel; ranging from 0 to 255. According to the RGB channel, the color picture comprises of three matrices associated with three color channels i.e. red, green and blue. In the images, the convolutional neural network is used to look for different patterns using mathematical multiplication. Two matrices are taken as image matrixes are multiplied to obtain a signifying feature. A filter or a kernel corresponds to a feature that is being looked in an image like horizontal or vertical edge detection filters, shape detection, color detection etc. This kernel is applied to image and subsequent feature map values.

\[
G(x,y) = (f * h)[x,y] = \sum_{k} \sum_{i} k[i,k] f[x-i, y-k]
\]

(1)

where, \( G \) is the output matrix or image, \( f \) is an input image, \( h \) is kernel and the index of resultant matrix is marked with \( x \) and \( y \). The kernel is placed on the image over the selected pixel; which is multiplied pair wise and finally summed up. Then, the next pixel value is selected and so on this process is repeated till last image.

\[
h, w = \left( \frac{g_{h} - k_{h} + 2p}{s} + 1 \right), \left( \frac{g_{w} - k_{w} + 2p}{s} \right)
\]

(2)

where, \( w \) represents width of an image, \( h \) represents height of an image, \( k \) is kernel, \( g \) is input image, \( p \) is padding size, \( s \) represent stride

**Model Evaluation And Results**

3.1. **Loss Function:** Typically, a neural network is trained using the stochastic gradient descent method and perfect weights for model is calculated using back propagation method of error algorithm. The model with a given set of weights is used to make prediction. The errors for those predictions are calculated. The gradient evaluation seeks to change the weights so that the model gives minimum error. The error function or loss function is used in model optimization which is needed to minimize over the iterations of back propagation. The loss function used in model compilation is the categorical cross entropy, as it is a categorical classification task. The binary cross entropy is calculated.

\[
loss = -\frac{1}{N} \sum_{i=1}^{N} y_i \cdot \log(y_i') + (1 - y_i) \cdot \log(1 - y_i')
\]

(3)
Sharma et al.,

Where yi is ith value of model output, yi is corresponding true value and N is the total number of values. To calculate binary cross entropy loss function consider a model using sigmoid function in output which predicts the following values

\[ y'i = [0.8, 0.4, 0.1] \]
\[ yi = [1, 1, 0] \]

According to the above formula the loss would be

\[
\text{loss} = -(1\log(0.8) + 0\log(0.2) + 1\log(0.39) + 0\log(0.6) + 0\log(0.1) + 1\log(0.9))
\]
\[ = 0.541/3 \]
\[ = 0.18 \]

Categorical Cross entropy can be calculated as follows:

\[
\text{loss} = -\sum_{i=1}^{N} y_i \cdot \log (y'_i) \tag{4}
\]

For the dataset taken, the accuracy obtained in 60 epochs. We have achieved an accuracy and loss of 91.03% and 0.373 on the training dataset, and 66.58% and 1.667 on the testing dataset as shown in Figure 4. So, the model is analyzed based on the two factors i.e. model loss and model accuracy for each epoch for training and testing data. The analysis has been depicted graphically in Figure 5 and Figure 6. The system has the capability to detect the face landmarks in real time. The various facial features are then identified by change in these landmark points whenever the person facing the camera expresses any emotion. The system calculates the landmark variations and predicts the emotion on the face.

### Table 1. Confusion matrix for Training set

<table>
<thead>
<tr>
<th></th>
<th>Angry</th>
<th>Disgust</th>
<th>Fear</th>
<th>Happy</th>
<th>Neutral</th>
<th>Sad</th>
<th>Surprise</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angry</td>
<td>518</td>
<td>51</td>
<td>533</td>
<td>1045</td>
<td>726</td>
<td>648</td>
<td>438</td>
</tr>
<tr>
<td>Disgust</td>
<td>80</td>
<td>9</td>
<td>54</td>
<td>103</td>
<td>71</td>
<td>68</td>
<td>51</td>
</tr>
<tr>
<td>Fear</td>
<td>604</td>
<td>56</td>
<td>539</td>
<td>1048</td>
<td>720</td>
<td>674</td>
<td>456</td>
</tr>
<tr>
<td>Happy</td>
<td>984</td>
<td>107</td>
<td>954</td>
<td>1847</td>
<td>1319</td>
<td>1207</td>
<td>797</td>
</tr>
<tr>
<td>Neutral</td>
<td>666</td>
<td>66</td>
<td>633</td>
<td>1269</td>
<td>869</td>
<td>886</td>
<td>549</td>
</tr>
<tr>
<td>Sad</td>
<td>637</td>
<td>67</td>
<td>671</td>
<td>1222</td>
<td>901</td>
<td>733</td>
<td>559</td>
</tr>
<tr>
<td>Surprise</td>
<td>424</td>
<td>48</td>
<td>426</td>
<td>828</td>
<td>605</td>
<td>511</td>
<td>329</td>
</tr>
</tbody>
</table>
3.3 Classification Report for Training and Test Set: The visualizer of classification report depicts the F1, recall, precision, and supports score for designed model. The visualization of classification metrics per class basis has been done in classification report. It provides a deeper intuition of the classifier behaviour over global accuracy, which can mask the number of functional weaknesses in one class of multiclass problem.

In FER 2013, support for class disgust is very low (436) whereas for class happy is as high as 7215 images. The imbalanced support in the training data indicates the structural weaknesses in the model. Table 6 shows the classification report for training set and Table 7 shows the classification report for testing set.

CONCLUSION AND FUTURE WORK

In this paper, the limitations of existing emotion recognition system have been analyzed. Considering the limitations of the existing model, a new model has been proposed and implemented for emotion recognition on the basis of facial recognition in virtual learning environments, at the same time their efficiency and accuracy are also considered. HAAR Cascade technique is used to extract facial features and Region of Interest (ROI) identifies all kinds of emotions through the CNN model, and achieved a combination of efficiency and accuracy. It is applicable to real time images.

The application of emotion recognition in virtual learning environments is one of the most researched topics. This system also aims to assist the Intelligent Personal Assistants (IPAs), medical practitioners and physically disabled people. In addition to this, it will be beneficial to the researchers and scholars to explore new dimensions in this field. For future work, techniques can be used to extract more features from Residual Network, Dense Net and Inception Network.

REFERENCES

Sharma et al.,


