

# ANALYSIS REPORT

Name: Kumar Raju Bandi

Roll Number: IWM2017502

In this assignment, a face detection system was designed and implemented from scratch using the PCA algorithm and tested with the given image dataset.

Apart from this:

- a) Analysis of k vs accuracy was done.
- b) Handling of imposters in train data was also done by labelling them as “not enrolled” in prediction.

Colab Link :

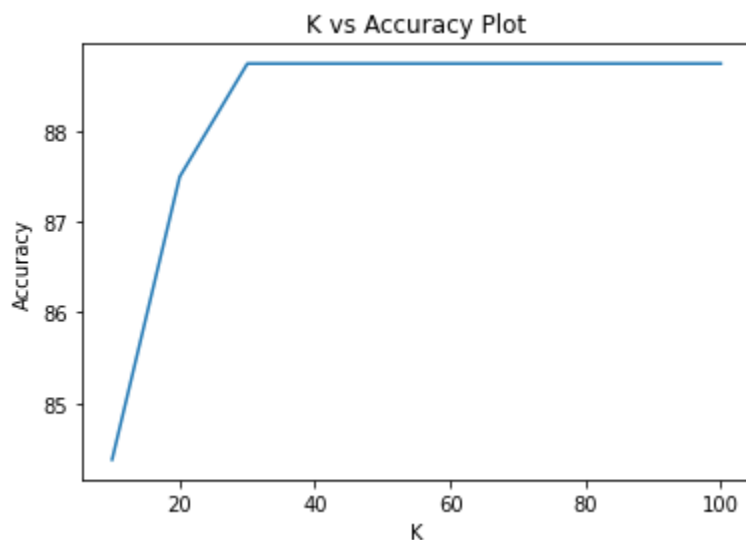
[https://colab.research.google.com/drive/1yfmlm\\_RVO\\_vCBNyGi3K-Ovig7kMo2hBN?usp=sharing](https://colab.research.google.com/drive/1yfmlm_RVO_vCBNyGi3K-Ovig7kMo2hBN?usp=sharing)

The model was implemented as discussed in the class and it was a very simple model to implement compared to others done so far.

Some really interesting analysis were drawn during the implementation and testing of the algorithm for face detection.

The accuracy achieved on the given dataset was **92.5%**. However, it is to be noted that the accuracy differs with every run of the python notebook as I am randomly shuffling the data samples before splitting them into 60:40 ratio.

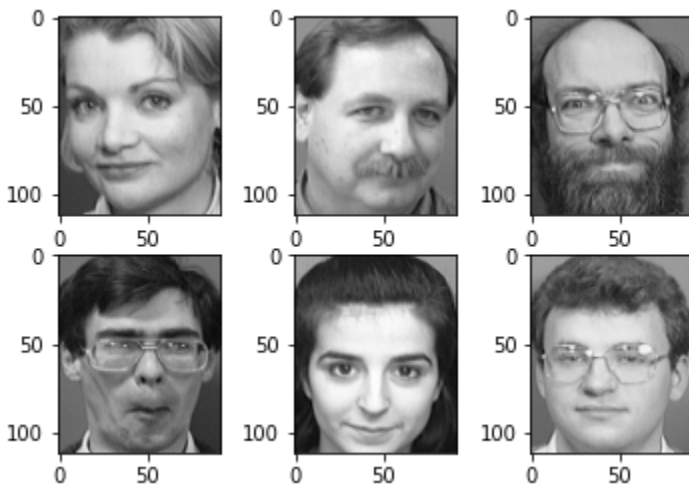
Firstly, below is the K vs accuracy plot for PCA model implemented:



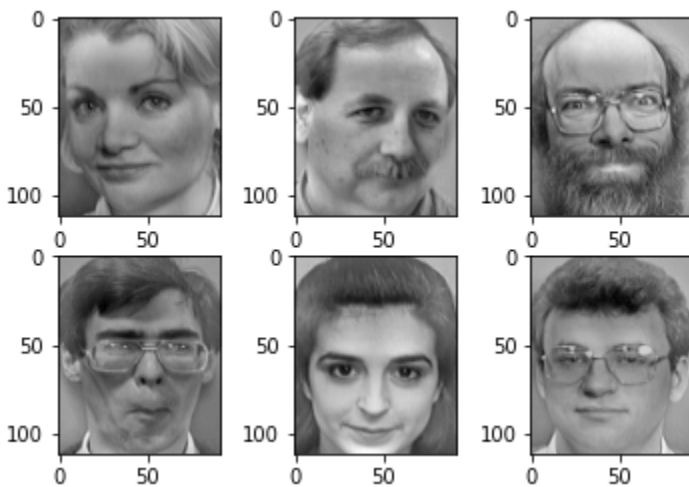
It can be observed from the above plot that from values after around 50, the accuracy is becoming stable. Thus, we can conclude that  $K = 50$  is a good and sufficient value for the model for this dataset.

Also, I have analysed the images before and after processing like centering the data for calculating the covariance matrix,  $C$ . Below are the results:

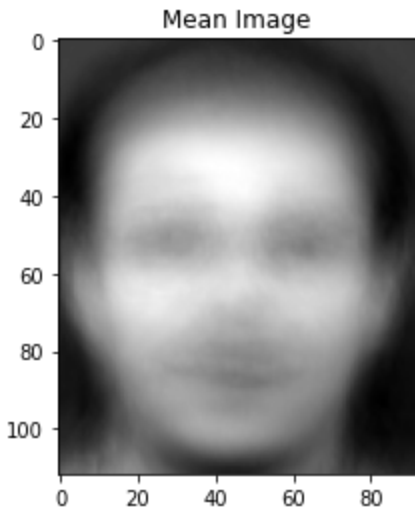
Before centering the data:



After centering the data:



The below results are depending on the mean image which I also tried to visualize and it is as follows:



This image basically represents the summary or average of all the faces in the train dataset.

I also implemented a feature to show imposters in the train data by using a threshold for similarity matching. This threshold will be used when calculating the L2 norm to match the images. I tried various values and came up with the threshold as 40000000 which did a good job in showing the imposters.

So, if any image is matching above this threshold, then the class name will be "Not Enrolled". Below is an example of lena image being passed in the test data:



The result is :

```
Class label for the above image : Not Enrolled
```