Face Recognition

Khoula K. Al-Kharusi
Computer Science, Sultan Qaboos University

Abstract

Face recognition is one of the biometrics applications. The face recognition system is a computer application that can recognize or check a person from the digital image or video stream from a video source by comparing the selected facial features.he face recognition model needs specific methods to extract unique features from the face images to deal with the high complexity. Principal Component Analysis (PCA) and Non-negative Matrix Factorization (NMF) algorithms are commonly used to reduce features' dimensionality space. This research aims to build SVM model and train it for face recognition task. The data will be shifted to a new space that is lower in dimensionality than the original space used. Two dimensionality reduction methods, PCA and NMF, are utilized. The subgoal is to compare the two methods and evaluate them.

Keywords: face recognition; classification; SVM; PCA; NMF

1 Introduction

Biometrics is the science of establishing an individual's identity based on the person's physical, chemical, or behavioural attributes (3).One of the biometrics applications is face recognition. The face recognition system is a computer application that can recognize or check a person from the digital image or video stream from a video source by comparing the selected facial features. Face recognition systems essential studies in image analysis and computer vision. Face recognition methods are widely used in security systems at airports and railway stations, finding the faces of a person in criminal situations, identifying employees at enterprises and other applications (4). The recognition complexity is linked to the inherent variety of faces resulting from age, gender, and facial expressions. Image quality and camera characteristics such as resolution, light, signal-to-noise ratio, and background are also critical in recognition. The face recognition model needs specific methods to extract unique features from the face images to deal with the high complexity. Principal Component Analysis (PCA) and Non-negative Matrix Factorization (NMF) algorithms are commonly used to reduce features' dimensionality space.

1.1 Problem Statement

Face recognition is the process of assigning labels or names to face images depending on who appears in that image. This research problem concentrated on the face classification of forty different people using the Support Vector Machine (SVM) model. Two dimensionality reduction methods will be used to reduce the load in the classifier model and fasten the classification process.

1.2 Motivation

Computer vision problems are complex due to the high dimensionality of the images and the number of information that is held by each pixel. Dimensionality reduction methods are developed for such problems. However, how useful they are in practice and what is the effect of abandoning some features in the new space.

1.3 Objectives

This research aims to build SVM model and train it for face recognition task. The data will be shifted to a new space that is lower in dimensionality than the original space used. Two dimensionality reduction methods, PCA and NMF, are utilized. The subgoal is to compare the two methods and evaluate them.

2 Method

2.1 Support Vector Machine (SVM)

SVM is one of the most significant developments in pattern recognition. The simplest form of SVM is a linear discriminant function that separates classes. For generalization, the hyperplane is selected to be as far as possible from any sample. The distance between the datapoints and the hyperplane is called the margin.

$$margin = \frac{2}{||w||} \tag{1}$$

w is the coefficients of the hyperplane. Convert it to a minimization problem to deal with is as a loss function.

$$J(w) = \frac{1}{2}||w||^2 \tag{2}$$

$$constrain \;\; z_i(w^T x_i + w_0) \geq 1 \quad \forall i \tag{3}$$

J(w) is quadratic function with single global minimum. using Kuhn-Tucker theorm (KKT) to convert the problem function to the form of

$$maximize \; L_D(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j z_i z_j x_i^T x_j \tag{4}$$

Where α is a value assign to each sample, $\alpha = 0$ for all samples accept the support vector samples (The closest points to the margin). This is the reason why adding data out of margin will not affect separating line. For soft margin a slack variable is added for each sample. this will change the constrain to be:

$$constrain \;\; z_i(w^T x_i + w_0) \geq 1 - \xi_i \quad \forall i \tag{5}$$

and the cost function:

$$J(w) = \frac{1}{2}||w||^2 + \beta \sum I(\xi > 0) \tag{6}$$

$$I(\xi > 0) = 1 \;\; if \;\; \xi > 0 \;\; I(\xi > 0) = 0 \;\; otherwise \tag{7}$$

SVM avoid curse of dimensionality from lilting data into a higher dimensional space by enforcing largest margin and the computation in the higher dimensional case is performed only implicitly through the use of kernel functions. Kernel is a similarity measure (modified dot product) between data points.

$$maximize \; L_D(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j z_i z_j K(x_i, x_j) \tag{8}$$

The aim is to find Kernel function that $\mathrm{k(x_i,\, x_j)} = \varphi(\mathrm{x_i})^T \; \varphi(\mathrm{x_j})$

commonly used kernels:

Linear

$$k(x_i, x_j) = x_i.x_j \tag{9}$$

Polynomial of power p

$$k(x_i, x\_j) = (1 + x_i.x_j)^P \tag{10}$$

Radial-basis function (rbf)

$$k(x_i, x_j) = exp[-\gamma.\; ||x_i - x_j||] \tag{11}$$

Sigmoid

$$k(x_i, x_j) = tanh(\beta_0 \; x_i \; x_j \; + \beta_1) \tag{12}$$

2.2 Dimensionality Reduction

Due to the high dimensionality of data, similarity and distance metrics are computationally expensive and some compaction of the original data is needed.

2.2.1 Principal Component Analysis (PCA)

The PCA is a well-known method used to approximate datasets with lower-dimensional feature vectors. In the case of face recognition, the data considered a grayscale image converted into a vector. Let $X = \{X_n \in R^d | n = 1, ..., N\}$be an set of image vectors. Let EX be the mean vector in the image set (5).

$$EX = \frac{1}{N}\sum_{n=1}^{N} X_n \tag{13}$$

After subtracting it from each vector of X, we get.

$$\hat{X} = \{\hat{X_n}, n = 1, ..., N\} \;\; with \;\; \hat{X_n} = X_n - EX \tag{14}$$

Then we define the covariance matrix M for X as

$$\Sigma = cov(\hat{X}) = E(\hat{X} \otimes \hat{X}) \tag{15}$$

where Σ is an N2xN2 matrix. PCA attempts to find a linear mapping M that maximizes the cost function trace $M^T \Sigma M$. It can be shown that this linear mapping is formed by the d principal eigenvectors of the sample covariance matrix of the zero-mean data($\hat{X}$). Hence, PCA solves the eigenproble

$$\Sigma M = \lambda M \tag{16}$$

PCA maximizes $M^T \Sigma M$ with respect to M, under the constraint that the L2-norm of each column m_j of M is 1, i.e.,that $||m_j||^2 = 1$. This constraint can be enforced by introducing a Lagrange multiplier λ. Hence, an unconstrained maximization of $m_j^T \Sigma m_j + \lambda(1 - m_j^T m_j)$ is performed. The stationary points of this quantity are to be found when $\Sigma m_j = \lambda m_j$.

2.2.2 Non-negative Matrix Factorization (NMF)

The dataset is represented as a nm matrix X. Each column is a non-negative vector of dimension n corresponding to a face image, and m is the number of images. Then we can find two new non-negative matrices (W and H) to approximate the original matrix X.

$$X_{ij} \simeq (WH)_{ij} = \sum_{a=1}^{r} W_{ia} H_{aj}, W \in R^{n \times r}, H \in R^{r \times m} \tag{17}$$

Where r is the number of base vectors usually chosen as small as possible for dimension reduction, each column of matrix W represents a basis vector, in contrast, each column of H means the weights used to approximate the corresponding column in X using the bases from W. For the NMF method, in contrast to PCA, no subtractions can occur, so the non-negativity constraints are compatible with the intuitive idea of combining parts to form a whole face, which is how NMF learns a parts-based representation (6). NMF involves approximately solving the optimization problem of minimizing the difference between X and WH:

$$\min_{W,H} ||X - (WH)||_F^2, \;\; such\ that\ W \geq 0 \;,\; H \geq 0 \tag{18}$$

Algorithm 1 shows a a general framework for solving NMF. Which starts with the given non-negative matrix X and the initial matrices, W^0 and H^0. Then it tries to find two non-negative matrices, W and H, such that the value of $||X - (WH)||$ is minimized. Based on the non-convexity property for NMF, it generally does not guarantee a unique solution. Its solution depends on choosing initialization for W and H demonstrated as W^0 and H^0. A good choice for initializing can significantly affect the algorithm's convergence rate and considerably reduce

```
Algorithm 1 The generic NMF Algorithm
Input: $X, W^0, H^0, \varepsilon$, and the rank of approximation r
Output: $W and H$
Set: Objective function: $F = ||X - (WH)||$

  Checking Stop Condition(s)
  if $||x^r - x^{r+1}|| < \varepsilon$ or finish iterations, then
      finish algorithm
  else
      Trial step calculation: Update $W$ and $H$
  end if
```

the value of the cost function. Based on the non-convexity property for NMF, it generally does not guarantee a unique solution. Its solution depends on choosing initialization for W and H demonstrated as W0 and H0 (1). A good choice for initializing can significantly affect the algorithm's convergence rate and considerably reduce the value of the cost function. The existing initialization approaches for NMF can be classified into four categories, as shown in Fig 1.

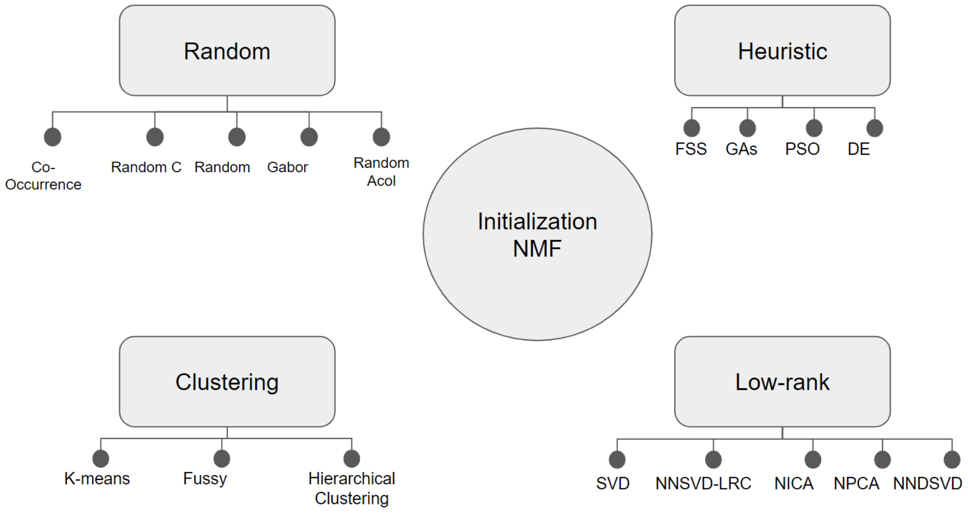

Figure 1: Proposed scheme for initialization NMF

Random schemes which only use the random strategy

- Random, which suggests initial matrix by using random
- Random Acol, which calculates initial matrix W by getting an average of q random columns of the matrix X
- Random C, which calculates initial matrix W by getting an average of the chooses q columns randomly from the longest (in the l2-norm) columns of X
- Co-Occurrence, which computes matrix W by using $X^T X$
- Gabor-based, which calculates the matrix W by using Gabor wavelet and it is suitable for image datasets

Clustering schemes which profit the clustering strategy

- K-means, which use the K-means algorithm for initialization matrix W
- Fuzzy C-means, which works based on the fuzzy roles
- Hierarchical Clustering, which groups similar objects into groups called clusters

Heuristic schemes which are based on Population-Based Algorithms (PBAs)

- Genetic Algorithm
- Particle Swarm Optimization
- Differential Evolution
- Fish School Search

Low-rank Approximation-Based schemes which works based on decreasing the matrix rank

- Singular Value Decomposition, which works based on SVD decomposition
- Nonnegative Singular Value Decomposition with Low-Rank Correction which generates a positive matrix
- Non-negative PCA, which works based on PCA algorithm
- Non-negative ICA, which works based on ICA algorithm

3 Experiments

3.1 Dataset

The Olivetti Research Laboratory (ORL) Database of Faces was collected by ATT Laboratories. This set of face images was taken between April 1992 and April 1994 at the lab. There are ten different images of each of 40 distinct subjects shown in Fig 2. The images were taken at different times for some subjects, varying the lighting, facial expressions (open/closed eyes, smiling / not smiling) and facial details (glasses / no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement). The dataset is loaded from sklearn library `sklearn.datasets.fetch_olivetti_faces` can be described in the Table 1:

Classes	40
Samples	400
Samples per Class	10
Dimensionality	$64 \times 64 = 4096$
Data type	float between 0 and 1

Table 1: Row dataset description

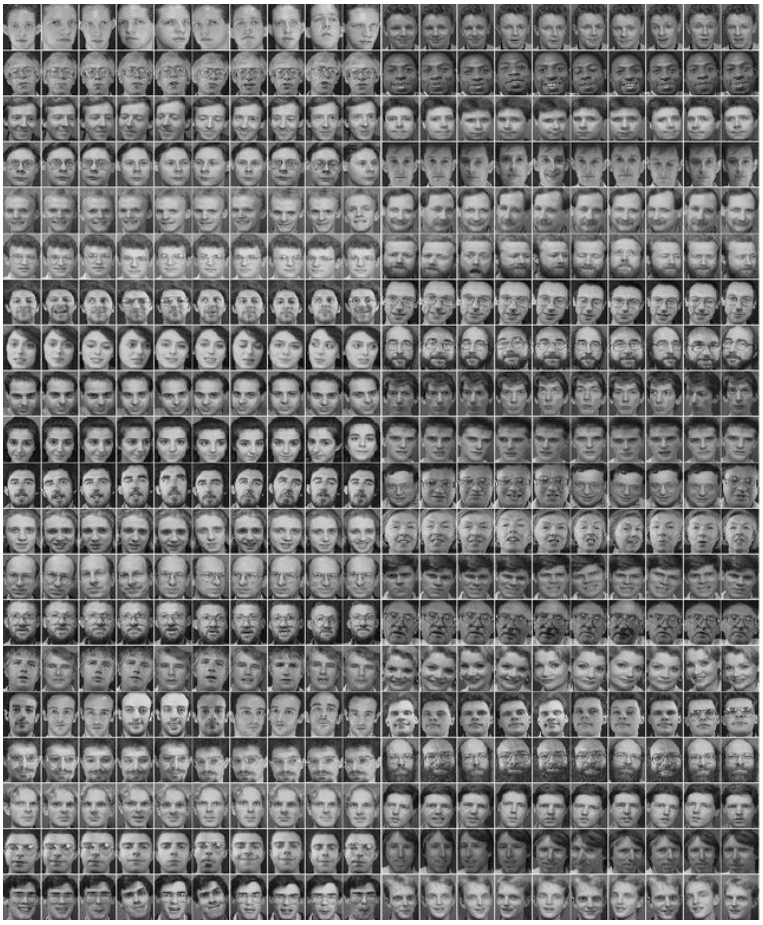

Figure 2: ORL Face Database

Since the data is already scaled to $X \in [0, 1]$ then there is no need to rescale the data. The images are reshaped to be a vector instead of 2-D matrix.

3.2 Experimental Setup

Python 3 has been used as a programming language for this project to process data and implement classification models because of its helpful libraries. Before implementing the models, the dataset was split to train and test sets. Afterwards, grid-search cross-validation was used to find the optimal SVM model's parameters. Then the data was shifted to a new lower dimensional space by PCA and NMF dimensionality reduction methods. The grid search is used for each new data generated by PCA and NMF to optimize SVM classifier for each data space (original, PCA and NMF).

3.2.1 Training and Testing Sets

The dataset was split into a train set and a test set. The training set contains 70% of the whole dataset, and the testing set is the remaining 30% of the dataset. To ensure we have an equally likely distribution of the test-train portions for each class 3 out of 10 class samples are used for test.

3.2.2 Dimensionality Reduction Experiments

Dimension reduction, is the transformation of data from a high-dimensional space into a low-dimensional space so that the low-dimensional representation retains some meaningful properties of the original data. Let d be the original high-dimensional space ans r the low-dimensional space. Since the aim is to reduce r then the smallest r is the better.Therefore, in this experiment we will test various values of r for PCA and NMF and report the affect of the change of r to the classification accuracy.

3.2.3 Models Tuning

Model tuning is the process of finding the optimal hyperparameters for the model. Grid-search cross-validation has been used to tune SVM classification model. Grid-search is a strategy used to search different values for model hyperparameters and choose a subset that results in a model that achieves the best performance on a given dataset. The researcher sets the values of the hyperparameters in advance for the algorithm. Then, the grid-search algorithm combines all possible sets of each hyperparameter values and evaluates the model using cross-validation. In Table 2 the hyper parameter used for grid-search to tune the classification model.

Model	Hyperparameter	Values
Support Vector Machines	C	0.1, 1, 10, 100, 1000
	gamma	scale, 1, 0.1, 0.01, 0.001, 0.0001
	kernel	linear, rbf, poly, sigmoid

Table 2: Hyperparameter values for SVM model tuning

Where the parameters are:

1. **C** is the regularization parameter. The strength of the regularization is inversely proportional to C. Must be strictly positive. The penalty is a squared l2 penalty.
2. **gammea** Kernel coefficient for ‘rbf’, ‘poly’ and ‘sigmoid’.
3. **kernel** Specifies the kernel type to be used in the algorithm.

3.2.4 Classification Performance Evaluation Metrics

There are a number of evaluation metrics for classification. The confusion matrix Fig 3 is one way of summarizing the classification results of a model. The rows represent the predicted label while the columns represent the actual label. It has 4 entries: TP and TN donate the number of positive and negative correctly classified instances while FP and FN donate the number of positive and negative miss-classified instances (2).

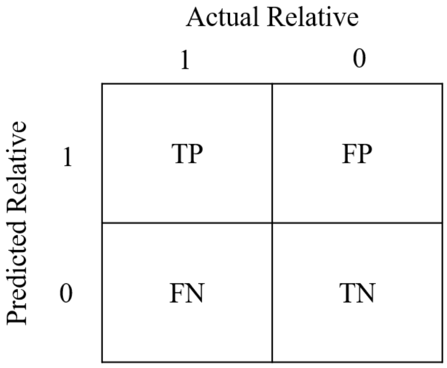

Figure 3: Confusion Matrix

Accuracy measures the ratio of correct predictions over the total number of instances evaluated.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{19}$$

Other metrics are preferred for in-balanced datasets, we list them in the following sections:

Precision measures the positive instances that are correctly predicted from the total positive predictions.

$$Precision = \frac{TP}{TP + FP} \tag{20}$$

Recall measures the fraction of positive data points that are correctly classified.

$$Recall = \frac{TP}{TP + TN} \tag{21}$$

F1 Score is the harmonic mean of both Precision and Recall.

$$F1Score = 2 * \frac{precision * recall}{precision + recall} \tag{22}$$

Since our classification model categories face images to multi-classes we are going to calculate the measurements for class separately.

4 Results and Discussion

4.1 Models Tuning Results

The models' hyperparameters values were sat depending on the grid search results. The first model used the original datast and the parameters are:

Hyperparameter	Values
C	0.1
gamma	scale
kernel	linear

Table 3: Original data model's parameter

The inverse regularization parameter is 0.1 which means that the SVM model is generalized. The kernel is linear and it is less complected to compute compare to the other kernels. The grid search is used to SVM models with PCA and NMF transformed data to $r = 30$.

Hyperparameter	Values
C	10
gamma	0.01
kernel	rbf

Table 4: PCA model's parameter

Hyperparameter	Values
C	10
gamma	scale
kernel	linear

Table 5: NMF model's parameter

4.2 Dimensionality Analysis

After tuning the parameter for each model, the dimensionality experiments will use the SVM models described in Sec 4.1.

4.2.1 PCA

Using PCA method the features transformed to new space that the values may be negative and it may be hard to interpret the features. However, we can visualise the components and the captured variability by them. In Fig 4 the first three component of PCA is shown. The components are a weighted combination of the original features.

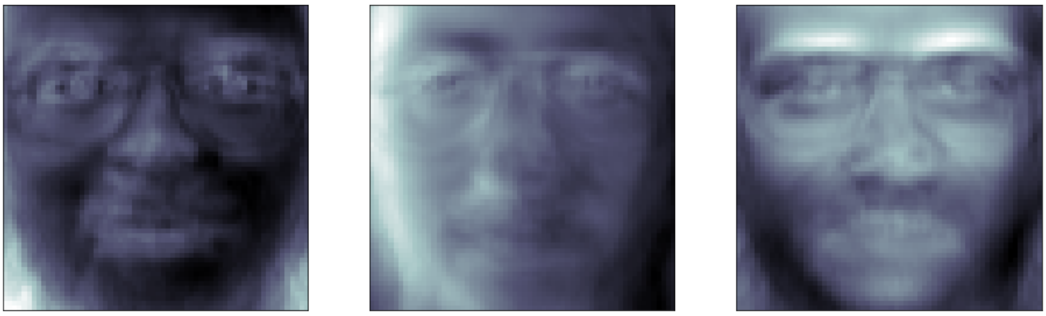

Figure 4: First three components of PCA

To find the variability captured by PCA we plot the eigenvalues and the cumulative sum in Fig5. The Bar chart is used to represent individual explained variances, and Step plot is used to represent the variance explained by different principal components. From the plot 90% of the original data variance is explained by the first 59 components.

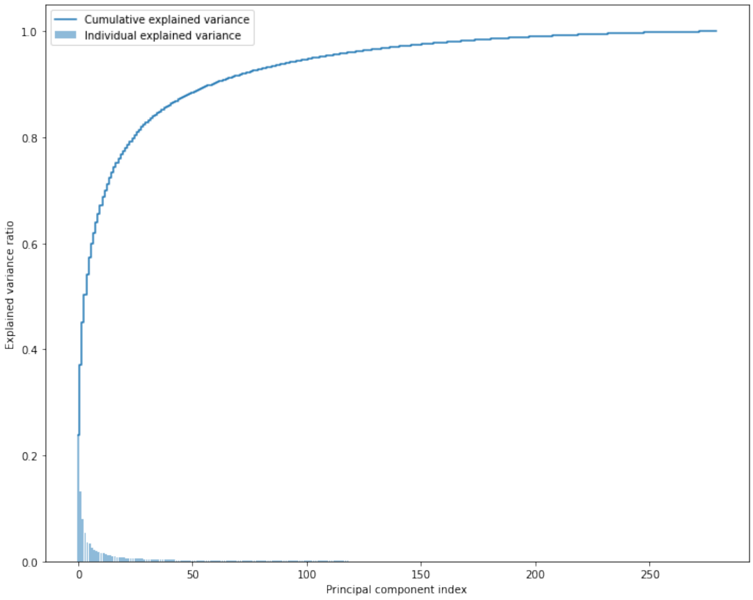

Figure 5: Explained variance ratio of PCA

For face recognition task, SVM classifier with table 4 has been build with distinct dimensions. The Fig6 plots the accuracy, precision, recall and F1 score of the face recognition task and the effect of number of PCA components in the scores.

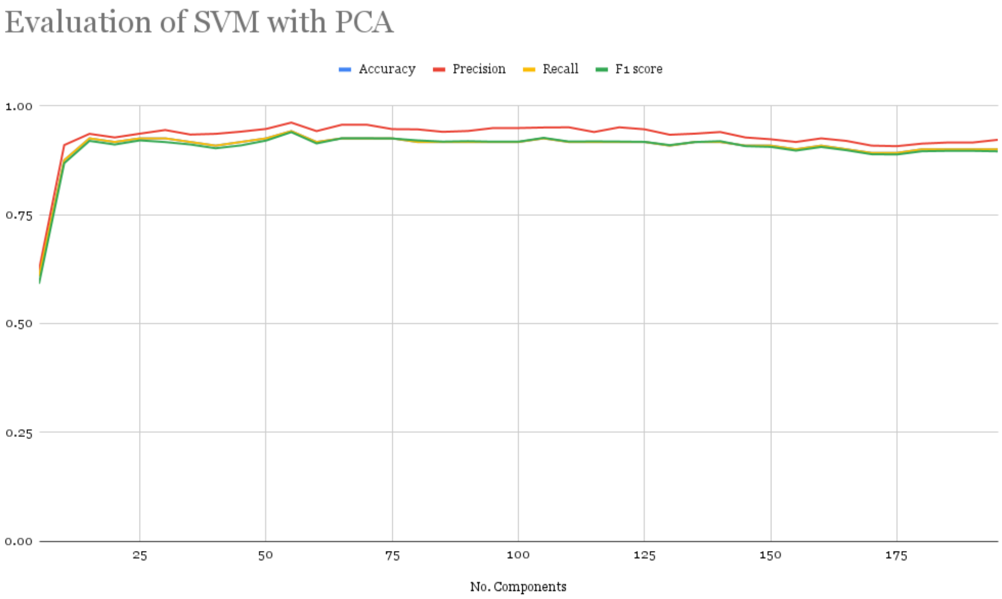

Figure 6: PCA dimensions and face recognition results

The accuracy score reached to 90% with only twenty features with PCA. After reaching this point there was no notable increase in the accuracy until the number of features reach to 150, from this point the accuracy slightly decrease.

4.2.2 NMF

Similar to the PCA analysis, NMF method create new feature space. Visualising the first three components we get the following images.

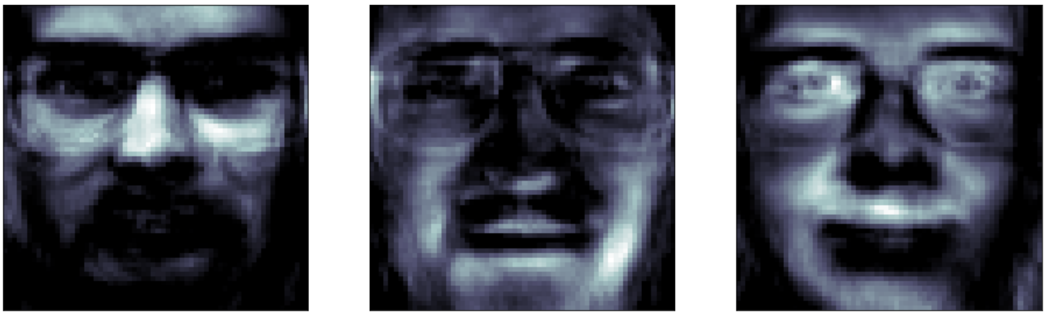

Figure 7: First three components of NMF

In the task of face recognition with data transformed by NMF with the SVM model in table 5 the classification measures plot appears in Fig 8. The NMF initialization is random and this explains the fluctuate of the classification measures.

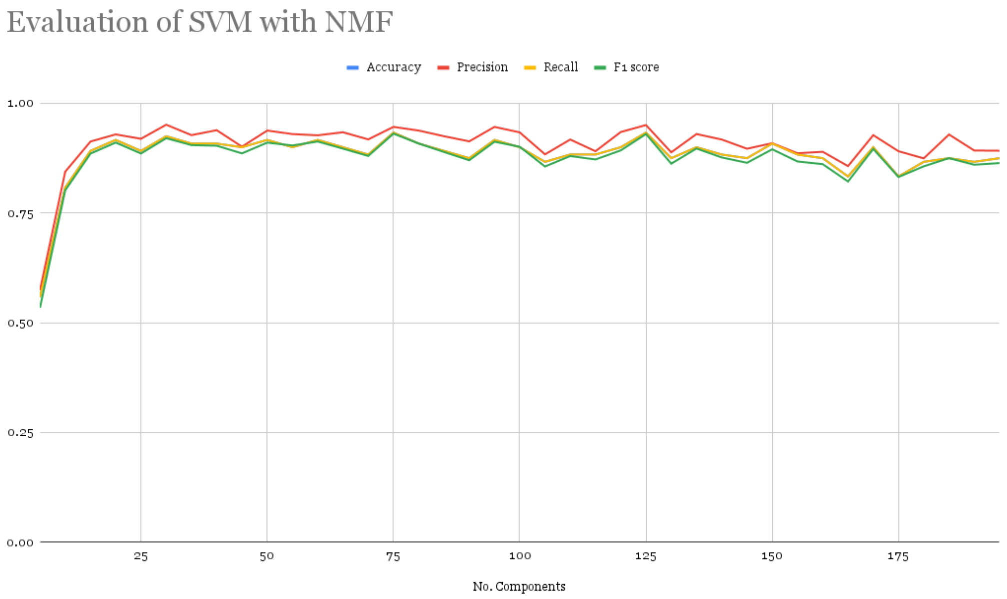

Figure 8: NMF dimensions and face recognition results

As the NMF can be initialised with many algorithms, in Fig 9 is a comparison between the accuracy of face recognition model with NMF data transformed by initialising NMF with random and with Nonnegative Double Singular Value Decomposition (NNDSVD).

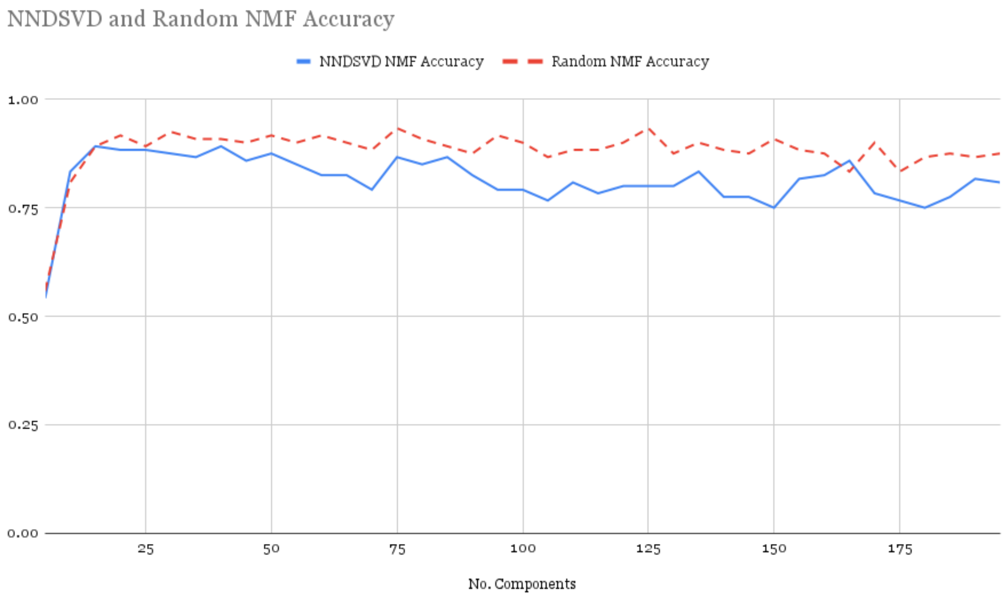

Figure 9: NMF initialisation strategies and face recognition accuracy

In this dataset the random initialisation shows superior results compared to NNDSVD initialisation. The NMF in most cases did not converge within the maximum number of iterations when NNDSVD used to initialise W^0 and H^0.

4.3 Classification Results

To analyse the classification of the two decomposition methods the dimension of the transformed data is 30. The following analysis based on random NMF.

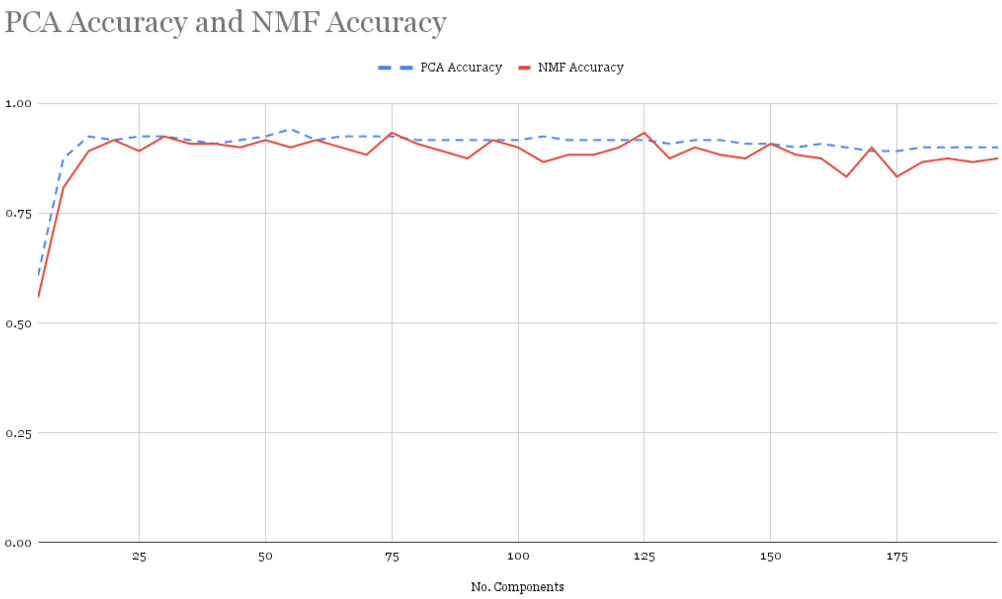

Figure 10: PCA Accuracy and NMF Accuracy

The two methods have similar accuracy recognition, however, PCA is more stable than NMF. Using the original dataset the classification accuracy is 96.7%, the classification accuracy of PCA with 30 features data is 92.5% and NMF data accuracy is 92%.

4.3.1 Classification Samples Analysis

In Fig 11 shows samples that had been classified correctly with the three models.

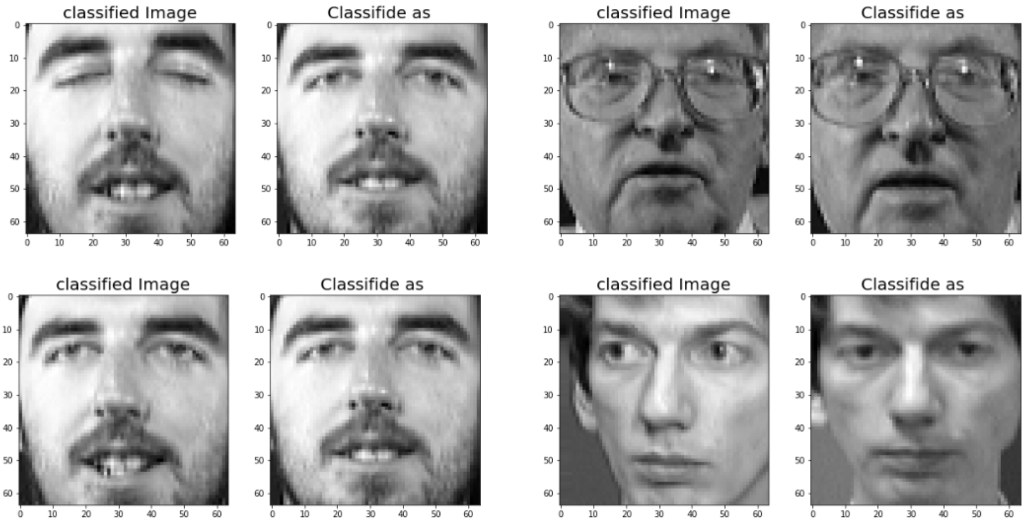

Figure 11: Faces accurately recognized by all datasets

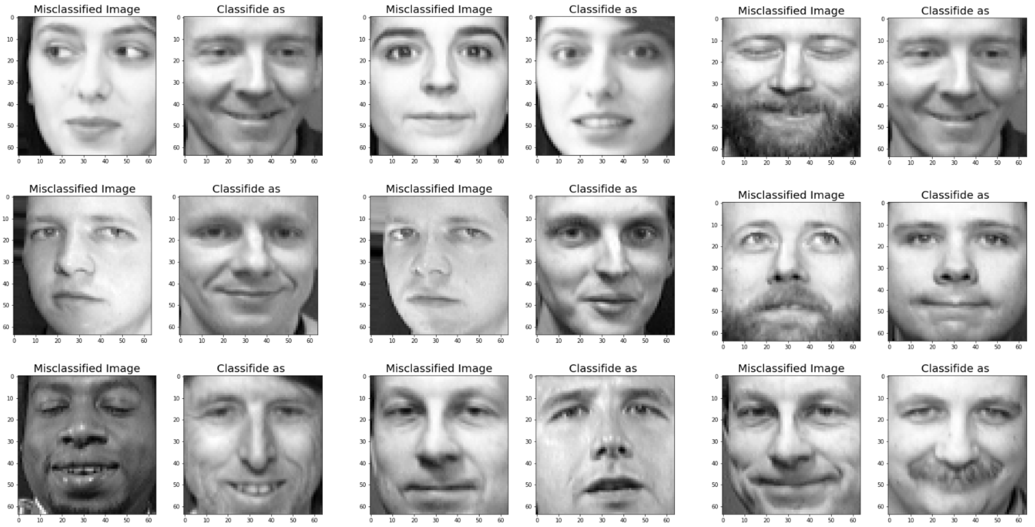

Figure 12: Misclassified images by PCA model

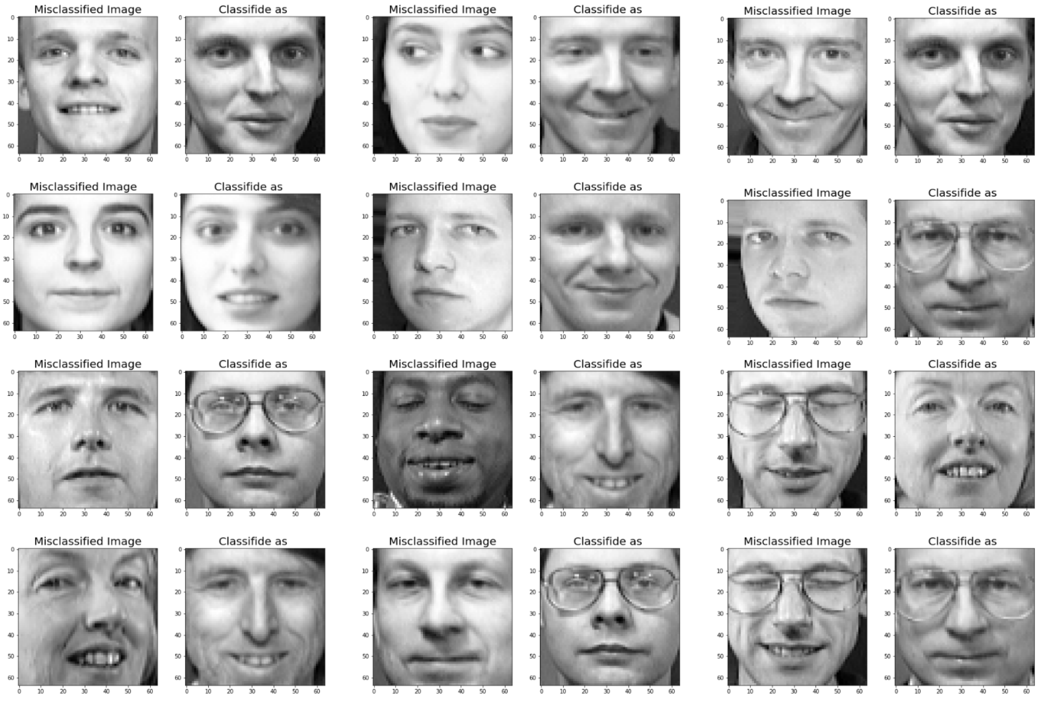

Figure 13: Misclassified images by NMF model

The overlap between face images in Fig 12 and 13 is a sign that theier representation in the dimensional space of NMF and PCA are related.

5 Conclusion

This project aimed to build classification models that accurately recognize faces images and compare their performance. The dataset is transformed using dimensionality reduction methods that are PCA, and NMF. The performance of the classification model with the original dataset is 96.7%, the classification accuracy of PCA with 30 features data is 92.5%, and NMF data accuracy is 92%. The dimension has been reduced by 99.3%, while the model accuracy was reduced by 4.7% only. The model can be improved by using another classifier like a neural network.

References

S. F. Hafshejani and Z. Moaberfard. Initialization for nonnegative matrix factorization: a comprehensive review, 2021.

M. Hossin and M. Sulaiman. A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining & Knowledge Management Process*, 5(2):1, 2015.

A. K. Jain, P. Flynn, and A. A. Ross. *Handbook of Biometrics.* Springer, Boston, MA, 1 edition, 2008.

W. Y. Min, E. Romanova, Y. Lisovec, and A. M. San. Application of statistical data processing for solving the problem of face recognition by using principal components analysis method. In *2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, pages 2208–2212, 2019.

L. van der Maaten, E. O. Postma, and J. van den Herik. Dimensionality reduction: A comparative review. 2009.

Y. Xue, C. S. Tong, W.-S. Chen, W. Zhang, and Z. He. A modified non-negative matrix factorization algorithm for face recognition. In *18th International Conference on Pattern Recognition (ICPR'06)*, volume 3, pages 495–498, 2006.