FACE RECOGNITION

Desai Divyeswar Reddy(200050033)

Kethavath Sai Yaswanth(200050061)

Reddy Bhavana(200050117)

Varre Suman Chaitanya(200050153)

Objective

The main objective of the project is to build a custom CNN model that can recognize the faces and post the appropriate attendance into the database

1.1 Dataset Creation

Cv2 module is used to capture the live images of a person as the dataset(training data), whenever a new person is added in data it captures 50 images using cv2 module and it stores them in a folder called image_data. This captures only faces.The images dimensions are 100*100.All this dataset is further stored into a csv file where the image data is mapped to respective class(student id)

1.2 Developing CNN model

A convolution neural network is a Deep Learning algorithm which can take in an input image assign learnable weights and biases to various aspects ,objects in the image and be able to differentiate one from the other

Essential layers to Develop CNN:

The CNN have several different kernels consisting of trainable parameters which can convolve on a given image spatially to detect features like edges and shapes. These high number of filters essentially learn to capture spatial features from the image based on the learned weights through back propogation and stacked layers of filters can be used to detect complex spatial shapes from the spatial features at every subsequent level. Hence they can successfully transform the given image into a highly abstracted representation which is easy for predicting

Different layers that are used to build an efficient CNN:

1) Convolution: Convolution is a unique idea that enable us to convert unstructured image data into structured image.The image is represented by a digital matrix of numbers.Now,using the filters,we sweep through the digital matrix of image (one filter block at a time).We move the pixel based on stride.Once the filters moves through the image , a summarized digital image is formed,it is also called feature map or the activation map.In these layer we extracted or mapped the most important features from the original image using filter

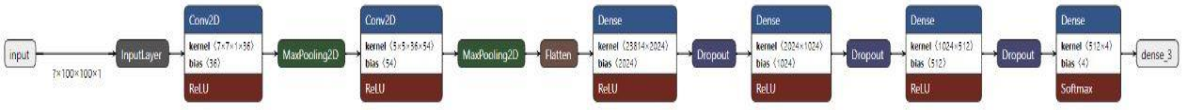
2) Max Polling: After Convolution ,we extract the most important features from the image.This step helps to reduce the dimensionality of the image data.There is a max-polling filter that is passed through the data and the maximum value in that filter is selected as the final fearture.The bigger the filter the more compressed the image information is.

3) Flattening: This step is used to extract the most important features from an image inside CNN.We just unfold the max-pooled data and convert it into a vector,simply by arranging the numbers in one dimension

These data can be fed to a fully connected Dense layer and learn it against the labels.

1.3 Architecture of Proposed CNN

The proposed CNN model contains of 12 layers. Which of them are Two-Dimensional Convolutional Layer(Conv2D),Max Pooling Layer, Dense Layer.The architecture takes a image with shape (100,100,1) and results the class label of the image as prediction



```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_4 (Conv2D)           (None, 94, 94, 36)        1800

 max_pooling2d_4 (MaxPooling  (None, 47, 47, 36)       0
 2D)

 conv2d_5 (Conv2D)           (None, 43, 43, 54)        48654

 max_pooling2d_5 (MaxPooling  (None, 21, 21, 54)       0
 2D)

 flatten_2 (Flatten)         (None, 23814)             0

 dense_8 (Dense)             (None, 2024)              48201560

 dropout_6 (Dropout)         (None, 2024)              0

 dense_9 (Dense)             (None, 1024)              2073600

 dropout_7 (Dropout)         (None, 1024)              0

 dense_10 (Dense)            (None, 512)               524800

 dropout_8 (Dropout)         (None, 512)               0

 dense_11 (Dense)            (None, 5)                 2565

=================================================================
Total params: 50,852,979
Trainable params: 50,852,979
Non-trainable params: 0
```

1.4 Training the CNN

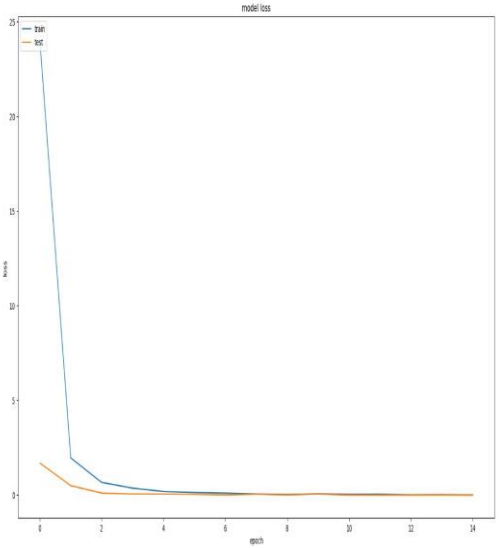
The proposed Convolution Neural Network is trained on the data containing num_clases(No. of students) classes. Which is divided in the ratio of 80:20 into train and test samples

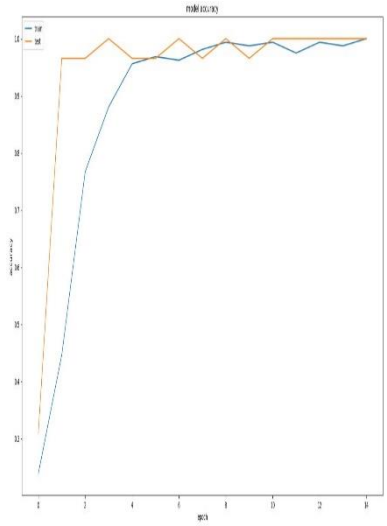
The following parameters are used in Training process:

- Optimizer - Adam
- Learning Rate -0.0001
- Loss function -Cateogeorical cross entropy
- Batch Size -5
- Epochs -15

The accuracy and the loss of the CNN model during the training wrt epochs on train and test data are represented in below figure.It is

evident that with increase in number of epochs the accuracy is increasing and loss is decreasing and at a certain number of epochs, both the loss and accuracy do not change further



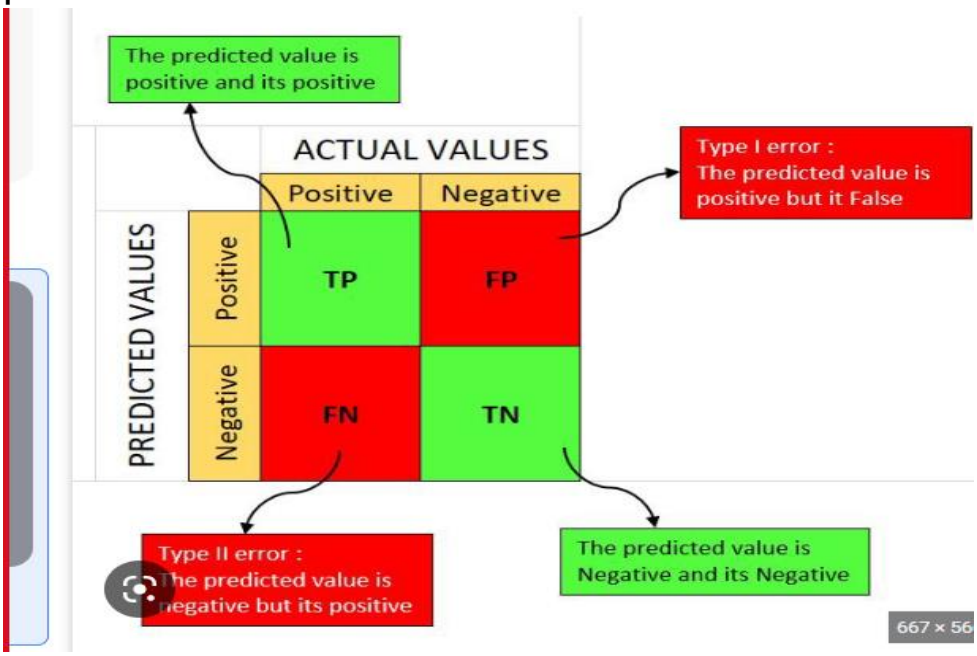


1.4 Evaluating CNN Model

Model evaluation metrics are required to quantify model performance. Some metrics, such as precission-recall,are used for evaluation

- Confusion Matrix :

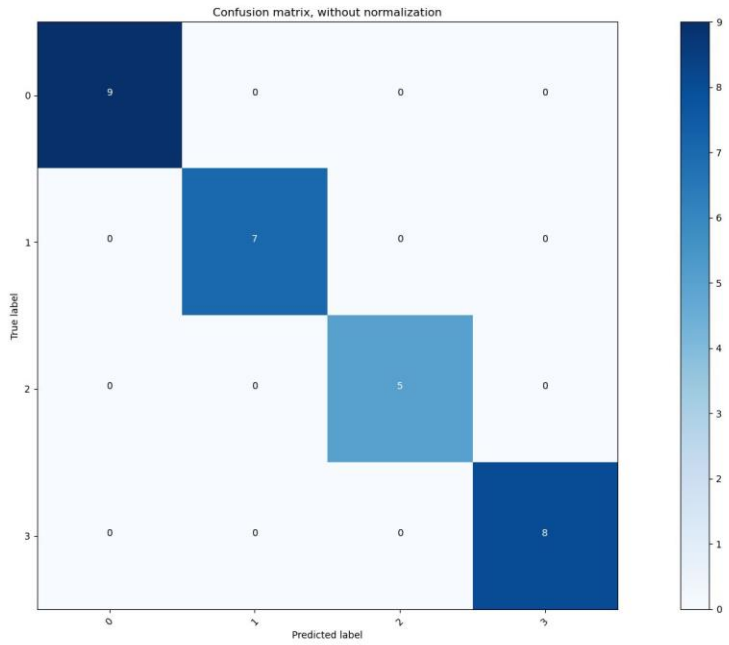
  A confusion matrix is a technique for summarizing the performance of a classification model.



- Precision: Precision is a valid choice of evaluation metric when we want to be very sure of our prediction
  Precision = (TP)/(TP+FP)
- Recall : Another very useful measure is recall, which answers a different question: what proportion of actual Positives is correctly classified
  Recall = (TP)/(TP+FN)
- F1 Score : The F1 Score is a number between 0 and 1 and is the harmonic mean of precision and recall
  F1 score=(2*Precision*Recall)/(Precision+Recall)

1.5 Results

Confusion Matrix:



Precision and Recall for each classes:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	9
1	1.00	1.00	1.00	7
2	1.00	1.00	1.00	5
3	1.00	1.00	1.00	8
accuracy			1.00	29
macro avg	1.00	1.00	1.00	29
weighted avg	1.00	1.00	1.00	29

test los 0.0061
test acc 1.0000