Shri Vile Parle Kelvani Mandal's

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

Approved by AICTE and Affiliated to the University of Mumbai

Department of Electronics & Telecommunication Engineering

Mini Project On

Title: **Face Recognition Using Neural Networks**

SUBMITTD BY:

Name of Student	SAP ID
Atulya Kumar	60002160052
Parth Mehta	60002160063
Lina Korda	60002178008

Teacher's Name: Prof. Vishakha Kelkar

Shri Vile Parle Kelvani Mandal’s

DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING

Approved by AICTE and Affiliated to the University of Mumbai

Department of Electronics & Telecommunication Engineering

CERTIFICATE

This is to certify that M/S.______________________________,

SAP ID ________________ of BE EXTC 1: has submitted their

Mini Project for Subject Name for the Academic Year 2019-2020.

Guide Examiner

Head of Department
EXTC Department

Index

Introduction: Face recognition involves comparing an image with database of stored faces in order to identify individual in that input image. This is where the system compares the given individual to all the other individuals in the database and gives a ranked list of matches. While humans have the innate ability to recognize and distinguish different faces for millions of years, computers are just now catching up. Here we have used neural networks, which is a powerful data modeling tool that is able to capture and represent complex input/output relationships .

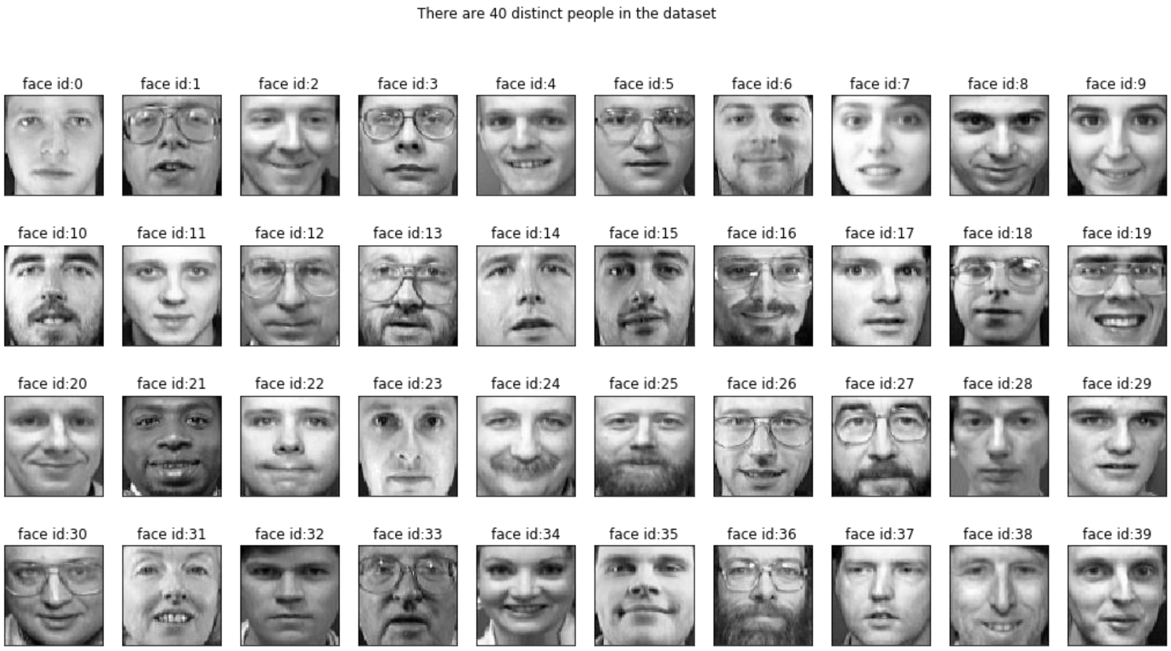
Software Used: Python 3.7

Dataset: We have used the *Olivetti* Dataset.

- There are ten different images of each of 40 distinct people
- There are 400 face images in the dataset
- Face images were taken at different times, varying lightning, facial express and facial detail
- All face images have black background
- The images are gray level
- Size of each image is 64x64

Working:

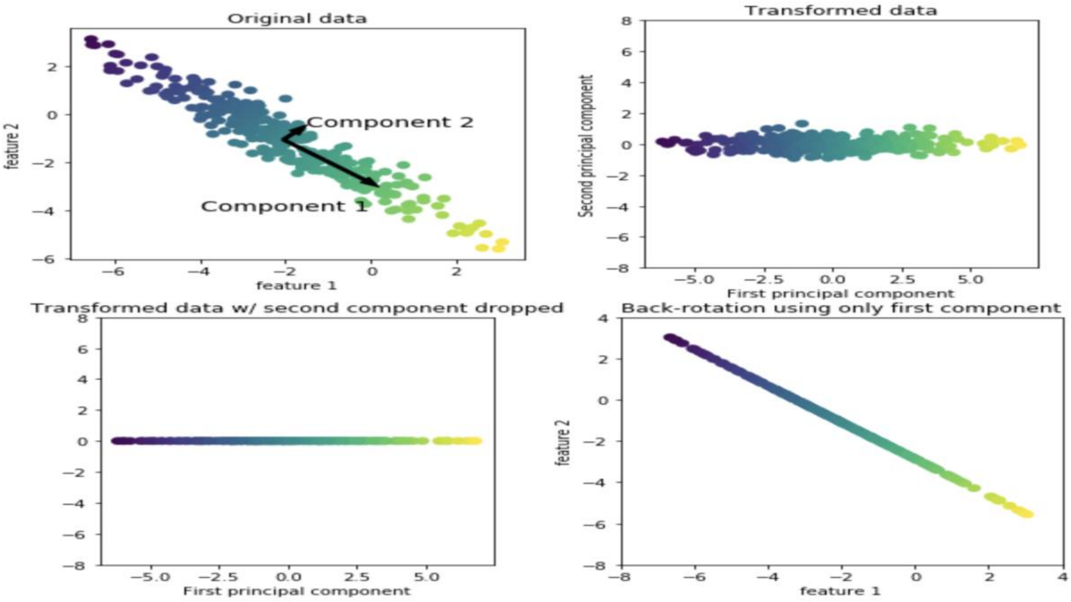
1. **Pixel values of all 400 images in the dataset are scaled from [0,1]**
2. **Names of 40 people are encoded to an integer form 0-39.**



3. **Convert image to a vector**
   Machine learning models can work on vectors. Since the image data is in the matrix form, it must be converted to a vector.

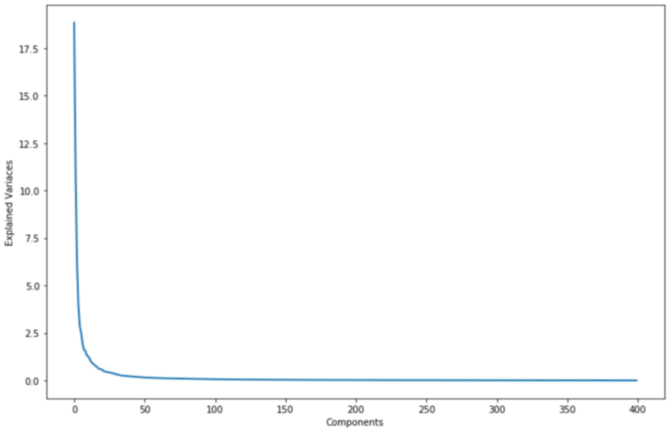
4. **Split target and data into random training and testing set**
   The data set contains 10 face images for each subject. Of the face images, 70 percent will be used for training, 30 percent for testing. Uses stratify feature to have equal number of training and test images for each subject. Thus, there will be 7 training images and 3 test images for each subject.

5. **Principle Component Analysis**
   Principle Component Analysis (PCA) is a method that allows data to be represented in a lesser size. According to this method, the data is transformed to new components and the size of the data is reduced by selecting the most important components



The above illustration shows a simple example on a synthetic two-dimensional data set. The first drawing shows the original data points colored to distinguish points. The algorithm first proceeds by finding the direction of the maximum variance labeled "Component 1". This refers to the direction in which most of the data is associated, or in other words, the properties that are most related to each other.
Then, when the algorithm is orthogonal (at right angle), it finds the direction that contains the most information in the first direction. There are only one possible orientation in two dimensions at a right angle, but there will be many orthogonal directions (infinite) in high dimensional spaces.

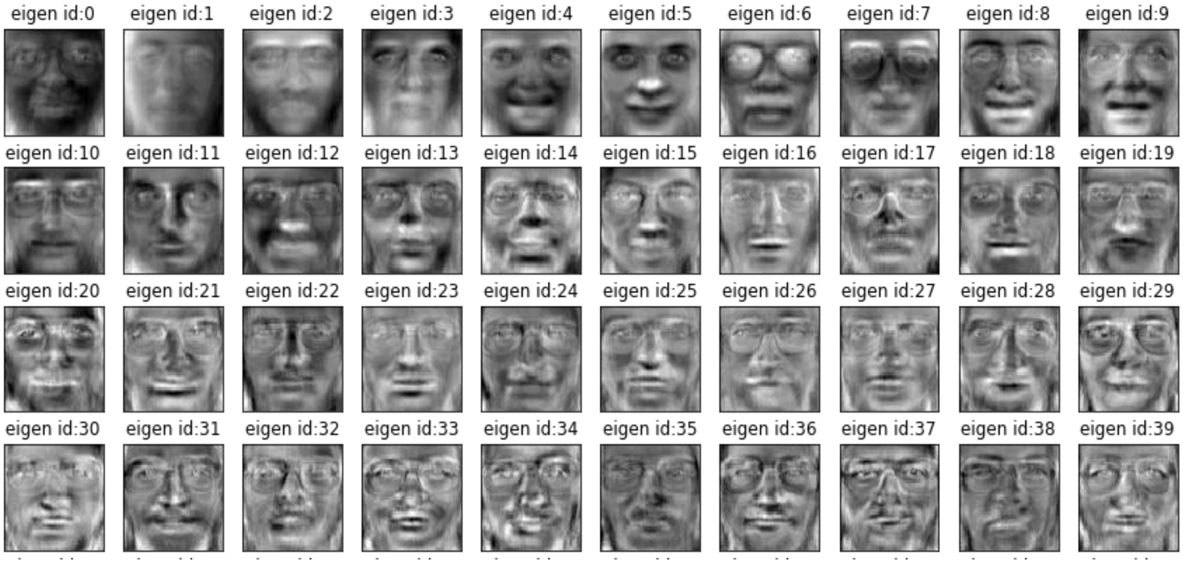
6. Finding Optimum Number of PCA Components



In the figure above, it can be seen that 90 and more PCA components represent the same data. Now let's make the classification process using 90 PCA components.

7. Show Eigen Faces

Eigenfaces refers to an appearance-based approach to face recognition that seeks to capture the variation in a collection of face images and use this information to encode and compare images of individual faces in a holistic (as opposed to a parts-based or feature-based) manner. Specifically, the eigenfaces are the principal components of a distribution of faces, or equivalently, the eigenvectors of the covariance matrix of the set of face images, where an image with N pixels is considered a point (or vector) in N-dimensional space.



8. Training

Training is done using 90 PCA Features. There are 280 images in the training set and 120 images in testing set. Learning rate is selected as 0.001. The architecture of MLP is of 5 layers with 90, 256, 128, 128 and 40 neurons respectively. Number of epochs is taken as 100.

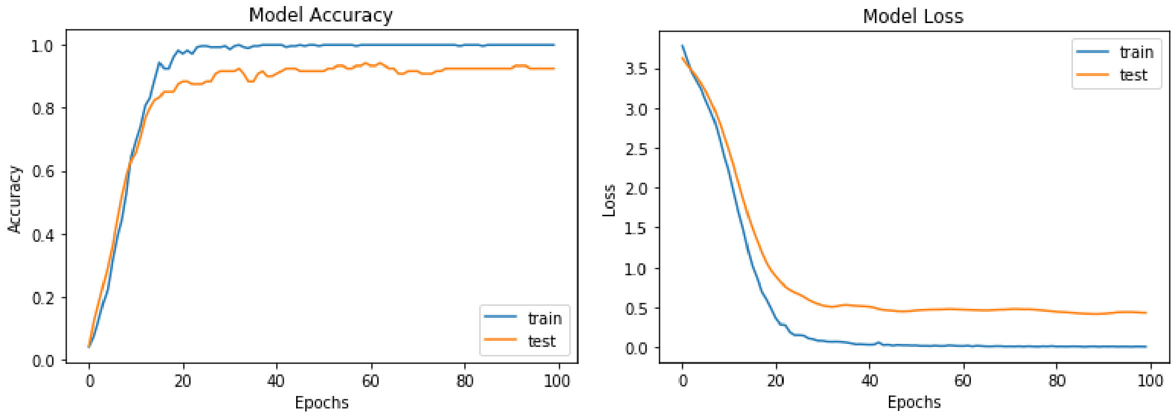
```
_________________________________________________________________
Layer (type)              Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 256)              23296
_________________________________________________________________
dropout_1 (Dropout)          (None, 256)              0
_________________________________________________________________
dense_2 (Dense)              (None, 128)              32896
_________________________________________________________________
dropout_2 (Dropout)          (None, 128)              0
_________________________________________________________________
dense_3 (Dense)              (None, 128)              16512
_________________________________________________________________
dropout_3 (Dropout)          (None, 128)              0
_________________________________________________________________
dense_4 (Dense)              (None, 40)               5160
=================================================================
Total params: 77,864
Trainable params: 77,864
Non-trainable params: 0
_________________________________________________________________
```

Train on 280 samples, validate on 120 samples

```
________________________________________________________________________
Epoch 1/100
280/280 [==============================] - 1s 2ms/step - loss: 3.7728 - acc: 0.0393 - val_loss:
3.6222 - val_acc: 0.0417
Epoch 2/100
280/280 [==============================] - 0s 59us/step - loss: 3.6113 - acc: 0.0714 - val_loss:
3.5406 - val_acc: 0.1167
________________________________________________________________________
________________________________________________________________________
Epoch 99/100
280/280 [==============================] - 0s 54us/step - loss: 0.0076 - acc: 1.0000 - val_loss:
0.4325 - val_acc: 0.9250
Epoch 100/100
280/280 [==============================] - 0s 54us/step - loss: 0.0056 - acc: 1.0000 - val_loss:
0.4307 - val_acc: 0.9250
________________________________________________________________________
```

Result:

Training Accuracy = 1.00 ***Validation Accuracy = 0.9250***



Conclusion: We have implemented an easy to use face recognition system. It can be easily implemented using already available hardware. This system has numerous applications in security as well in identification systems.
It can be further enhanced by introducing sentiment analysis to widen its scope.

References:

1. S.N. Sivanandam and S.N. Deepa *Introduction to Soft Computing*, Wiley India Publications
2. https://www.kaggle.com/imrandude/olivetti#olivetti_faces.npy
3. https://medium.com/@aptrishu/understanding-principle-component-analysis-e32be0253ef0
4. https://keras.io