Smile Detection Using CelebFaces Dataset

Group members :

Name : Saffat Rafiq Raaz Email : 160204041@aust.edu
Name : Tauhidul Islam Email : 160204031@aust.edu
Name : Noor Islam Sunny Email : 160204038@aust.edu

➢ Introduction :

- **Definition of task :** Facial expressions resulting from movement of the facial muscles convey various human emotions and states. Smile is one of the basic human facial expressions which correlates to joy or happiness. In this project we will try to implement simple deep neural network models which can detect smiles from facial images of celebrities. Such smile detection models can be used in various fields like using computer vision to detect human emotions, in camera based technologies like smartphones, digital cameras etc.

- **Motivation :** We are doing this project to examine how efficient simple deep neural networks can be in detecting complex human facial emotions and what settings give the best outputs. This type of comparative analysis can further help in creating more efficient and low cost facial expression recognizers. Using computers to detect human emotions can pave the way to creating more complex systems which can autonomously perform various tasks based on people's emotional state. Auto selfie takers in smartphones, categorization of human emotion from pictures etc. application rely on such models.

- **Why the task is challenging :** Human faces are complex and can express a lot of different emotions by moving facial muscles. Every human face is very different from others and facial structures vary a lot. So generalizing a facial emotion into digital data is very complex. Without neural networks it is very hard to process human smiles into digital data. There are a lot of approaches to building a neural network to detect smile but it requires a ton of experiment to find an efficient and low cost method.

➢ Related Works :

❑ List of some important related works :

[1] Required resolution | Axis Communications
[2] Shan, C. (2011). Smile detection by boosting pixel differences. IEEE transactions on image processing, 21(1), 431-436.
[3] Hand, E., Castillo, C., & Chellappa, R. (2018, April). Doing the best we can with what we have: Multi-label balancing with selective learning for attribute prediction. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 32, No. 1).

❑ Summarization :

In work [1], an extensive research has been done about the minimal resolution needed for training neural networks on human facial features. It has been found that the required minimal pixel/face is : identification in challenging conditions 80px/face, identification in good conditions 40px/face, recognition 20px/face, face detection 4px/face. As CelebFaces images are in good condition we have decided to resize our images to 48*40 pixels. This created minimum input parameters of 1920 for our smile detection models.

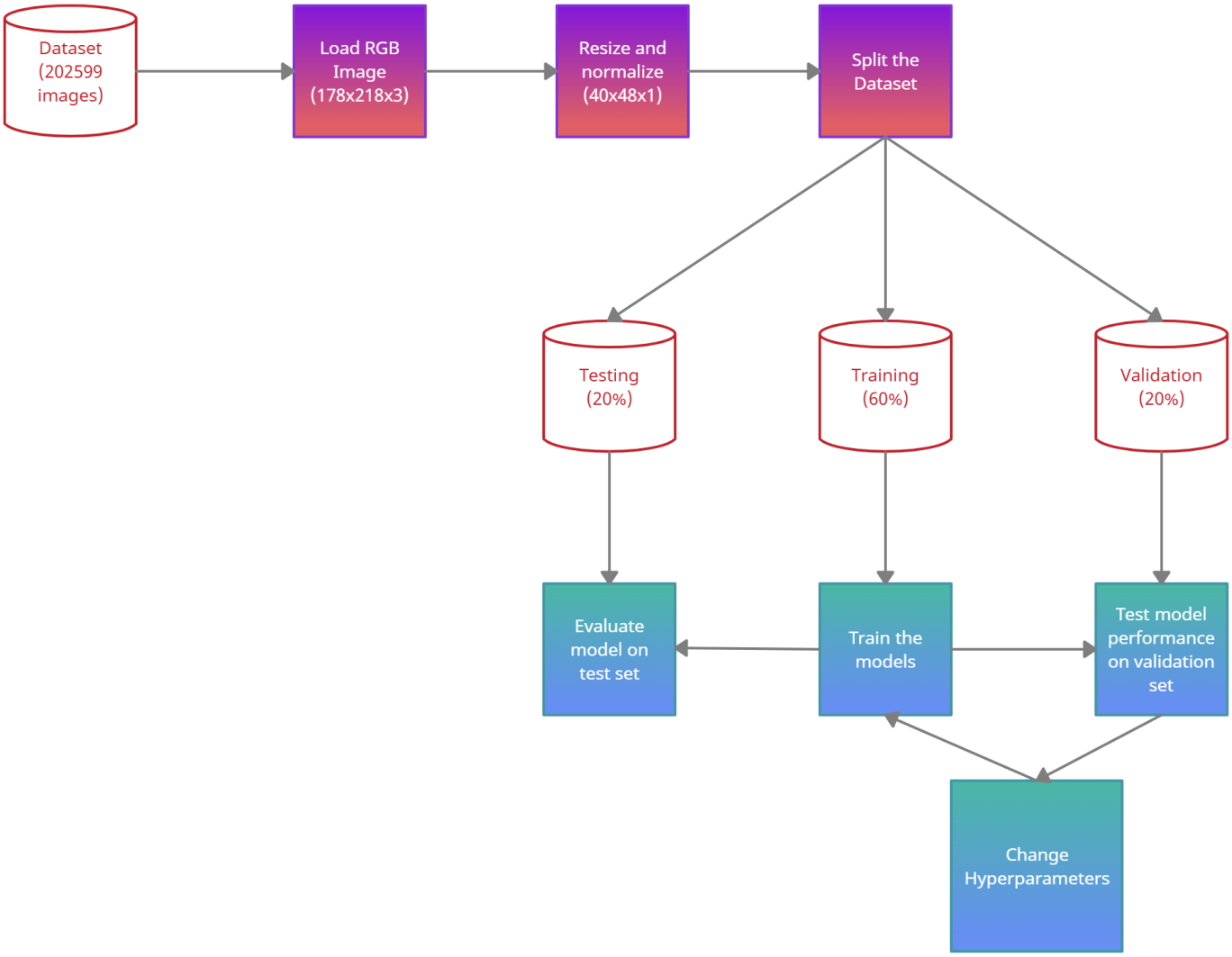
In paper [2], first the face and eyes are automatically located. The image is rotated, cropped, and scaled to ensure a constant location of the center of the eyes on the image plane. Next, the image is encoded as a vector of real-valued numbers, which is passed through a bank of filters to integrate into real-valued number.This number is then thresholded to classify the image as smiling or not smiling. Gabor features provide the accuracy of 89.55%, whereas LBP features achieve 87.10%.And with the grayscale pixel values, a linear SVM achieved the performance of 80.38%. Here filters were used to extract information. In our model we are not using any filters but rather deep neural networks.

In the paper [3], a detailed analysis has been done on CelebA dataset. Using LFWA dataset along with CelebA dataset they have detected some attributes having bias in CelebA dataset. We are working with smile attribute in our project. As this attribute had an even distribution in the dataset, the chances of having bias in training set is very low.

In most recent papers complex convolutional neural networks have been used for facial gesture recognition. But in this project we wanna test how simple deep neural networks perform in detection smiles.

➢ Project Objectives :

❑ Subtasks :



CelebFaces Attribute dataset is a resized and cropped version of CelebA dataset where images are only focused around the face region. The images are of 178*218*3 size.

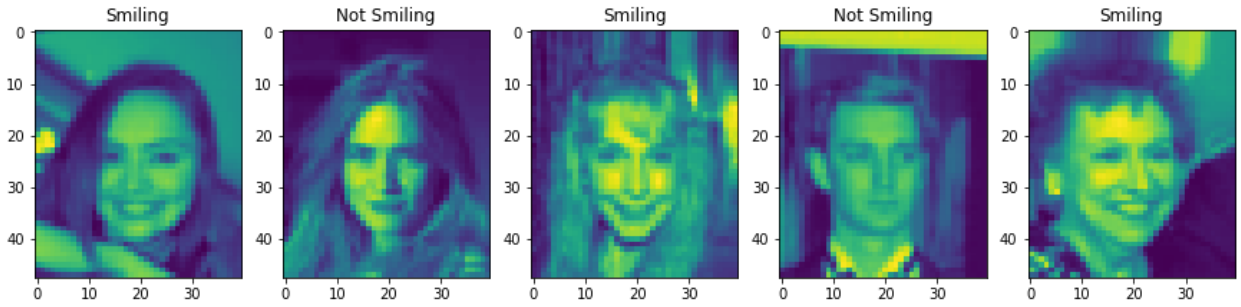
Our first subtask is loading the images. We first load the “list_attr_celeba.txt” where images and corresponding attributes are stored. We then select the ‘smiling’ attribute and discard the rest. We read the image names from the loaded text file and load corresponding images from zipped dataset.

Our second subtask is resizing and normalising the images. For our model we resize and normalize the images to 40*48*1 size and group them together with ‘smiling’ attribute values. In the text file the attribute values are numbered -1 for ‘not smiling’ and 1 for ‘smiling’. For our model predictions we convert (-1,1) values to (0,1).

We then split the dataset into 60%-20%-20% train-test-validation sets. We used the train set to train our models through forward and back propagation. We used the validation set to calculate training accuracy during training. We did not use validation set to calculate final accuracy as it may create bias in the result. We made changes to hyperparameters based on training accuracy. After getting satisfying result we finally used the testing set, which the model never saw, to get unbiased performance metrics. This ensures to give us an idea how our models can perform on unknown inputs and the generalization ability of our models.

We have used some test set images to use as dummy input on our model to generate outputs.

Dummy input :



Dummy output :

```
Predicted labels :
Picture 0 : Smiling
Picture 1 : Not Smiling
Picture 2 : Smiling
Picture 3 : Not Smiling
Picture 4 : Smiling
```

➢ Methodologies :

In this project we have used simple deep neural network to build our models. We did not use any convolution or dropout layers. We have used 4 models to train on our dataset. There were two kinds of models :

1. **Two hidden layers deep neural network models :**

Model	Learning rate	Losss function	Optimizer	Batch	Epochs	Iteration	Hidden layer activation	Neuron Count
Model 1	0.005	Cross-Entropy	Adam	100	1	1216	Tanh, ReLU	500, 100
Model 2	0.005	Cross-Entropy	Adam	100	2	2432	ReLU, ReLU	500, 100
Model 3	0.1	Cross-Entropy	SGD	100	1	1216	ReLU, ReLU	500, 100

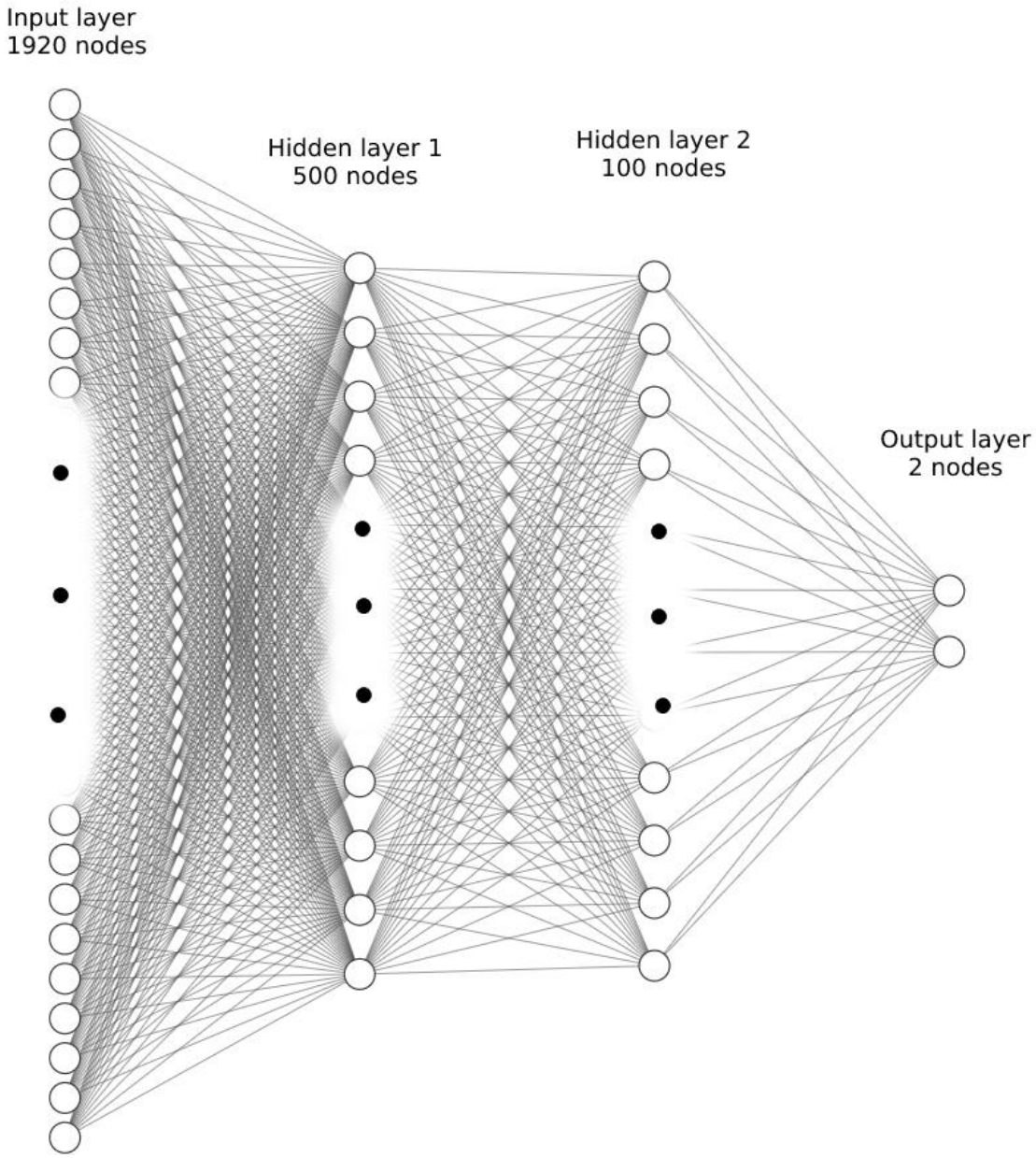


Figure : Two hidden layers deep neural network structure.

2. Three hidden layer deep neural network model :

Model	Learning rate	Losss function	Optimizer	Batch	Epochs	Iteration	Hidden layer activation	Neuron Count
Model 4	0.001	Cross-Entropy	Adam	100	2	2432	ReLU, LeakyReLU, LeakyReLU	500, 100,100

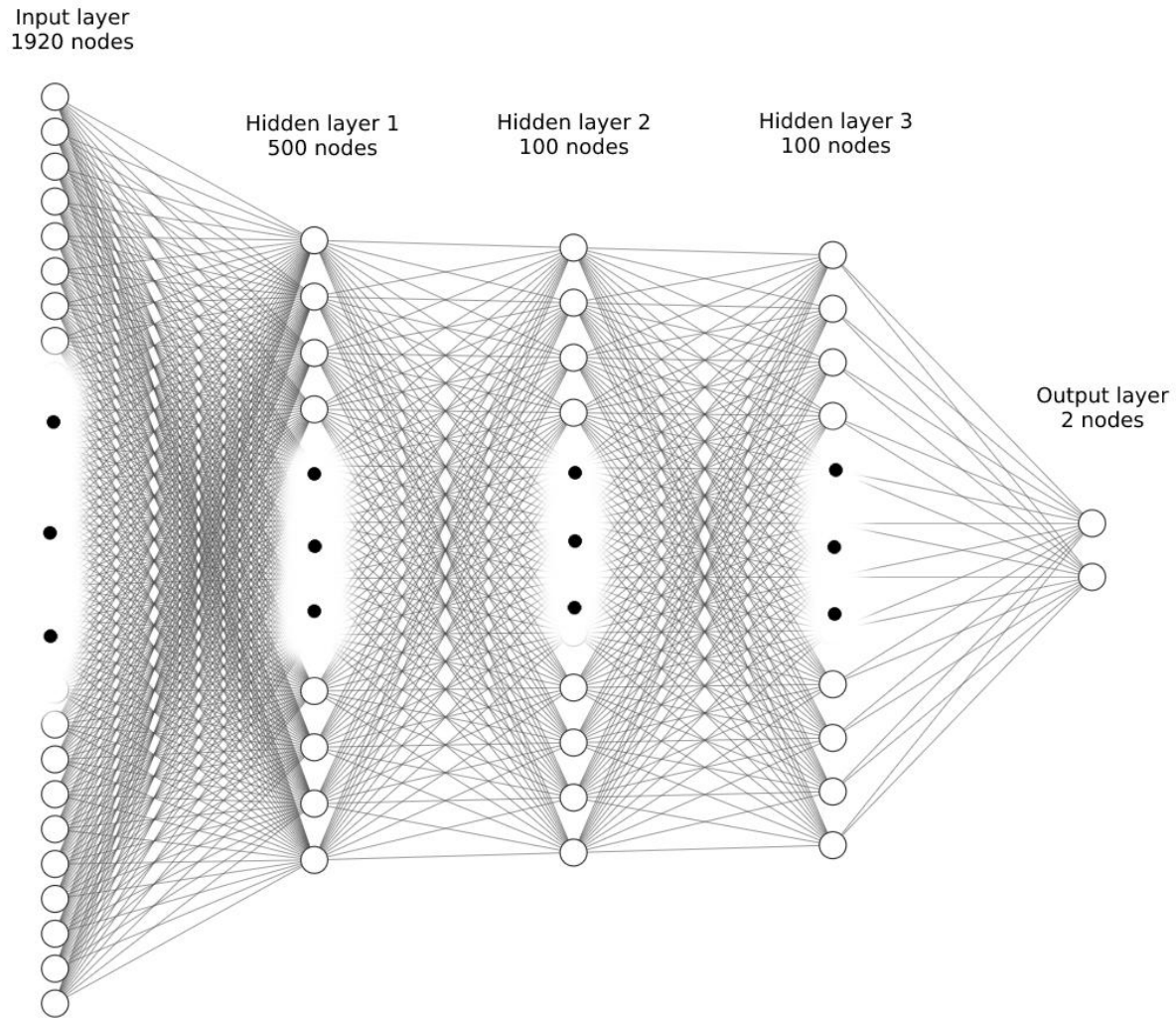
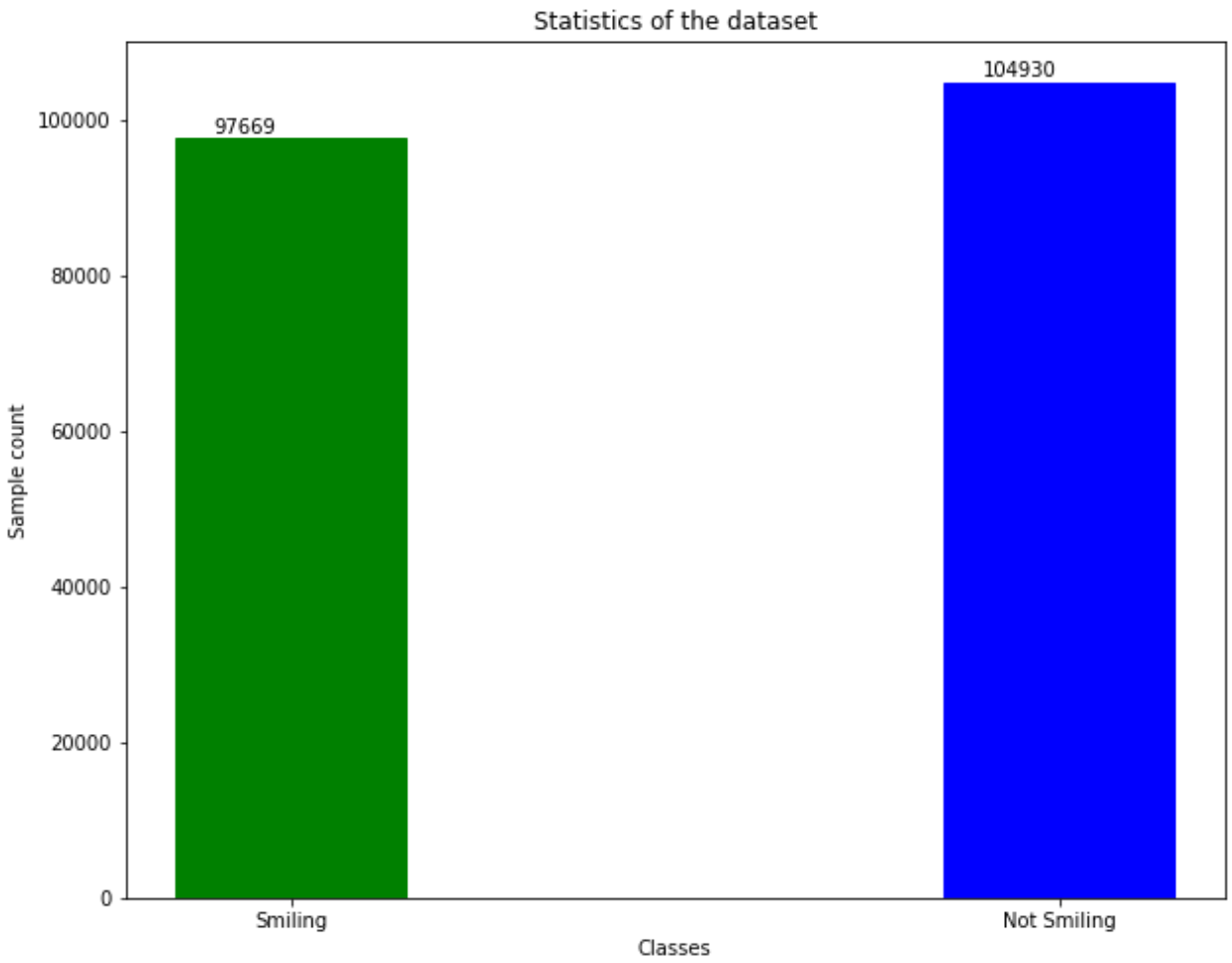


Figure : Three hidden layers deep neural network structure.

➢ Experiments :

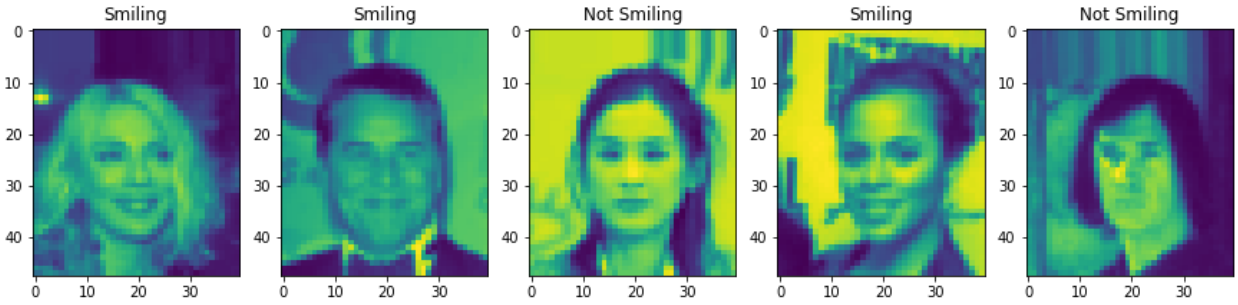
- **Dataset :** The CelebFaces dataset contains 202599 RGB images of various celebrities . The images are of 178*218*3 size. There are 97669 smiling face images and 104930 not smiling face images.



Some samples of the dataset is given below :

	filename	Smiling
0	000001.jpg	1
1	000002.jpg	1
2	000003.jpg	-1
3	000004.jpg	-1
4	000005.jpg	-1
...	...	...
202594	202595.jpg	-1
202595	202596.jpg	1
202596	202597.jpg	1
202597	202598.jpg	1
202598	202599.jpg	-1

Here, 1 value stands for smiling face and -1 stands for not smiling face. After loading the images and translating the labels we can show the samples as :



We split the dataset of 202599 images into three parts : 60% training, 20% validation, 20% testing. As a result we get 121560 training samples, 40519 testing samples and 40520 validation samples.

- **Evaluation Metrics :** We will evaluate our training performance with accuracy and loss plots with the help of validation set. We will use the test set once to evaluate the finalized trained model. We will measure the performance of the models on test set using accuracy, precision, recall and f1 score. We will calculate them by detecting true positive (TP),true negative (TN), false positive(FP), false negative(FN) samples.

  Equations for calculating accuracy, precision, recall and f1 score :

$$\text{precision} = \frac{\text{tp}}{\text{tp} + \text{fp}}$$

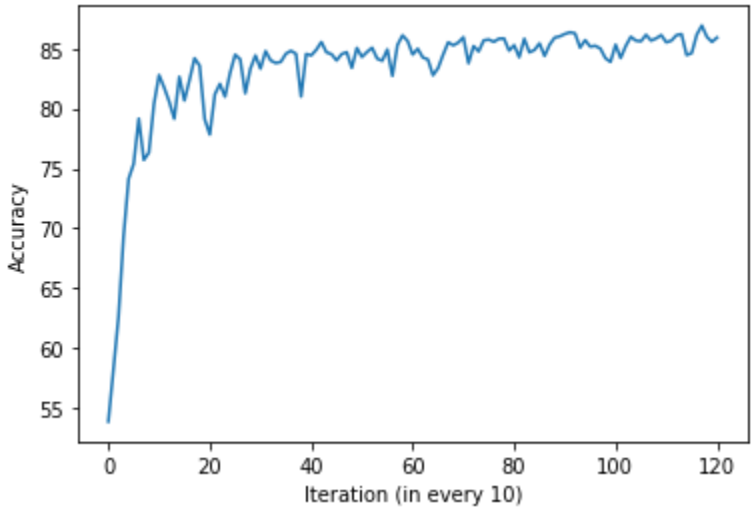
$$\text{recall} = \frac{\text{tp}}{\text{tp} + \text{fn}}$$

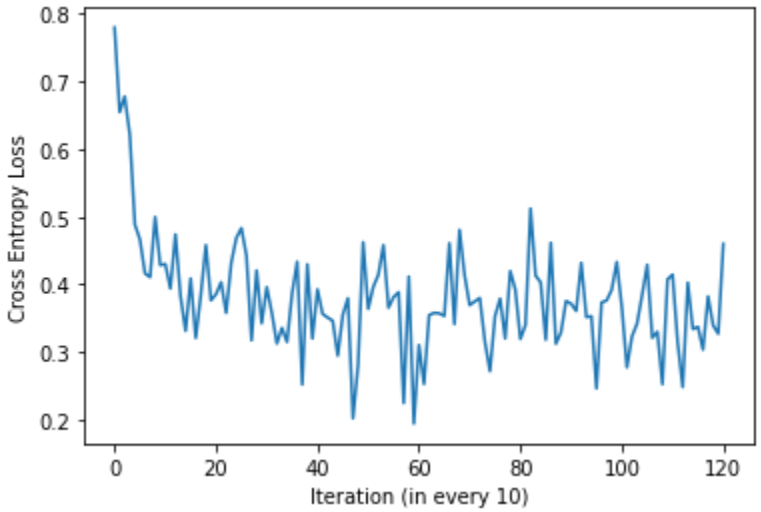
$$\text{accuracy} = \frac{\text{tp} + \text{tn}}{\text{tp} + \text{tn} + \text{fp} + \text{fn}}$$

$$\text{F}_1 \text{ score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

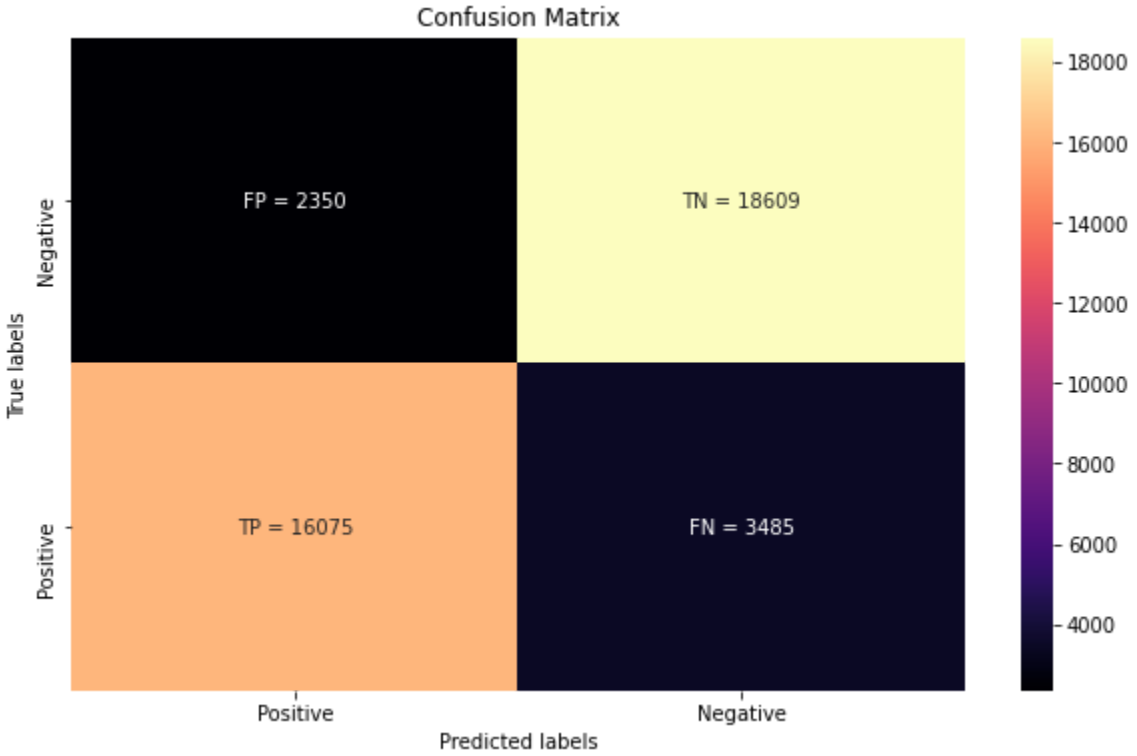
❑ **Results :**

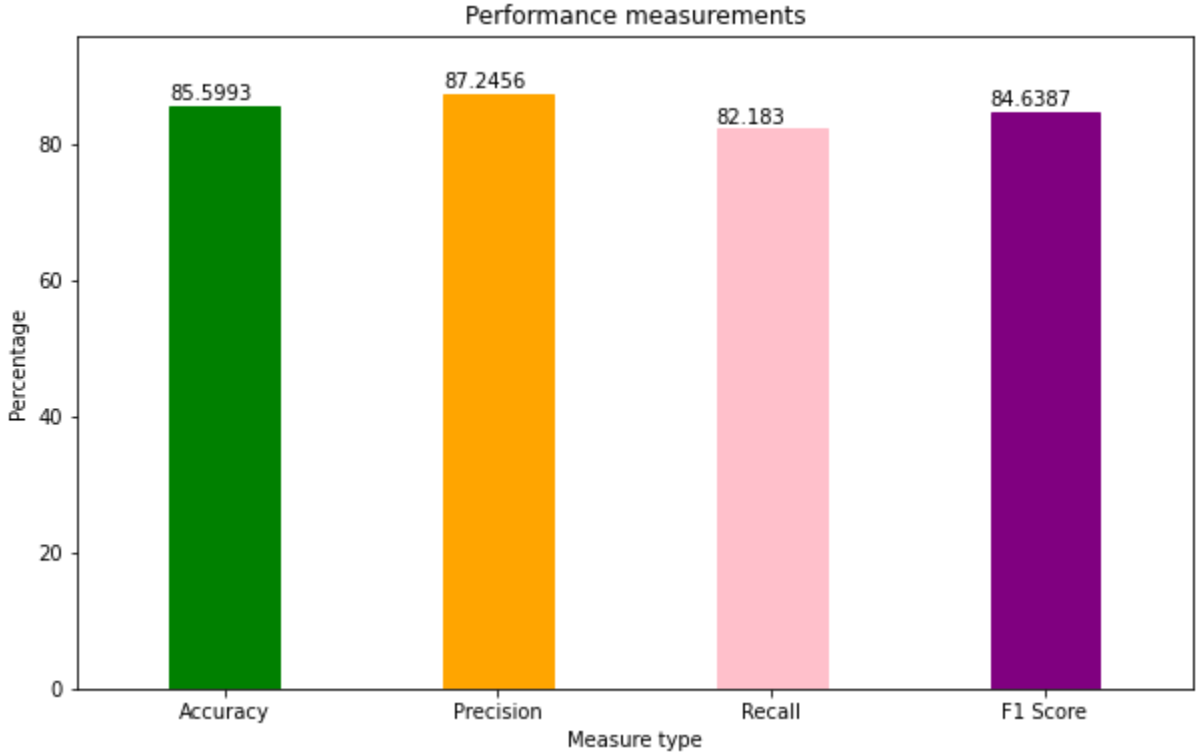
Model 1: Training accuracy : 85.92% , Training Loss : 0.46





Test :

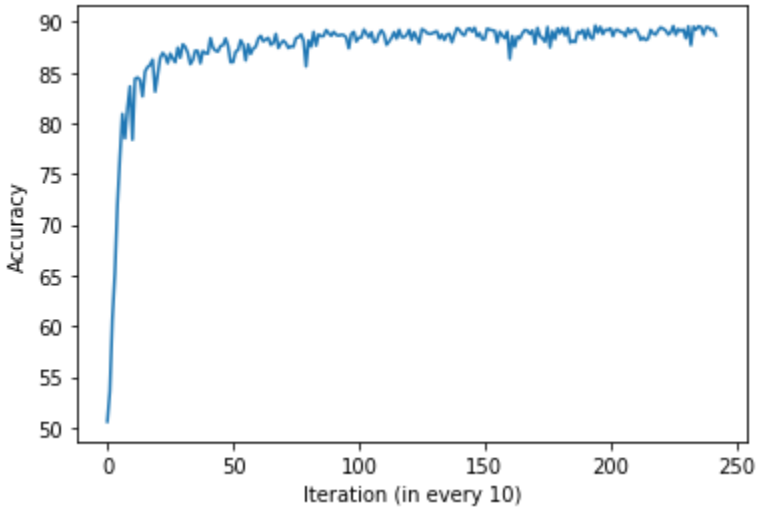


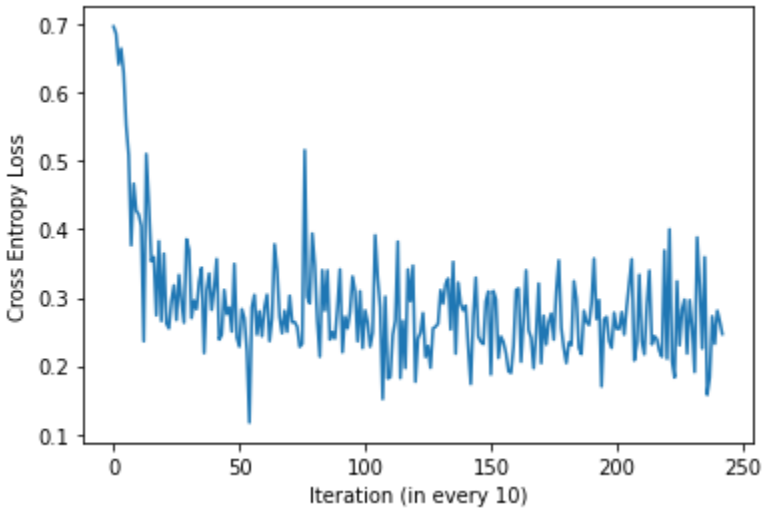


In model 1 we trained our dataset in 1 epoch with Adam optimizer and learning rate 0.005. We used Tanh and ReLU activation on hidden layers. We got

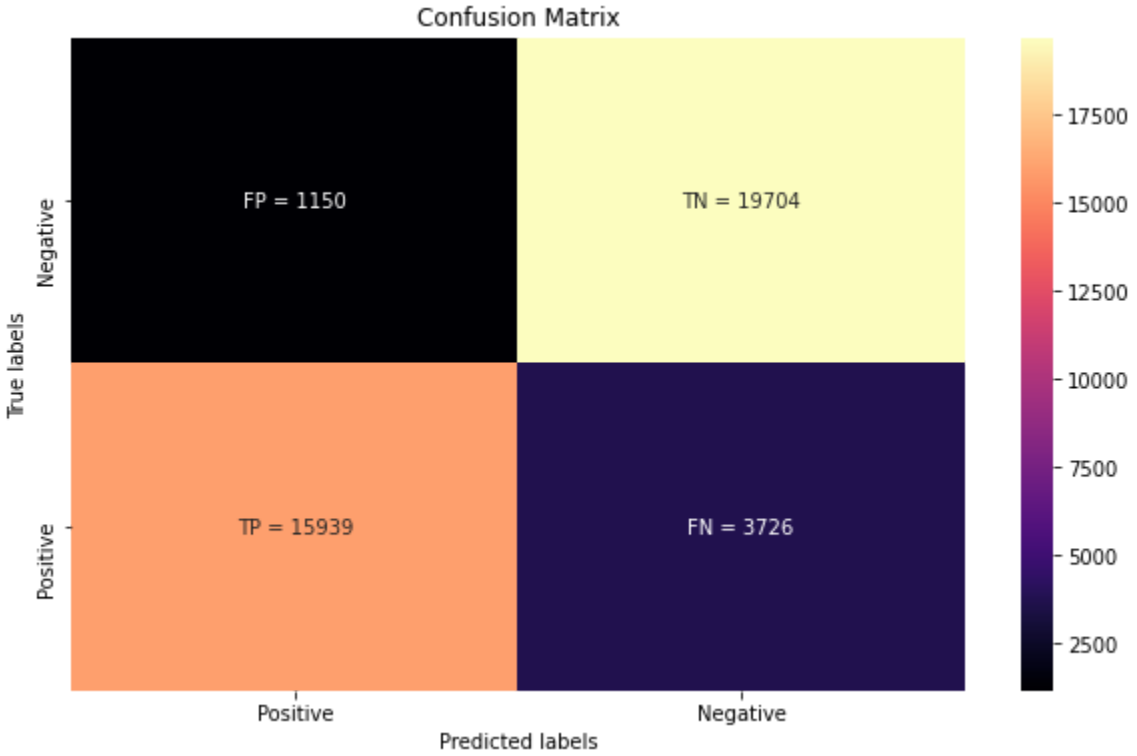
85.92% training accuracy and 85.6% testing accuracy. This shows our model will perform well on unknown samples and well generalized in detecting smiles.

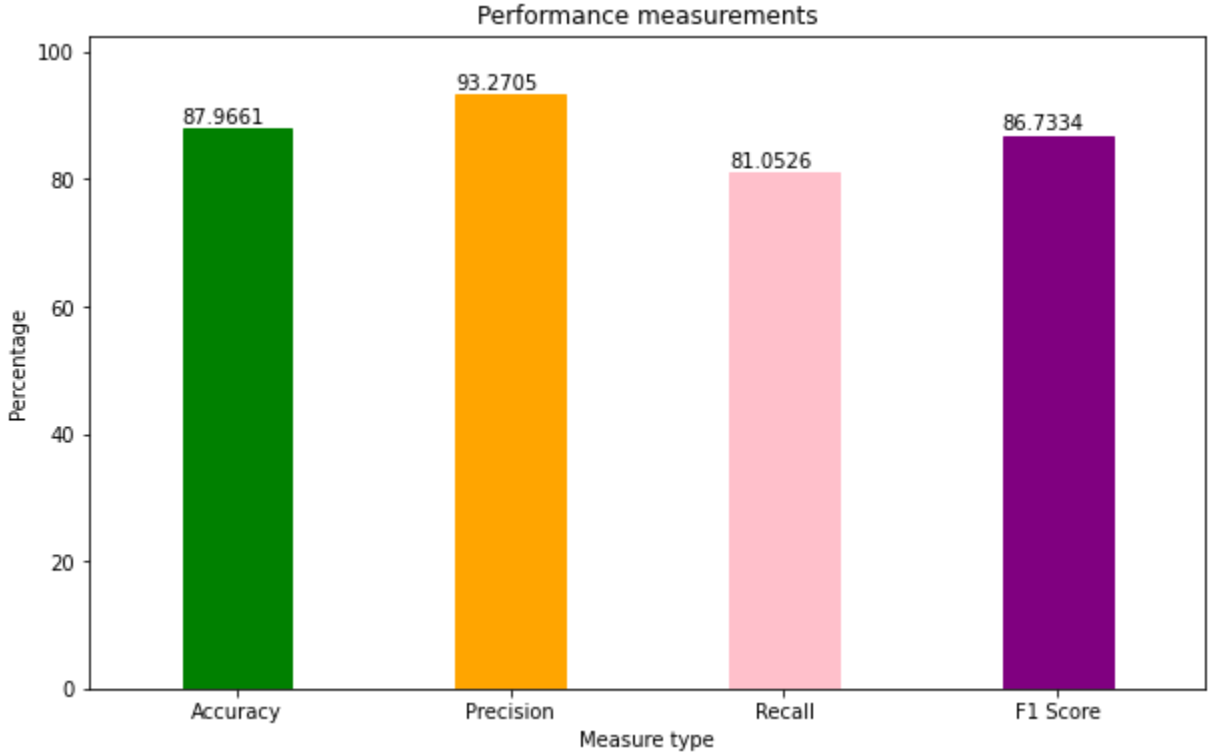
Model 2 : Training accuracy : 88.68% , Training Loss : 0.24





Test :

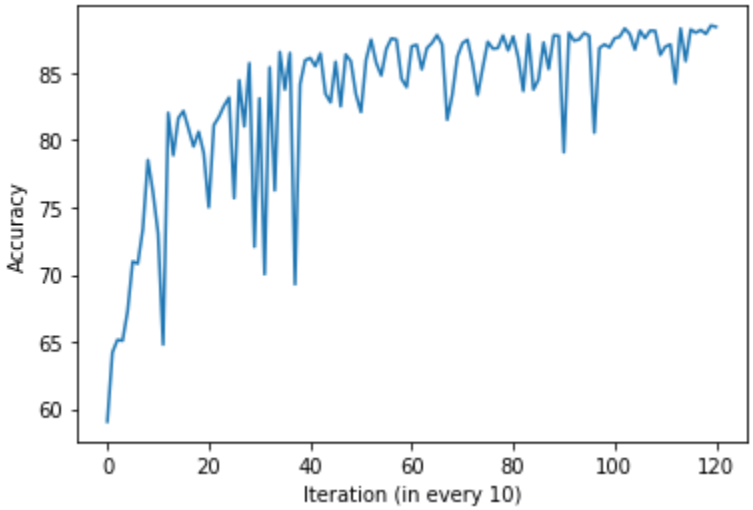


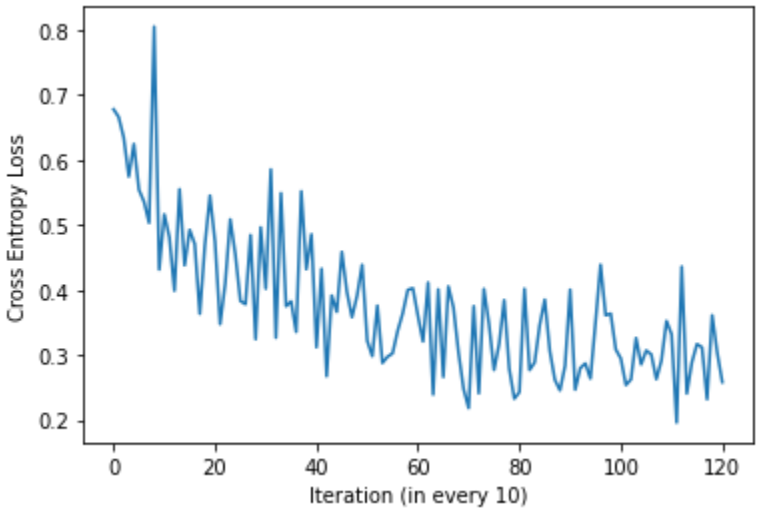


In model 2 we changed the first hidden layer activation from Tanh to ReLU. We also increased Epoch to 2. This made training time longer but we achieved

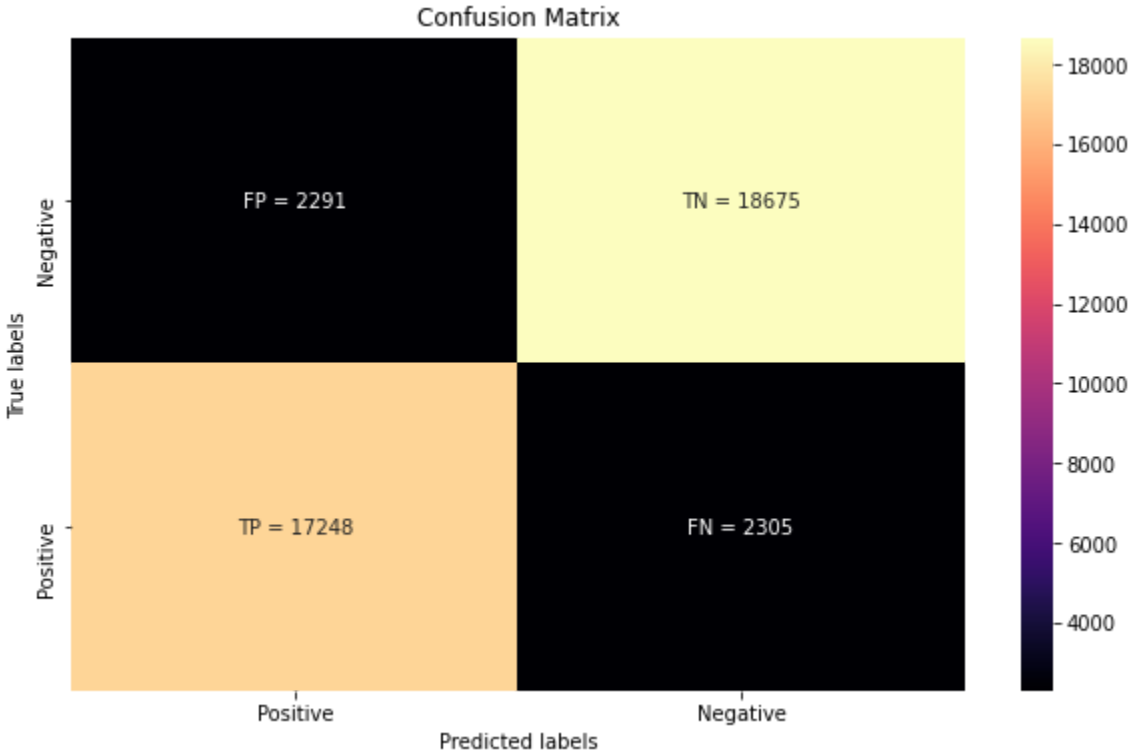
a much higher testing accuracy of 87.96%. The precision was much higher at 93.27% but recall was lower at 81.05%. Although the number of false positives decreased, the number of false negatives increased.

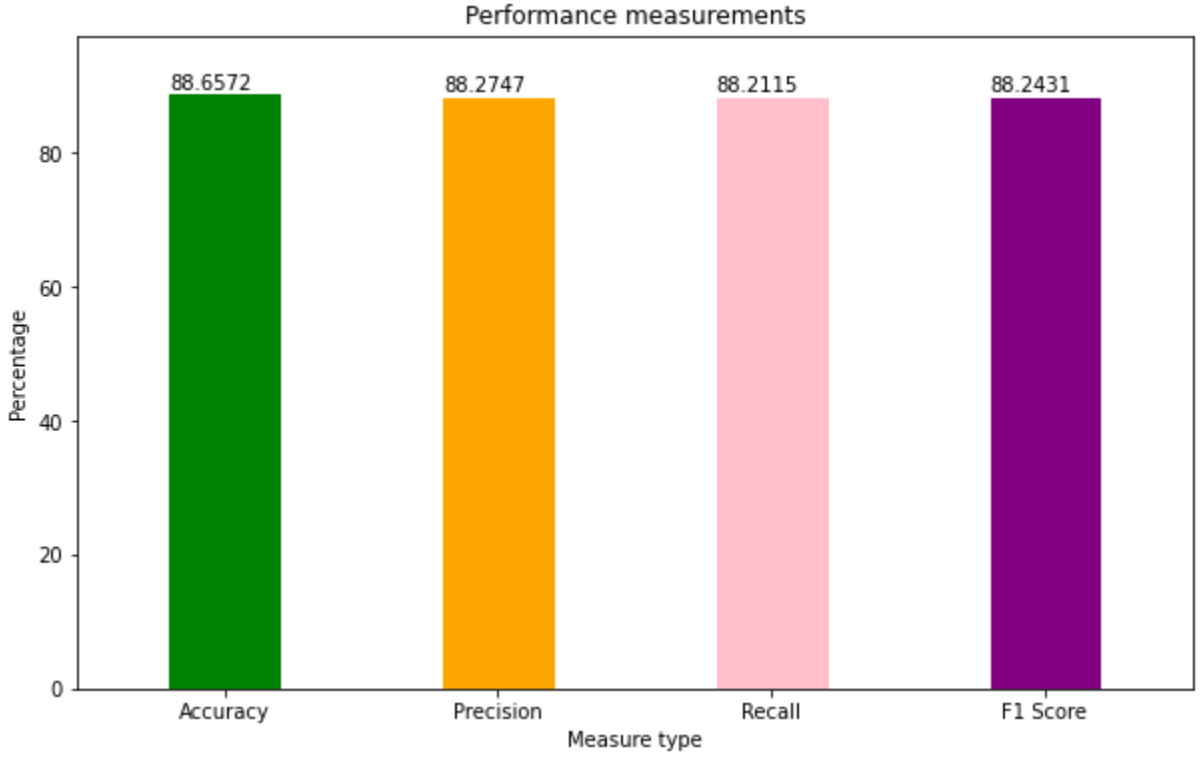
Model 3 : Training accuracy : 88.41% , Training Loss : 0.25





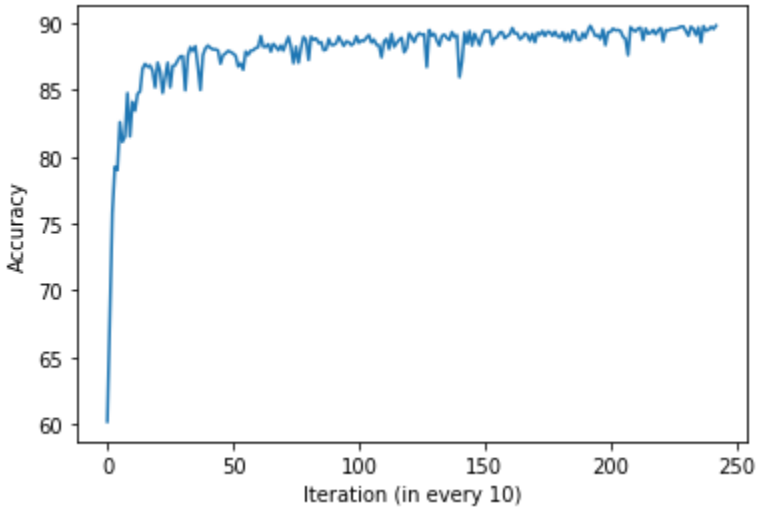
Test :

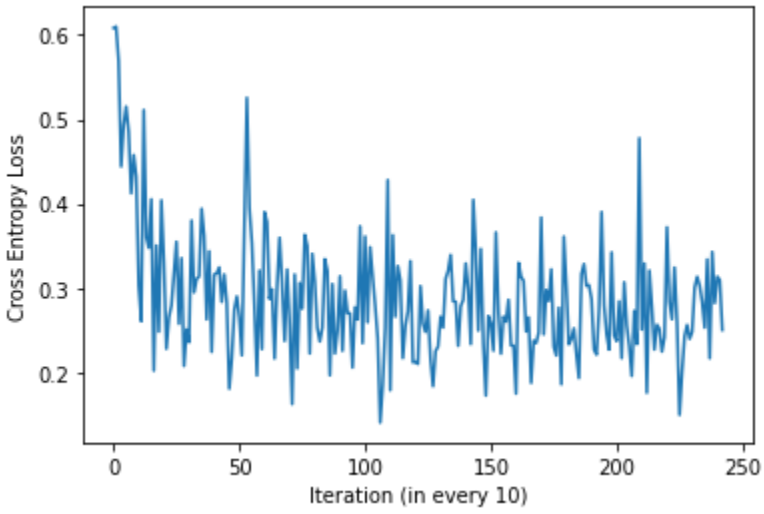

Confusion Matrix
FP = 2291
TN = 18675
TP = 17248
FN = 2305
Negative
Positive
True labels
Positive
Negative
Predicted labels
18000
16000
14000
12000
10000
8000
6000
4000


Performance measurements
88.6572
88.2747
88.2115
88.2431
Percentage
0
20
40
60
80
Accuracy
Precision
Recall
F1 Score
Measure type

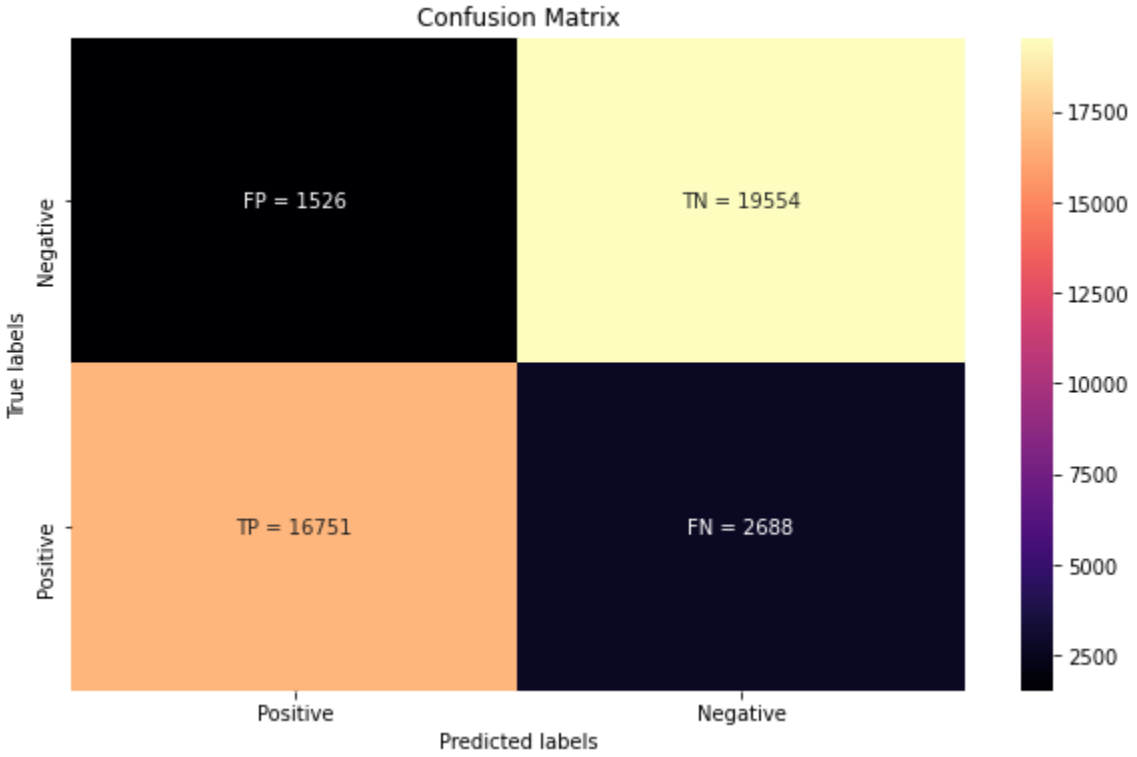
In model 3 we used SGD optimizer instead of Adam. We changed the learning rate to 0.1 as SGD needs a higher learning rate than Adam. This model performed slightly better than previous models and achieved a testing accuracy of 88.65%. From Precision and recall we can see that both values are very high means this model lowered both false positives and false negatives. Thus a high F1 score was achieved.

Model 4 : Training accuracy : 89.8% , Training Loss : 0.25





Test :


Confusion Matrix
FP = 1526
TN = 19554
TP = 16751
FN = 2688
Negative
Positive
True labels
Positive
Negative
Predicted labels
17500
15000
12500
10000
7500
5000
2500


Performance measurements
89.5999
91.6507
86.1721
88.827
100
80
60
40
20
0
Percentage
Accuracy
Precision
Recall
F1 Score
Measure type

In model 4 we used 3 layers adding a 100 node extra hidden layer. The activation functions were ReLU, LeakyReLU, LeakyReLU. We used Adam optimizer for this model with learning rate 0.001 and trained in 2 epochs. We trained for longer duration with lower learning rate achieving a stable increase in accuracy. We achieved a testing accuracy of 89.6% which is the highest of all models. Although Recall was high, precision was much higher. Only 1526 samples were false positives. Meaning this model misclassified fewer 'not smiling' images as 'smiling images'. The overall F1 score also was the highest of all the models.

➢ Conclusion :

In this project we have achieved a high accuracy of detecting smiles from images by using 2 or 3 hidden layer deep neural networks. We observed that adding another layer only increased accuracy slightly . We only need 40px/face to collect necessary features to detect smiles. Keeping image size low reduces out input parameters a lot and makes our models much faster. These models can be improved by adding convolution and dropout layers.