NATIONAL ECONOMICS UNIVERSITY

Data Mining & Advanced Statistical Methods in Business Analysis

FACE RECOGNITION

DSEB 63 – Group 5

Supervisor: Pham Tuan Minh, PhD

List of members

Nguyen Cam Ly	**11213588**
Tran Phuong Anh	**11219258**
Tran Hai Nam	**11219279**
Le Thi Quynh Anh	**11219256**
Tran Thi Ngan Ha	**11211959**

Ha Noi, 2024

Table of contents

1 Introduction 3

1.1 Problem definition 3

1.2 Motivation 3

1.3 Problem setting 4

1.3.1 Datasets 4

1.3.2 Goal 5

1.4 Demo 5

2 Literature review 6

3 Solution 7

3.1 Dimensionality reduction techniques 7

3.1.1 PCA 7

3.1.2 Kernel PCA 7

3.1.3 Incremental PCA 7

3.1.4 Sparse PCA 7

3.1.5 Randomized PCA 8

3.1.6 Modular PCA 8

3.2 ***Task 1:*** *Face recognition using distance measures* 8

3.2.1 Distance measures 8

3.2.2 Experiments and results 11

3.3 ***Task 2:*** *Face recognition using machine learning classifiers* 13

3.3.1 Machine learning classifiers 13

3.3.2 Experiments and results 13

3.4 ***Task 3:*** *Improve system by using Modular PCA and image enhancement techniques* 14

3.4.1 Image enhancement techniques 14

3.4.2 Experiments and results 17

4 Results 20

4.1 Metrics 20

4.2 Results 20

5 Conclusion 21

5.1 Research overview 21

5.2 Future works 21

References 22

1 Introduction

1.1 Problem definition

The human face is considered the prime factor in recognizing a person's identity in our society. Humans can quickly identify individuals through their visuals. The face recognition system was built based on the fact that each person's face is unique.

This technology serves two main purposes: *verification* and *identification*. In verification, a face image is matched with a claimed identity to determine its accuracy. For example, if someone says they're John Smith, the system checks if their face matches John Smith's. In identification, a face image is used to determine the person's identity by comparing face with the database of images of known individuals.

Face recognition finds applications in various fields such as security, surveillance, investigation, identity verification, and more. Its versatility makes it invaluable in enhancing safety, efficiency, and accuracy across different sectors.

1.2 Motivation

In sectors such as retail and hospitality, we can use face recognition technology at entry points or check-in counters to instantly identify the customer. By identifying, we can automatically extract meaningful information such as how frequently the customer has used our service, their past purchase history, and demographic information. This enables staff to provide personalized recommendations, enhancing their shopping experience and increasing the likelihood of purchasing. Whether it's suggesting personalized menu items in a restaurant or showcasing products that align with a customer's past preferences in a retail setting, face recognition optimizes the upselling process, increasing the chances of additional purchases.

Beyond customer interactions, face recognition in retail and hospitality contribute to operational efficiency. It assists in managing queues, tracking foot traffic, and optimizing staff allocation based on peak hours. This not only improves the overall flow of operations but also enables businesses to allocate resources more effectively, enhancing productivity.

In conclusion, face recognition technology in retail and hospitality goes beyond just identifying customers; it becomes a catalyst for elevating the entire customer journey. From personalized experiences to operational efficiency, its integration holds the promise of revolutionizing how businesses engage with and cater to their clientele, ultimately leading to increased customer satisfaction and loyalty.

1.3 Problem setting

1.3.1 Datasets

Task 1, 2: AT&T dataset

The dataset consists of 400 images featuring 40 unique individuals. The images were captured against a consistent dark background, with subjects standing upright and facing forward, allowing for slight variations in head position. Lighting, facial expressions (eyes open/closed, smiling/not smiling), and the presence or absence of eyeglasses were deliberately varied during image capture. Each image has a resolution of 92 x 112 pixels.

Task 3: LFW (Labeled Faces in the Wild) dataset

This dataset focuses on the unconstrained face recognition problem, which explores recognizing faces despite variations in factors like posture, facial expression, hairstyles, and focus. The dataset is a preprocessed subset of the "Labeled Faces in the Wild" dataset available through scikit-learn. It includes 1,288 images from 7 individuals, each with at least 70 images included. This selection ensures a good representation of each individual despite potential variations in their appearance.

Figure 1. *Sample images from AT&T dataset*

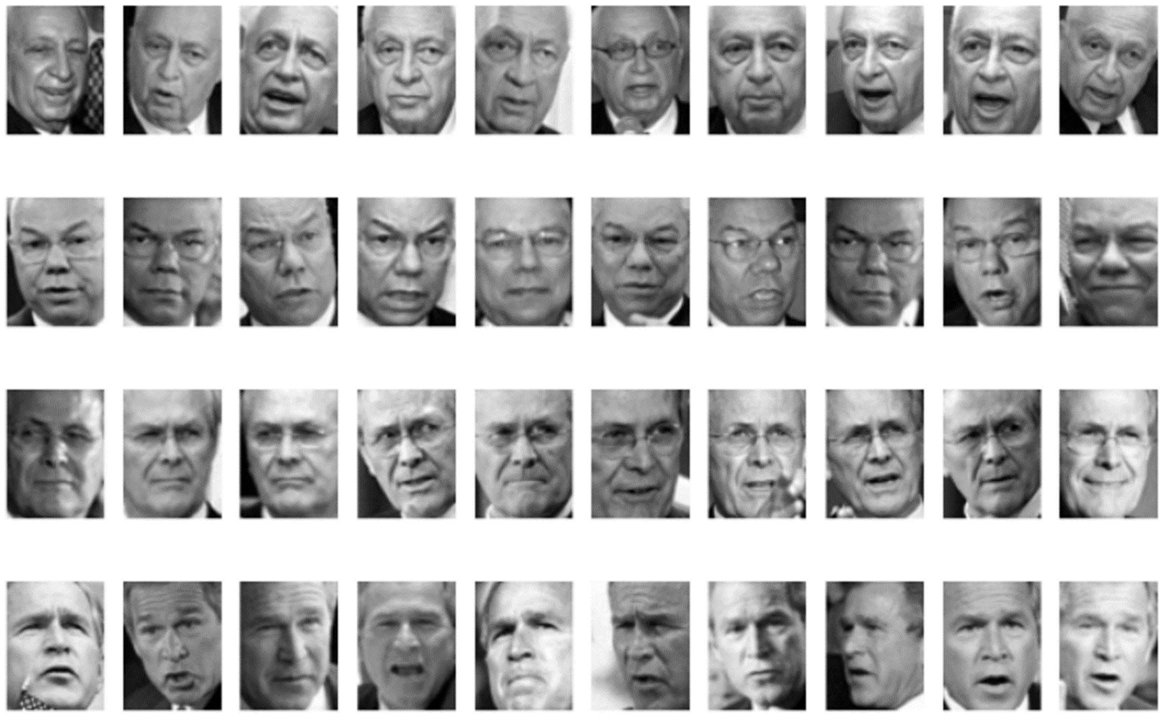

Figure 2. *Sample images from LFW dataset*

1.3.2 Goal

Our goal is to develop a robust, reliable, and understandable system that can effectively and efficiently recognize and verify individuals based on their facial features under varying conditions. By improving our face recognition technology, we aim to enhance the accuracy and f1-score of the model. We aspire to provide businesses with a state-of-the-art tool that consistently delivers reliable results, fostering trust and confidence in the capabilities of facial recognition technology.

1.4 Demo

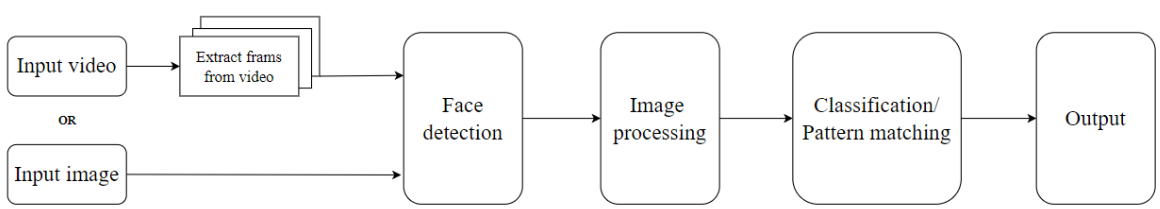

Figure 3. *Face recognition pipeline*

- *Face detection:* is the process of locating human faces in a particular image. This step aims to identify whether the image has human faces or not. Challenges such as differences in lighting and facial expressions can hinder accurate face detection.

- *Image processing:* is to ensure consistent and clear data, while feature enhancement techniques amplify the details crucial for reliable identification.
- *Classification/ Pattern matching*: is to identify the individual in the detected face(s) by comparing facial features to known faces.
- *Output:* provides the identified name of the person in the image or the identity that best matches the detected face.

Figure 4. *Real-time face recognition*

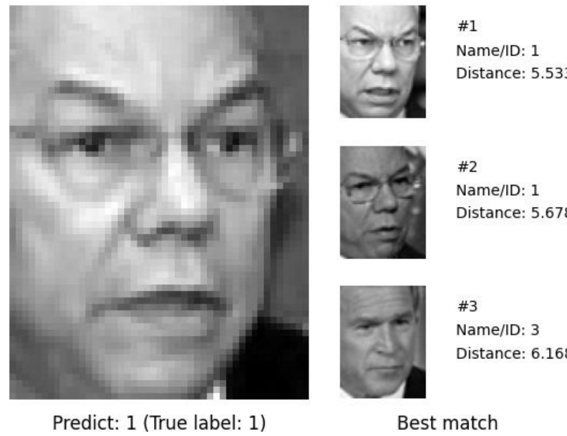

Figure 5. *Prediction & Best match*

Within the face recognition system, the algorithm searches for the identity in the input image or frames of a (real-time) video within the database. This process utilizes machine learning classifiers and can generate a list of the most accurate matches based on distance measures.

2 Literature review

The field of face recognition research has been experiencing rapid advancements. According to **M.K.Rusia (2022)**, the process of recognizing a person's face can be broken down into two steps: first, detecting the location of the face, and second, classifying the detected region.

Over the last decade, numerous approaches have been introduced for face recognition, which can be categorized into four types: appearance-based, feature-based, model-based, and hybrid approaches. Today, deep learning technology is becoming increasingly popular due to its ability to recognize patterns and handle variations.

Several approaches have been experimented with ORL database and achieved a significant result. The eigenfaces method introduced by **M. Turk** and **A. Pentland (1991)** produced 89.5% recognition rate. **Lawrence et al (1997)** also reported a 96.2% recognition rate using hybrid neural network solution. LFW dataset is also a widely used benchmark for face recognition algorithms, **D. Yi et al (2014)** achieve accuracy of 96.33% after tuning with PCA. **Taigman et al. (2014)** presented DeepFace, a deep learning model that achieved an impressive accuracy of 97.35% on the LFW dataset.

Chellapa et al (1995) presented a survey on several statistics, machine learning and deep learning methods for face recognition. Currently, one of the methods that yields promising results on frontal face recognition is the principal component analysis (PCA). Some advantages of PCA are reducing the number of variables in a dataset while retaining the most important information; improving the interpretability of the data, making it easier to understand; simplifying high-dimensional data visualization, and revealing patterns and relationships. Therefore, we decided to use PCA with some kinds of distances and classification methods to build a face recognition system.

The main objective of this research is to improve the accuracy and precision of face recognition under varying facial expression, illumination, and pose conditions in ORL and LFW dataset by modular PCA method with advanced image processing techniques such as Contrast Adjustment, Bilateral filter, and Gaussian Blur.

3 Solution

3.1 Dimensionality reduction techniques

3.1.1 PCA

PCA, or *Principal Component Analysis*, is a widely used technique in data analysis and machine learning. It tackles the challenge of reducing the dimensionality of data while aiming to preserve the most important information.

Imagine a dataset with many features, potentially leading to complexities in visualization and analysis. PCA helps by identifying new, uncorrelated variables called principal components (PCs). These PCs capture the most significant variation in the original data, effectively compressing information into a lower-dimensional space.

3.1.2 Kernel PCA

Kernel PCA addresses the limitation of PCA in handling non-linear relationships in data. It transforms the data into a higher-dimensional space using a kernel function, essentially making it linearly separable. This allows standard PCA to be applied effectively, enabling the extraction of meaningful components even when the original data exhibits non-linear patterns.

3.1.3 Incremental PCA

Incremental PCA tackles the challenge of dealing with large datasets that cannot be loaded into memory at once. It updates the principal components incrementally as new data arrives, requiring significantly fewer resources compared to processing the entire dataset up front. This makes IPCA ideal for real-time applications and situations where processing massive datasets in chunks is necessary.

3.1.4 Sparse PCA

Sparse PCA goes beyond simply reducing dimensions. It introduces a sparsity constraint during the process, aiming for a lower-dimensional representation with fewer non-zero components. This allows SPCA to identify the most influential features in the data and promotes interpretability, especially valuable when dealing with high-dimensional data containing many irrelevant or redundant features.

3.1.5 Randomized PCA

Randomized PCA prioritizes computational efficiency over exactness. It utilizes random projections to approximate the covariance matrix, significantly reducing the computational complexity compared to standard PCA. While sacrificing some accuracy for efficiency, RPCA is a valuable option for situations where precise principal components are not crucial but obtaining results quickly is essential.

3.1.6 Modular PCA

In this research, we propose the modular PCA method, which extends the conventional PCA method and draws inspiration from the modular PCA method introduced by Rajkiran Gottumukkal and Vijayan K.Asari (2003).

The modular PCA method is a technique used to divide face images into smaller, equally sized images represented as a NumPy matrix. The number of sub-images N should be a square number. Figure 1 shows an image divided into four regions (N = 4). The matrix of each image is reshaped as an array and concatenated horizontally to create a new sample and then we apply the conventional PCA to the new samples to reduce dimension.

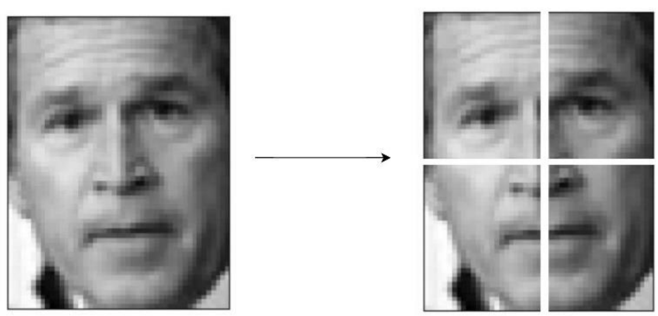

Figure 6. *An image divided into 4 smaller, equally sized sub-images.*

By segmenting the facial image into smaller, localized regions, modular PCA can capture specific characteristics that exhibit relative invariance to variations in pose and illumination. This approach also possesses the potential to mitigate the impact of certain types of image noise.

3.2 ***Task 1:*** *Face recognition using distance measures*

3.2.1 Distance measures

Distance measures play a crucial role in face recognition algorithms by quantifying the similarity or dissimilarity between facial feature representations. These measures help determine how closely a pair of faces match, and they are often used in the context of comparing feature vectors extracted from facial images. In this task, we tried using 16 different distance measures, combined with all of dimensionality reduction techniques mentioned above. Here are 16 distance measures that we used:

(1) Minskowski distance (Lp metrics), here p>0

$$d(\boldsymbol{X},\boldsymbol{Y}) = L_p(\boldsymbol{X},\boldsymbol{Y}) = \left(\sum_{i=1}^{n} |x_i - y_i|^p\right)^{1/p}$$

(2) Manhattan distance (L1 metrics, city block distance)

$$d(\boldsymbol{X},\boldsymbol{Y}) = L_{p=1}(\boldsymbol{X},\boldsymbol{Y}) = \sum_{i=1}^{n} |x_i - y_i|$$

(3) Euclidean distance (L2 metrics)

$$d(\boldsymbol{X},\boldsymbol{Y}) = L_{p=2}(\boldsymbol{X},\boldsymbol{Y}) = \| \boldsymbol{X} - \boldsymbol{Y} \| = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

(4) Squared euclidean distance (sum square error, SSE), mean square error (MSE)

$$d(\boldsymbol{X},\boldsymbol{Y}) = L_{p=2}^2(\boldsymbol{X},\boldsymbol{Y}) = \text{SSE} = \| \boldsymbol{X} - \boldsymbol{Y} \|^2 = \sum_{i=1}^{n} (x_i - y_i)^2$$

$$d(\boldsymbol{X},\boldsymbol{Y}) = \frac{1}{n} L_{p=2}^2(\boldsymbol{X},\boldsymbol{Y}) = \text{MSE} = \frac{1}{n}\sum_{i=1}^{n} (x_i - y_i)^2$$

(5) Angle-based distance

$$d(\boldsymbol{X},\boldsymbol{Y}) = -\cos(\boldsymbol{X},\boldsymbol{Y})$$

$$\cos(\boldsymbol{X},\boldsymbol{Y}) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{m} x_i^2 \sum_{i=1}^{m} y_i^2}}$$

(6) Correlation coefficient-based distance

$$d(\boldsymbol{X},\boldsymbol{Y}) = -r(\boldsymbol{X},\boldsymbol{Y})$$

$$r(\boldsymbol{X},\boldsymbol{Y}) = \frac{n\sum_{i=1}^{n} x_i y_i - \sum_{i=1}^{n} x_i \sum_{i=1}^{n} y_i}{\sqrt{\left(n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2\right)\left(n\sum_{i=1}^{n} y_i^2 - \left(\sum_{i=1}^{n} y_i\right)^2\right)}}$$

(7) Mahalanobis distance and Mahalanobis distance between normed vectors

$$d(\boldsymbol{X},\boldsymbol{Y}) = -\sum_{i=1}^{n} z_i x_i y_i$$

$$d(\boldsymbol{X},\boldsymbol{Y}) = -\frac{1}{\sqrt{\sum_{i=1}^{n} x_i^2}\sqrt{\sum_{i=1}^{n} y_i^2}}\sum_{i=1}^{n} z_i x_i y_i$$

here $z_i = \sqrt{\frac{\lambda_i}{\lambda_i+\alpha^2}}$, $\alpha = 0.25$, λ_i - corresponding eigenvalues, or simplified Mahalanobis distance versions with $z_i = \sqrt{\lambda_i/(\lambda_i+\alpha^2)} \simeq \sqrt{1/\lambda_i}$

(8) Weighted Manhattan distance

$$d(\boldsymbol{X},\boldsymbol{Y}) = \sum_{i=1}^{n} z_i|x_i - y_i|, z_i = \sqrt{1/\lambda_i}$$

(9) Weighted SSE distance

$$d(\boldsymbol{X},\boldsymbol{Y}) = \sum_{i=1}^{n} z_i(x_i - y_i)^2,\ z_i = \sqrt{1/\lambda_i}$$

(10) Weighted angle-based distance

$$d(\boldsymbol{X},\boldsymbol{Y}) = -\frac{\sum_{i=1}^{n} z_i x_i y_i}{\sqrt{\sum_{i=1}^{m} x_i^2 \sum_{i=1}^{m} y_i^2}},\ z_i = \sqrt{1/\lambda_i}$$

(11) Chi square distance

$$d(\boldsymbol{X},\boldsymbol{Y}) = \chi^2 = \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{x_i + y_i}$$

(12) Canberra distance

$$d(\boldsymbol{X},\boldsymbol{Y}) = \sum_{i=1}^{n} \frac{|x_i - y_i|}{|x_i| + |y_i|}$$

(13) Modified Manhattan distance

$$d(\boldsymbol{X},\boldsymbol{Y}) = \frac{\sum_{i=1}^{n} |x_i - y_i|}{\sum_{i=1}^{n} |x_i| \sum_{i=1}^{n} |y_i|}$$

(14) Modified SSE-based distance

$$d(\boldsymbol{X},\boldsymbol{Y}) = \frac{\sum_{i=1}^{n} (x_i - y_i)^2}{\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i^2}$$

(15) Weighted modified Manhattan distance

$$d(\boldsymbol{X},\boldsymbol{Y}) = \frac{\sum_{i=1}^{n} z_i|x_i - y_i|}{\sum_{i=1}^{n} |x_i| \sum_{i=1}^{n} |y_i|},\ z_i = \sqrt{1/\lambda_i}$$

(16) Weighted modified SSE-based distance

$$d(\boldsymbol{X},\boldsymbol{Y}) = \frac{\sum_{i=1}^{n} z_i (x_i - y_i)^2}{\sum_{i=1}^{n} x_i^2 \sum_{i=1}^{n} y_i^2},\ z_i = \sqrt{1/\lambda_i}$$

The choice of distance measure depends on the nature of the facial feature representations. Different algorithms and models may use specific distance measures that are well-suited to their design and objectives.

3.2.2 Experiments and results

For experiments we used the AT&T dataset containing 400 images from 40 distinct subjects (varying the lighting, facial expressions and facial details). The results of experiments are summarized in Table 1-2. We use accuracy and f1-score for measuring the overall goodness of all the distance measures. In these table, we can see how different distance measures affect the recognition accuracy and f1-score.

Table 1
Accuracy of experiments using distance measures on AT&T dataset.

Distance measure	PCA	Modular PCA	Kernel PCA	Sparse PCA	Randomized PCA	Incremental PCA
Euclidean	0.94	0.94	0.94	0.96	0.94	0.96
Weighted SSE	0.96	0.96	0.95	0.95	0.89	0.95
Weighted angle-based	0.95	0.95	0.95	0.95	0.91	0.95
Chi square	0.02	0.02	0.01	0.01	0.02	0.01
Canberra	0.92	0.92	0.93	0.93	0.92	0.93
Modified Mahattan	0.96	0.96	0.96	0.95	0.92	0.96
Modified SSE-based	**0.96**	**0.96**	**0.96**	**0.96**	0.92	0.96
Weighted modified Mahattan	0.92	0.92	0.92	0.92	0.87	0.92
Weighted modified SSE-based	0.96	0.96	0.95	0.95	0.89	0.95
Minkowski	0.94	0.94	0.94	0.96	0.94	**0.97**
Manhattan	0.94	0.93	0.94	0.96	0.93	0.96
Squared Euclidean	0.94	0.94	0.95	0.96	0.94	0.96
Angle-based	0.95	0.94	0.94	0.96	**0.95**	0.97
Correlation coefficient-based	0.94	0.94	0.84	0.96	0.94	0.96
Mahalanobis between normed vector	0.95	0.95	0.95	0.96	0.95	0.97
Weighted Manhattan	0.92	0.91	0.92	0.92	0.92	0.93

Table 2
F1-Score of experiments using distance measures on AT&T dataset.

Distance measure	PCA	Modular PCA	Kernel PCA	Sparse PCA	Randomized PCA	Incremental PCA
Euclidean	0.92	0.93	0.92	0.96	0.92	0.96
Weighted SSE	0.93	0.93	0.93	0.93	0.85	0.93
Weighted angle-based	0.91	0.91	0.91	0.91	0.86	0.91
Chi square	0.02	0.02	0.01	0.00	0.02	0.01
Canberra	0.88	0.88	0.88	0.88	0.88	0.88
Modified Mahattan	0.94	0.94	0.93	0.93	0.89	0.93
Modified SSE-based	**0.95**	**0.95**	**0.96**	**0.96**	0.88	0.96
Weighted modified Mahattan	0.89	0.89	0.89	0.88	0.83	0.89
Weighted modified SSE-based	0.93	0.93	0.93	0.93	0.85	0.93
Minkowski	0.92	0.92	0.92	0.95	0.92	**0.96**
Manhattan	0.91	0.91	0.92	0.93	0.91	0.94
Squared Euclidean	0.92	0.92	0.94	0.96	0.92	0.95
Angle-based	0.94	0.94	0.94	0.95	**0.94**	0.96
Correlation coefficient-based	0.94	0.94	0.81	0.95	0.94	0.95
Mahalanobis between normed vector	0.94	0.94	0.94	0.95	0.94	0.95
Weighted Manhattan	0.90	0.89	0.89	0.88	0.90	0.92

From the entire results table of experiments, we got the best results for each dimensionality reduction technique and put them in a line chart for easy viewing.

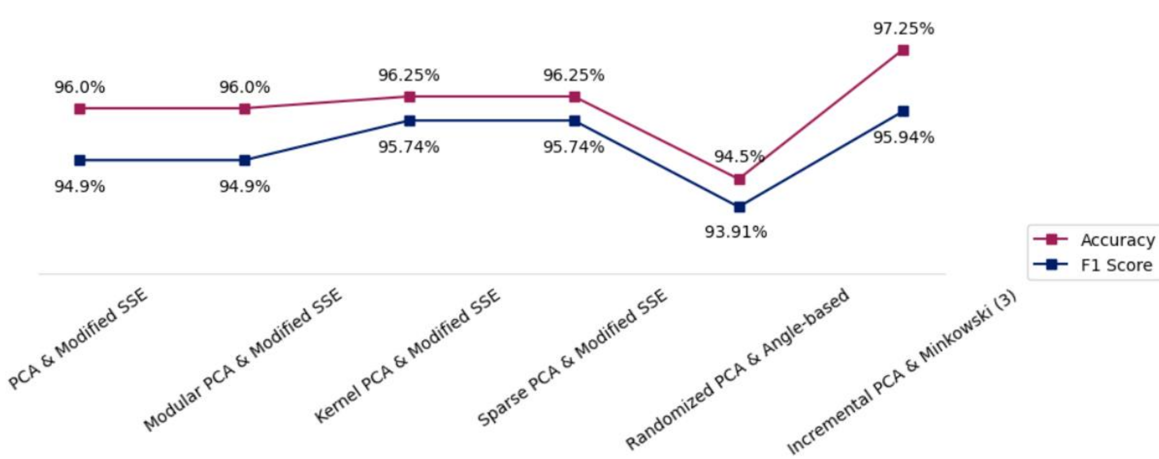

Figure 7. *Top-performing Distance Measures on AT&T dataset*

As the results are presented in table and line chart, it can be seen that Modified SSE-based distance is suitable for most of dimensionality reduction techniques that we used. But the best score for the whole task is Incremental PCA combined with Minkowski distance. It is reasonable, beacause Incremental PCA processes the massive dataset in different chunks; simultaneously, Minkowski distance (includes the Euclidean distance and the Manhattan distance as special case) can be tuned using the parameter p to adjust sensitivity to different dimensions.

*3.3 **Task 2:** Face recognition using machine learning classifiers*

3.3.1 Machine learning classifiers

In our quest to identify the most efficient model for face recognition on this dataset, we evaluated the performance of six widely-used machine learning classification models: Logistic Regression, Support Vector Machines (SVM), Naive Bayes, K-Nearest Neighbors (KNN), Decision Trees, and Random Forests.

These models generally exhibit strong performance on small datasets like the ORL face database. Subsequently, we will delve into the results obtained by applying these models to datasets processed using six different principal component analysis (PCA) variations.

3.3.2 Experiments and results

To facilitate further analysis, we present the performance metrics of each model after the tunning hyperparameters process with optuna in conjunction with different PCA techniques. The following table summarizes the accuracy and F1-score for each combination:

Table 3
Accuracy of models using machine learning classifiers on AT&T dataset.

Distance measure	PCA	Modular PCA	Kernel PCA	Sparse PCA	Randomized PCA	Incremental PCA
SVM	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**
Decision tree	0.66	0.69	0.74	0.74	0.66	0.73
K-Nearest Neighbor	0.99	0.99	0.99	0.99	0.99	0.99
Gaussian Naive Bayes	0.96	0.98	0.98	0.96	0.96	0.96
Logistic Regression	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**
Random Forest	0.99	0.99	0.99	0.99	0.99	0.98

Table 4
F1-Score of models using machine learning classifiers on LFW dataset.

Distance measure	PCA	Modular PCA	Kernel PCA	Sparse PCA	Randomized PCA	Incremental PCA
SVM	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**
Decision tree	0.67	0.68	0.72	0.72	0.62	0.66
K-Nearest Neighbor	0.99	0.99	0.99	0.99	0.99	0.99
Gaussian Naive Bayes	0.94	0.95	0.96	0.95	0.94	0.95
Logistic Regression	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**
Random Forest	0.97	0.97	0.97	0.97	0.97	0.96

Summary Classification Models on AT&T Datasets

(1) Support Vector Machine: The SVM model achieved perfect accuracy and F1-score. This indicates that it correctly classified all instances in the dataset.

(2) Decision Tree: The decision tree model's performance is relatively lower. Decision trees tend to overfit the training data, resulting in poor generalization to unseen examples.

(3) *K-Nearest Neighbors:* K-NN performed well, with high accuracy and F1-score. K-NN relies on local patterns in the data, making it effective when instances have similar neighbors. However, it can be sensitive to noise and outliers.

(4) Gaussian Naive Bayes: Gaussian Naive Bayes assumes that features are conditionally independent given the class label. Despite this simplifying assumption, it performs reasonably well. However, it may struggle with correlated features or complex relationships.

(5) Logistic Regression: Logistic Regression achieved perfect scores. This indicates that it correctly classified all instances in the dataset.

***(6) Random Forest*:** Random Forest combines multiple decision trees to improve performance. It provides good accuracy while mitigating overfitting.

3.4 ***Task 3:*** *Improve face recognition system by using Modular PCA and image enhancement techniques*

3.4.1 Image enhancement techniques

Image enhancement is a process that can be applied to photographs, scans, and digital images to improve their visual quality and overall appearance. Some objectives of image enhancement are using contrast adjustment, Gaussian blur, and Bilateral filter to reduce the noise of images.

(1) Contrast adjustment involves modifying the difference in intensity between the light and dark regions of an image to help correct defects or flaws in the image. The basic formulation for contrast adjustment can be expressed as:

$$dst\,(x,y) = saturatecast < uchar > (\alpha.src(x,y) + \beta)$$

Where:

- dst(x, y): pixel value at position (x, y) in the destination (output) image
- src(x, y): pixel value at position (x, y) in the source (input) image
- α: the contrast value.
  To lower the contrast, use $0 < \alpha < 1$. And for higher contrast use $\alpha > 1$
- β: the brightness value.
  A good range for brightness value is [-127, 127].
- saturatecast<uchar>: a function that clips the result to the range [0, 255] to ensure that the result is a valit unsigned 8-bit integer

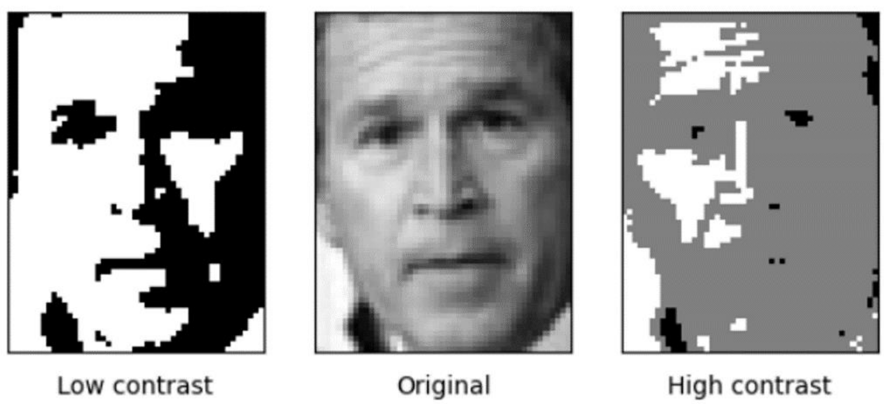

Figure 8. *Images using contrast adjustment*

(2) Gaussian blur uses a weighted average of neighboring pixels to smooth out noise and reduce artifacts. The formula for the Gaussian blur operation is as follows:

$$dst\,(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} \cdot \exp\left(-\frac{x^2+y^2}{2\sigma_x^2}\right) \cdot src(x,y)$$

Where:

- dst(x, y): pixel value at position (x, y) in the destination (output) image
- src(x, y): pixel value at position (x, y) in the source (input) image
- σ_x and σ_y are the kernel size, or the standard deviations in the X and Y directions, respectively. Larger kernel sizes and standard deviations result in stronger smoothing effects.

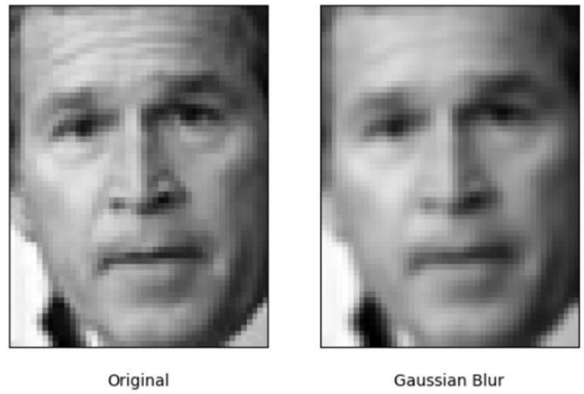

Figure 9. *Image using Gaussian Blur*

(3) Bilateral filter is used for smoothening images and reducing noise, while preserving edges. Its formulation is slightly more complex compared to Gaussian blur and defined as follows:

$$dst(x,y) = \frac{1}{W_c} \sum_{(i,j)\,\in\, neighborhood} G_{\sigma_s}(x-i, y-i) \cdot G_{\sigma_r}(src(i,j) - src(x,y)) \cdot src(i,j)$$

Where:

- dst(x, y): pixel value at position (x, y) in the destination (output) image
- src(x, y): pixel value at position (x, y) in the source (input) image
- $G_{\sigma_s}(x-i, y-i)$: the range weight for smoothing differences in coordinates
- $G_{\sigma_r}(src(i,j) - src(x,y))$: the range kernel for smoothing differences in intensities
- $G_{\sigma}(x,y) = \frac{1}{2\pi\sigma^2} \cdot \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$ is the Gaussian function.
- W_c: normalization term that is defined as:

$$W_c = \sum_{(i,j)\,\in\, neighborhood} G_{\sigma_s}(x-i, y-i) \cdot G_{\sigma_r}(src(i,j) - src(x,y))$$

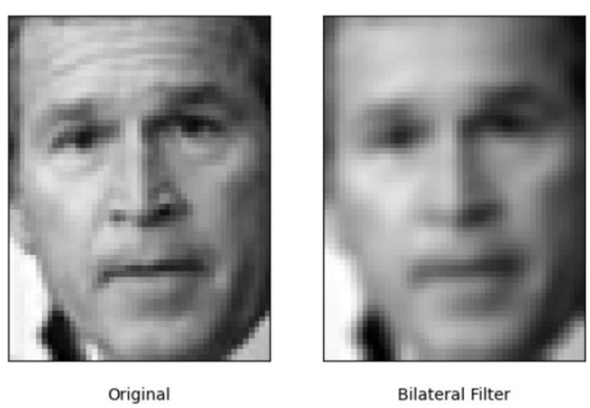

Figure 10. *Image using Bilateral filter*

Compared to *Gaussian blur*, the *Bilateral filter* has the normalization factor and the range weight as new terms. The Bilateral filter considers both the spatial proximity and intensity similarity between pixels when determining the smoothing effect. It ensures that only those pixels with intensity values similar to that of the central pixel are considered for blurring, while sharp intensity changes are maintained.

3.4.2 Experiments and results

For this experiment, we first compared the accuracy and f1-score of 6 different types of PCA combined with 16 distance measures and 6 machine learning classifiers. The purpose was to analyze the variation in posture, facial quality, hairstyles, and focus using images from the LFW dataset. The results are summarized in Tables 5–8.

Table 5
Accuracy of models using distance measures on LFW dataset.

Distance measure	PCA	Modular PCA	Kernel PCA	Sparse PCA	Randomized PCA	Incremental PCA
Euclidean	0.48	0.46	0.47	0.47	0.48	0.47
Weighted SSE	0.54	**0.63**	0.54	0.54	0.44	0.54
Weighted angle-based	0.52	0.54	0.50	0.51	0.56	0.50
Chi square	0.22	0.20	0.22	0.25	0.24	0.22
Canberra	**0.59**	0.58	**0.57**	**0.57**	**0.59**	**0.57**
Modified Mahattan	0.51	0.60	0.53	0.53	0.51	0.53
Modified SSE-based	0.48	0.46	0.47	0.47	0.50	0.47
Weighted modified Mahattan	0.51	0.57	0.51	0.51	0.45	0.51
Weighted modified SSE-based	0.54	**0.63**	0.54	0.54	0.44	0.54
Minkowski	0.46	0.43	0.46	0.46	0.46	0.46
Manhattan	0.51	0.55	0.53	0.53	0.51	0.53
Squared Euclidean	0.48	0.46	0.47	0.47	0.48	0.47
Angle-based	0.48	0.47	0.49	0.49	0.48	0.49
Correlation coefficient-based	0.49	0.47	0.49	0.49	0.49	0.50
Mahalanobis between normed vector	0.42	0.43	0.44	0.42	0.42	0.42
Weighted Manhattan	0.46	0.45	0.48	0.47	0.46	0.47

Table 6
Accuracy of models using machine learning classifiers on LFW dataset.

Distance measure	PCA	Modular PCA	Kernel PCA	Sparse PCA	Randomized PCA	Incremental PCA
SVM	**0.83**	0.83	**0.84**	**0.84**	0.82	0.82
Decision tree	0.42	0.47	0.43	0.43	0.45	0.44
K-Nearest Neighbor	0.59	0.59	0.58	0.58	0.60	0.59
Gaussian Naive Bayes	0.69	0.72	0.70	0.70	0.73	0.69
Logistic Regression	**0.83**	**0.84**	0.82	0.82	**0.83**	**0.83**
Random Forest	0.51	0.52	0.51	0.51	0.50	0.50

Table 7
F1-Score of models using distance measures on LFW dataset.

Distance measures	PCA	Modular PCA	Kernel PCA	Sparse PCA	Randomized PCA	Incremental PCA
Euclidean	0.34	0.32	0.35	0.35	0.34	0.35
Weighted SSE	0.43	0.48	0.43	0.43	0.32	0.43
Weighted angle-based	0.45	0.45	0.41	0.42	**0.48**	0.41
Chi square	0.14	0.12	0.14	0.13	0.16	0.12
Canberra	**0.46**	0.47	**0.45**	**0.47**	0.46	**0.45**
Modified Mahattan	0.35	**0.51**	0.39	0.39	0.41	0.39
Modified SSE-based	0.34	0.35	0.35	0.35	0.39	0.35
Weighted modified Mahattan	0.41	0.45	0.40	0.41	0.32	0.39
Weighted modified SSE-based	0.43	0.48	0.43	0.43	0.32	0.43
Minkowski	0.34	0.32	0.35	0.35	0.34	0.35
Manhattan	0.35	0.39	0.39	0.39	0.35	0.39
Squared Euclidean	0.34	0.32	0.35	0.35	0.34	0.35
Angle-based	0.38	0.36	0.39	0.39	0.38	0.39
Correlation coefficient-based	0.38	0.35	0.38	0.38	0.38	0.39
Mahalanobis between normed vector	0.29	0.31	0.32	0.29	0.29	0.29
Weighted Manhattan	0.36	0.37	0.38	0.36	0.36	0.36

Table 8
F1-Score of models using machine learning classifiers on LFW dataset.

Distance measure	PCA	Modular PCA	Kernel PCA	Sparse PCA	Randomized PCA	Incremental PCA
SVM	0.71	0.75	**0.75**	**0.75**	0.73	**0.73**
Decision tree	0.29	0.30	0.22	0.22	0.20	0.25
K-Nearest Neighbor	0.41	0.41	0.37	0.37	0.40	0.38
Gaussian Naive Bayes	0.59	0.60	0.60	0.60	0.62	0.59
Logistic Regression	**0.74**	**0.77**	0.73	0.73	**0.74**	**0.73**
Random Forest	0.22	0.24	0.23	0.23	0.21	0.22

The tables provided demonstrate how different distance measures and machine learning classifiers affect accuracy and f1-score. When combined with Modular PCA, both distance measures and machine learning classifiers perform well, especially Logistic Regression and Support Vector Machines. Based on the highest scores achieved, we have decided to use two models - Modular PCA combined with Weighted SSE and Linear Regression - as the foundation for improving scores with image enhancement techniques.

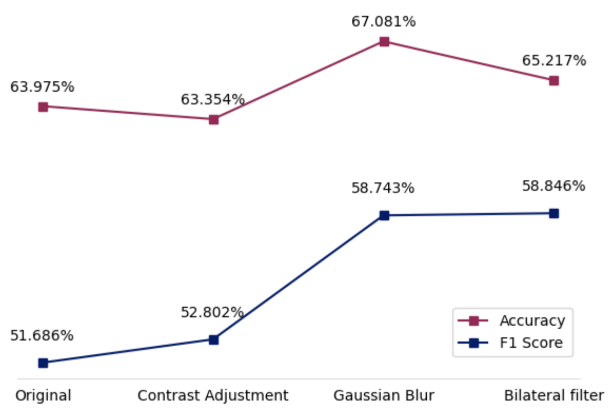

***Figure 11**. Top-performing models based on image enhancement using Modular PCA and Weighted SSE*

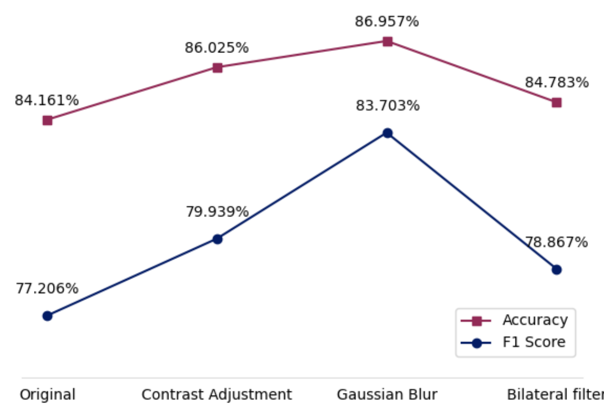

***Figure 12.** Top-performing models based on image enhancement using Modular PCA and Logistic Regression*

Applying image enhancement techniques, such as noise removal and filtering, demonstrably improves model performance in this case, with Gaussian blur yielding the best results. This suggests that pre-processing data to highlight relevant features and reduce noise through image enhancement can significantly benefit models.

4 Results

4.1 Metrics

We used two basic measures in the studies we conduct, Accuracy and F1-Score, to evaluate how effectively our models perform.

Accuracy is a metric used to measure the overall correctness of a model's predictions. It represents the ratio of correctly predicted instances to the total number of instances in the dataset. In other words, accuracy tells us how often the model's predictions are correct.

Mathematically, accuracy is calculated as:

$$Accuracy = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

On the other hand, the F1-Score, derived from the harmonic mean of precision and recall, offers a balanced assessment of the model's performance, particularly in scenarios with disparate class distributions. This metric ensures a nuanced evaluation by considering both precision, which measures the accuracy of positive predictions, and recall, which gauges the model's ability to capture all positive instances.

The formula for the F1-Score is:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.2 Results

Table 9

Best performance evaluation using Accuracy & F1-Score for two datasets.

Dataset	Accuracy	F1-Score
AT&T dataset	**1.0**	**1.0**
LFW dataset	0.86957	0.83703

The AT&T dataset, consisting of images of faces under controlled conditions, served as a benchmark for evaluating the model's accuracy and precision. We achieved an exceptional accuracy score of 1.0, indicating that our models accurately predicted all instances in the dataset. The corresponding F1-Score of 1.0 reaffirmed the reliability and effectiveness of our models in classifying faces with high precision and recall.

In contrast, the LFW dataset presented a more challenging scenario, featuring facial images captured under unconstrained conditions, including variations in lighting, pose, and facial expressions. Despite these complexities, our models demonstrated commendable performance, achieving an accuracy score of 0.86957 and an F1-Score of 0.83703. These results underscore the adaptability and robustness of our models in handling real-world scenarios and diverse facial attributes.

5 Conclusion

5.1 Research overview

In this study, we conducted comprehensive face recognition experiments employing various techniques including Principal Component Analysis (PCA), distance metrics, classification algorithms, and image enhancement processes. Our investigation revealed that Logistic Regression and Support Vector Machines exhibited superior performance compared to other methods when analyzing the AT&T dataset. Additionally, Modular PCA demonstrated remarkable efficacy when analyzing the LFW dataset. Furthermore, we observed that image enhancement techniques significantly enhanced recognition scores compared to the original dataset.

5.2 Future works

Our future research efforts will be focused on solving novel problems and investigating cutting-edge areas of facial recognition technology. In particular, the creation of algorithms that can efficiently handle 3D picture data and identify faces that are unknown will be the top priority in our future work. Through further exploration of these fields, we want to improve the resilience and adaptability of face recognition systems, opening the door for creative uses in a variety of fields.

References

Gottumukkal, R., & Asari, V. K. (2004). An improved face recognition technique based on modular PCA approach. *Pattern Recognition Letters*, *25*(4), 429-436.

Perlibakas, V. (2004). Distance measures for PCA-based face recognition. *Pattern recognition letters*, *25*(6), 711-724.

Faruqe, M. O., & Hasan, M. A. M. (2009, August). Face recognition using PCA and SVM. In *2009 3rd international conference on anti-counterfeiting, security, and identification in communication* (pp. 97-101). IEEE.

Müge Çarıkçı, Figen Özen (2011). A face recognition system based on eigenfaces method.

üge Çarıkçı, M., & Özen, F. (2012). A face recognition system based on eigenfaces method. *Procedia Technology*, *1*, 118-123.

Adjabi, I., Ouahabi, A., Benzaoui, A., & Taleb-Ahmed, A. (2020). Past, present, and future of face recognition: A review. *Electronics*, *9*(8), 1188.

Zhang, W., Zhao, X., Morvan, J. M., & Chen, L. (2018). Improving shadow suppression for illumination robust face recognition. *IEEE transactions on pattern analysis and machine intelligence*, *41*(3), 611-624.

Bah, S. M., & Ming, F. (2020). An improved face recognition algorithm and its application in attendance management system. *Array*, *5*, 100014.

Paul, L. C., & Al Sumam, A. (2012). Face recognition using principal component analysis method. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, *1*(9), 135-139.

De Mel, V. L. B. Survey of Evaluation Metrics in Facial Recognition Systems.

Jafri, R., & Arabnia, H. R. (2009). A survey of face recognition techniques. *journal of information processing systems*, *5*(2), 41-68.

Rusia, M. K., & Singh, D. K. (2023). A comprehensive survey on techniques to handle face identity threats: challenges and opportunities. *Multimedia Tools and Applications*, *82*(2), 1669-1748.

Kim, H. I., Lee, S. H., & Yong, M. R. (2015, September). Face image assessment learned with objective and relative face image qualities for improved face recognition. In *2015 IEEE International Conference on Image Processing (ICIP)* (pp. 4027-4031). IEEE.

Turk, M. A., & Pentland, A. P. (1991, January). Face recognition using eigenfaces. In *Proceedings. 1991 IEEE computer society conference on computer vision and pattern recognition* (pp. 586-587). IEEE Computer Society.

Turk, M. (1991). Pentland. Eigenfaces for recognition. *K. Cogn. Neurosci*, *4*, 72-86.

Lawrence, S., Giles, C. L., Tsoi, A. C., & Back, A. D. (1997). Face recognition: A convolutional neural-network approach. *IEEE transactions on neural networks*, *8*(1), 98-113.

Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1701-1708).

Chellappa, R., Wilson, C. L., & Sirohey, S. (1995). Human and machine recognition of faces: A survey. *Proceedings of the IEEE*, *83*(5), 705-741.