Efficient Face Detection and Recognition with PCA and Eigenfaces: A Comprehensive Analysis

Md. Ibrahim Abdullah
Department of Computer Science and Engineering, Islamic University, Kushtia, Bangladesh
Email: ibrahim@cse.iu.ac.bd

Md. Nazrul Islam
Department of Computer Science and Engineering, Islamic University, Kushtia, Bangladesh
Email: shilunazrul@gmail.com

Diponkor Bala
Department of Computer Science and Engineering, Islamic University, Kushtia, Bangladesh
Email: diponkor.b@gmail.com

Mohammad Alamgir Hossain
School of Information Science and Technology,
University of Science and Technology of China (USTC), Anhui, Hefei, China
Email: alamgir@mail.ustc.edu.cn

Md. Atiqur Rahman
Department of Institute of Genetics and Developmental Biology,
University of Chinese Academy of Sciences (UCAS), Beijing, China
Email: atique@genetics.ac.cn

Md. Shohidul Islam
Department of Computer Science and Engineering, Islamic University, Kushtia, Bangladesh
Email: shohid7@cse.iu.ac.bd

ABSTRACT

Face detection and recognition are critical tasks in computer vision with applications in security systems, biometric authentication, and human-computer interaction. This paper presents a comprehensive study leveraging Principal Component Analysis (PCA) and Eigenfaces for efficient dimensionality reduction and compact, discriminative facial feature representation. The study introduces a robust pipeline integrating preprocessing, feature extraction, and efficient training. Using the CelebA dataset for training and the LFW dataset for evaluation, the system addresses real-world challenges, including variations in lighting, expressions, and poses. The performance is analyzed across configurations, exploring the trade-off between dimensionality reduction and recognition accuracy. Experimental results demonstrate that the PCA-based approach achieves high recognition accuracy while maintaining computational efficiency, making it suitable for resource-constrained environments. The findings highlight the system's robustness, scalability, and practical applicability in both constrained and real-time scenarios. This work concludes with an analysis of strengths and limitations and offers recommendations for integrating non-linear techniques and advanced learning models to further enhance scalability, accuracy, and real-world performance.

Keywords - PCA, Eigenfaces, Face Detection, Face Recognition, Machine Learning, Computer Vision

Date of Submission: 01.06.2025 Date of Acceptance: 26.06.2025

1 Introduction

Face detection and recognition are critical tasks in computer vision with widespread applications, including biometric authentication, access control, video surveillance, and human-computer interaction. These tasks remain challenging due to variations in lighting, facial expressions, occlusions, and pose. The development of computationally efficient and robust algorithms for accurate face recognition continues to be a pivotal area of research.

This work focuses on a classical yet practical approach to face recognition using **Principal Component Analysis (PCA)** and **Eigenfaces**. PCA is a powerful dimensionality reduction technique that projects high-dimensional data into a lower-dimensional subspace while preserving significant variations. Eigenfaces, derived from PCA, capture the principal components of facial images, forming a compact and discriminative representation of facial features. Unlike modern deep learning approaches, PCA-based methods offer interpretability, lightweight computation, and scalability, making them suitable for real-time systems and resource-constrained environments. Such systems have shown promise in enhancing computational efficiency in various domains like smart IoT applications and healthcare, demonstrating their real-world applicability in resource-limited scenarios [12] and [1].

1.1 Problem Statement

Despite advances in deep learning-based face recognition methods, the high computational and storage requirements of these models limit their applicability in real-time and resource-constrained scenarios. Furthermore, achieving interpretability in recognition systems remains a challenge with such models. Classical methods like PCA and Eigenfaces present a promising alternative by offering interpretable, computationally efficient, and scalable solutions, but they face challenges in handling real-world variations in lighting, pose, and expression.

This study addresses these challenges by implementing a PCA-based face recognition pipeline tailored for robustness and efficiency. Key questions include: How well can PCA and Eigenfaces handle real-world variations? What is the optimal balance between dimensionality reduction and recognition performance? How does this classical approach compare to modern methods?

1.2 Background and Motivation

Face recognition has been an integral topic in computer vision since the introduction of Eigenfaces by Turk and Pentland. Traditional PCA-based methods gained prominence due to their simplicity and efficiency. However, with the advent of deep learning, attention has shifted to neural networks, which offer state-of-the-art accuracy but demand significant computational resources and large datasets for training. This shift has created a research gap for lightweight, interpretable models suitable for constrained environments.

The motivation behind this study lies in revisiting PCA and Eigenfaces to explore their relevance in modern applications. By optimizing preprocessing steps and systematically evaluating performance on diverse datasets, this work aims to demonstrate that classical methods can still play a vital role in face recognition, especially in applications requiring real-time execution and computational efficiency. Hybrid models, combining classical and deep learning methods, such as those seen in healthcare analytics [1], are emerging as effective solutions in balancing performance with efficiency.

1.3 Objectives and Significance of the Study

The study aims to develop a robust PCA-based recognition pipeline for face detection and recognition, evaluating its performance across real-world conditions such as lighting, pose, and expression. It also seeks to optimize the balance between efficiency and accuracy while comparing PCA with modern deep learning models in terms of computational efficiency, accuracy, and scalability.

By leveraging PCA and Eigenfaces, the research bridges the gap between computational efficiency and recognition accuracy. The proposed system offers reduced computational and storage footprints, making it suitable for real-time applications with limited resources. It provides interpretable results, aiding system validation, and is scalable to diverse datasets without the need for extensive retraining. Additionally, the study provides insights into optimizing dimensionality reduction for efficient and accurate face recognition.

By revisiting PCA and Eigenfaces in the context of modern challenges, this work highlights their continued relevance and identifies avenues for integrating these methods with advanced techniques to enhance scalability, robustness, and real-world performance.

2 Related Work

Face detection and recognition have been pivotal research areas in computer vision, with numerous methods developed to improve robustness, accuracy, and efficiency. This section provides an overview of major approaches, including traditional methods based on **Principal Component Analysis (PCA)**, modern deep learning approaches, and the datasets commonly used for benchmarking.

2.1 Overview of Face Recognition Approaches

Face recognition methods have evolved significantly over the past decades, transitioning from classical statistical techniques to deep learning-based models. Early methods relied on hand-crafted features, such as Eigenfaces [14], Fisherfaces [3], and Local Binary Patterns (LBP) [2], to encode facial features for recognition. While computationally efficient, these approaches struggled with variations in lighting, pose, and expression.

In recent years, deep learning has revolutionized the field with models like DeepFace [13], FaceNet [10], and VGGFace [8], which leverage convolutional neural networks (CNNs) to learn hierarchical representations of facial features. Despite their high accuracy, these models require extensive computational resources and large labeled datasets, limiting their feasibility for real-time or resource-constrained applications.

Hybrid approaches that combine traditional methods, such as PCA, with modern deep learning techniques have also emerged, offering a balance between efficiency and accuracy [16]. These developments highlight the diversity of face recognition approaches and the trade-offs between accuracy, interpretability, and computational complexity.

2.2 Traditional PCA-Based Approaches

The introduction of the Eigenfaces approach by Turk and Pentland [14] marked a significant milestone in face recognition. By leveraging PCA, this method reduced the dimensionality of facial images while retaining discriminative features, enabling efficient recognition. However, its sensitivity to variations in lighting, pose, and expression necessitated further enhancements.

Belhumeur et al. [3] extended PCA with Linear Discriminant Analysis (LDA) in the Fisherfaces method, optimizing class separability to improve robustness. Probabilistic approaches, such as those by Moghaddam and Pentland [7], incorporated statistical modeling to address uncertainties in recognition tasks. Additionally, Kernel PCA [9] addressed the linearity constraints of traditional PCA by projecting data into a higher-dimensional feature space, capturing nonlinear variations in facial data.

Hybrid methods have further improved PCA-based recognition. For example, PCA combined with LBP [2] and Support Vector Machines (SVM) [4] has demonstrated enhanced accuracy while maintaining computational efficiency. Shakya and Shrestha [11] revisited PCA for small-scale systems, emphasizing its relevance for resource-constrained applications. These advancements underscore the versatility of PCA-based methods for lightweight and interpretable face recognition systems.

2.3 Modern Deep Learning Approaches

Deep learning has transformed face recognition by enabling the learning of robust feature representations directly from data. Models such as DeepFace [13], FaceNet [10], and VGGFace [8] utilize CNN architectures to extract hierarchical features that are invariant to lighting, pose, and expression variations. These models achieve state-of-the-art performance on benchmark datasets, such as Labeled Faces in the Wild (LFW), but require significant computational resources and large-scale labeled datasets for training.

Recent advancements have focused on improving the efficiency and scalability of deep learning-based systems. Lightweight architectures, such as MobileFaceNet, aim to reduce model size while retaining high accuracy. Additionally, hybrid methods integrating PCA with deep learning [16] have been explored to leverage the strengths of both approaches. These systems use PCA for dimensionality reduction and deep learning for feature extraction, offering a balance between interpretability, efficiency, and performance.

Despite their success, deep learning models are often viewed as black-box solutions, lacking the interpretability of classical methods like PCA. This motivates ongoing research into integrating traditional techniques with modern approaches to achieve robust and explainable face recognition systems.

2.4 Datasets for Face Recognition: CelebA and LFW

Datasets play a crucial role in evaluating face recognition systems, providing benchmarks for assessing performance under varying conditions. Two widely used datasets are CelebA and Labeled Faces in the Wild (LFW):

CelebA: The CelebA dataset [6] contains over 200,000 celebrity images with 40 attribute annotations, making it a comprehensive resource for training face recognition systems. Its diversity in facial expressions, poses, and lighting conditions allows for evaluating robustness under real-world scenarios. CelebA is particularly suitable for training due to its large size and extensive annotations.

LFW: The LFW dataset [5] comprises over 13,000 images of faces captured in unconstrained environments, with variations in pose, expression, and occlusion. Designed for evaluating face verification systems, LFW serves as a benchmark for assessing recognition accuracy in real-world conditions. Its challenging nature makes it a valuable resource for testing the generalizability of face recognition methods.

These datasets have facilitated consistent comparisons across traditional PCA-based approaches and modern deep learning models, driving advancements in the field. They also highlight the trade-offs between computational efficiency and recognition performance, particularly in resource-constrained settings.

3 Methodology

This section details the theoretical foundation, techniques, and pipeline for implementing the proposed face detection and recognition system. The methodology incorporates Principal Component Analysis (PCA) for dimensionality reduction and Eigenfaces for feature extraction. Preprocessing, detection, and classification techniques are also discussed to provide a comprehensive understanding of the system.

3.1 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a linear dimensionality reduction technique widely utilized in computer vision tasks. In the context of face recognition, PCA projects high-dimensional facial image data into a lower-dimensional subspace while preserving the most significant variations. This section elaborates on the mathematical framework, dimensionality reduction, and reconstruction process in PCA.

PCA operates by identifying the directions of maximum variance in the dataset and projecting the data onto these directions, called principal components. Let a dataset of N facial images, each represented as a d-dimensional vector, be denoted as $\mathbf{X} \in \mathbb{R}^{N \times d}$. The mathematical steps are as follows:

Mean Centering: The mean image vector $\bar{\mathbf{x}}$ is computed:

$$\bar{\mathbf{x}} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i, \tag{1}$$

where $\mathbf{x}_i \in \mathbb{R}^d$ represents the i-th image vector. The mean vector is subtracted from each image to obtain the centered data matrix $\tilde{\mathbf{X}}$:

$$\tilde{\mathbf{X}} = \mathbf{X} - \bar{\mathbf{x}}. \tag{2}$$

Covariance Matrix: The covariance matrix $\mathbf{C} \in \mathbb{R}^{d\times d}$ is computed to capture the relationships between pixel intensities:

$$\mathbf{C} = \frac{1}{N}\tilde{\mathbf{X}}^\top\tilde{\mathbf{X}}. \tag{3}$$

Eigenvalue Decomposition: PCA solves the eigenvalue problem:

$$\mathbf{C}\mathbf{u}_k = \lambda_k\mathbf{u}_k, \tag{4}$$

where λ_k is the eigenvalue and $\mathbf{u}_k$ is the corresponding eigenvector. Eigenvectors with the largest eigenvalues represent directions of maximum variance.

Projection Matrix: The top K eigenvectors are selected to form the projection matrix $\mathbf{U} \in \mathbb{R}^{d\times K}$:

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_K]. \tag{5}$$

By projecting the data onto the subspace spanned by $\mathbf{U}$, PCA reduces the dimensionality while retaining critical information:

$$\mathbf{y} = \mathbf{U}^\top(\mathbf{x} - \bar{\mathbf{x}}), \tag{6}$$

where $\mathbf{y} \in \mathbb{R}^K$ is the lower-dimensional representation of the facial image.

3.1.1 Dimensionality Reduction in PCA

PCA achieves dimensionality reduction by transforming the high-dimensional dataset into a lower-dimensional subspace defined by the principal components. This transformation retains the directions of maximum variance, ensuring that the most critical information is preserved.

The reduction process involves selecting the number of principal components K such that the cumulative explained variance exceeds a predefined threshold (e.g., 95%). The cumulative variance is computed as:

$$\text{Cumulative Variance} = \frac{\sum_{k=1}^{K}\lambda_k}{\sum_{j=1}^{d}\lambda_j}, \tag{7}$$

where λ_k is the eigenvalue corresponding to the k-th principal component. Figure 12 illustrates how the cumulative explained variance increases with the number of components.

Dimensionality reduction significantly reduces computational requirements, making the system efficient for real-time applications. For example, reducing the dimensionality from 4096 (corresponding to 64×64 images) to 50 components can lower storage and computation costs while retaining high recognition accuracy.

3.1.2 Reconstruction with PCA

PCA enables the reconstruction of facial images from their lower-dimensional representations, demonstrating the fidelity of dimensionality reduction. The reconstruction process involves projecting the data back to the original space using the top K principal components:

$$\hat{\mathbf{x}} = \bar{\mathbf{x}} + \mathbf{U}\mathbf{y}, \tag{8}$$

where $\hat{\mathbf{x}} \in \mathbb{R}^d$ is the reconstructed image, $\mathbf{y}$ is the lower-dimensional representation, and $\bar{\mathbf{x}}$ is the mean image vector.

Reconstruction error quantifies the loss of information during dimensionality reduction and is defined as the mean squared error (MSE) between the original and reconstructed images:

$$\text{Reconstruction Error} = \frac{1}{d}\|\mathbf{x} - \hat{\mathbf{x}}\|^2. \tag{9}$$

Figure 1 illustrates progressive reconstruction of a sample image using an increasing number of principal components. The reconstruction quality improves with more components, demonstrating the trade-off between dimensionality reduction and fidelity.

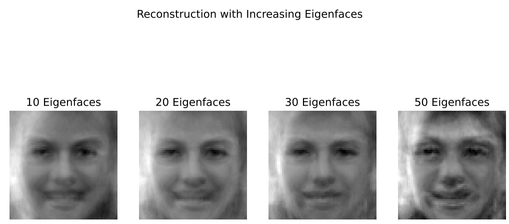

Figure 1: Reconstruction of a facial image using an increasing number of principal components. As the number of components increases, the reconstruction quality improves, but computational cost also rises.

The reconstruction capability of PCA highlights its effectiveness in retaining essential features, making it suitable for tasks like facial recognition and compression.

3.2 Eigenfaces

Eigenfaces are the principal components of a facial image dataset derived using Principal Component Analysis (PCA). Each Eigenface represents a unique mode of variation within the dataset, capturing the most significant patterns in facial features. These patterns are not individual facial components, such as eyes or noses, but rather a weighted combination of pixel intensity variations across the dataset. Eigenfaces form a basis for the "face space," where facial images can be represented as linear combinations of these principal components.

By reducing the dimensionality of the data while preserving its most discriminative features, Eigenfaces enable computational efficiency and improve recognition accuracy. This section delves into the concept, generation, and application of Eigenfaces in face recognition systems.

The concept of Eigenfaces was introduced by Turk and Pentland [14], marking a significant milestone in the field of face recognition. Eigenfaces represent the directions of maximum variance within a facial dataset. These directions are computed by applying PCA to the covariance matrix of the dataset, with each Eigenface corresponding to a principal component.

Mathematically, an Eigenface $\mathbf{u}_k$ is derived as an eigenvector of the covariance matrix $\mathbf{C}$:

$$\mathbf{C}\mathbf{u}_k = \lambda_k \mathbf{u}_k, \tag{10}$$

where λ_k is the eigenvalue associated with the eigenvector $\mathbf{u}_k$. Larger eigenvalues indicate greater variance captured by the corresponding eigenvector.

Each facial image in the dataset can be expressed as a linear combination of Eigenfaces:

$$\mathbf{x} = \bar{\mathbf{x}} + \sum_{k=1}^{K} w_k \mathbf{u}_k, \tag{11}$$

where $\bar{\mathbf{x}}$ is the mean face, $\mathbf{u}_k$ are the Eigenfaces, and w_k are the weights that define the contribution of each Eigenface to the image.

The Eigenfaces provide a compact and discriminative representation of facial features, enabling efficient storage, processing, and comparison of facial images.

3.2.1 Generation of Eigenfaces

The process of generating Eigenfaces involves the following steps:

Dataset Preparation: Facial images are preprocessed by aligning, cropping, and resizing them to a uniform dimension, such as 64×64 pixels. Each image is then flattened into a one-dimensional vector.

Mean Face Computation: The mean face $\bar{\mathbf{x}}$ is calculated by averaging all image vectors in the dataset:

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_i, \tag{12}$$

where N is the number of images.

Data Centering: Each image vector $\mathbf{x}_i$ is centered by subtracting the mean face:

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i - \bar{\mathbf{x}}. \tag{13}$$

Covariance Matrix Computation: The covariance matrix $\mathbf{C}$ of the centered data is computed as:

$$\mathbf{C} = \frac{1}{N} \tilde{\mathbf{X}}^T \tilde{\mathbf{X}}, \tag{14}$$

where $\tilde{\mathbf{X}}$ is the matrix of centered image vectors.

Eigenvalue Decomposition: Eigenvectors $\mathbf{u}_k$ and their corresponding eigenvalues λ_k are obtained by solving the eigenvalue problem for $\mathbf{C}$.

Selection of Principal Components: The top K eigenvectors, corresponding to the largest eigenvalues, are selected to form the Eigenfaces.

Reshaping: The Eigenfaces are reshaped from vectors back into the original image dimensions for visualization and analysis.

Figure 7 illustrates the first ten Eigenfaces generated from a training dataset. Each Eigenface captures a distinct pattern of variation, such as lighting gradients, facial orientations, or expressions.

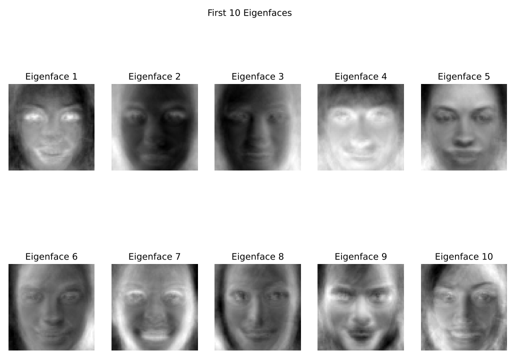

Figure 2: Visualization of the first ten Eigenfaces generated from the training dataset. These Eigenfaces represent the primary modes of variation in facial features.

3.3 Face Detection and Preprocessing

Face detection and preprocessing are essential steps in the face recognition pipeline to ensure consistency, alignment, and uniformity of input images. This section discusses the use of the Haar Cascade for face detection and the subsequent steps for cropping and aligning faces, preparing them for Principal Component Analysis (PCA).

3.3.1 Haar Cascade for Face Detection

The Haar Cascade classifier, introduced by Viola and Jones [15], is a widely used method for real-time face detection due to its efficiency and robustness. The classifier relies on a cascade of simple features, computed using integral images, to detect facial regions in an input image. These features include edge, line, and rectangle patterns that capture the structural characteristics of a face.

Working Mechanism:

1. **Feature Extraction:** Haar-like features are computed by summing pixel intensities in adjacent rectangular regions of the image. These features are sensitive to contrasts between regions, such as the difference between the eyes and the cheeks.

2. **Integral Image Representation:** The integral image is a data structure that allows rapid computation of Haar-like features by enabling constant-time summation of pixel values in rectangular regions.

3. **Cascade of Classifiers:** The Haar Cascade consists of multiple stages of classifiers arranged in a cascade.

Each stage filters out non-face regions, progressively narrowing down the search to regions most likely to contain a face.

4. **Sliding Window:** The classifier scans the image using a sliding window approach, evaluating each window at different scales to detect faces of varying sizes.

Visualization of Face Detection: To demonstrate the system's performance, face detection was applied to group images containing faces at varying scales and orientations. The results illustrate the system's ability to detect faces consistently under different conditions.

Figure 3: Group Detected Faces: The Haar Cascade classifier successfully identifies faces in a group setting with diverse poses and lighting conditions.

Figure 4: Group Scale Detected Faces: Faces detected across multiple scales in a single image, demonstrating the classifier's ability to handle size variability.

3.3.2 Cropping and Alignment of Faces

After detecting faces, the next step is to crop and align the face regions to ensure consistency across the dataset. Cropping removes irrelevant parts of the image, such as the background, while alignment ensures that facial features are consistently positioned in the cropped region. These steps are crucial for accurate and efficient feature extraction using PCA.

Steps for Cropping and Alignment:

1. **Bounding Box Extraction:** The coordinates of the detected face regions, provided by the Haar Cascade, are used to extract bounding boxes.

2. **Cropping the Face:** The bounding boxes are used to crop the facial region from the original image.

3. **Resizing to Target Size:** Each cropped face is resized to a fixed dimension (e.g., 64×64 pixels) to standardize the input size for PCA processing.

4. **Fallback for Non-Detection:** If no face is detected in an image, the entire image is resized to the target size as a fallback mechanism, ensuring compatibility with the PCA pipeline.

Visualization of Preprocessing: Figure 5 illustrates the preprocessing steps, including face detection, cropping, and alignment.

Figure 5: Face preprocessing steps: (a) Original image, (b) Detected face with bounding box, (c) Cropped face, (d) Aligned face resized to 64×64 pixels.

3.4 Recognition Pipeline

The recognition pipeline is the core of the face recognition system, utilizing the compact and discriminative features extracted through PCA. This section outlines the process of projecting facial images into the PCA subspace, performing classification using nearest neighbor search, and evaluating system performance with suitable metrics.

3.4.1 Projection into PCA Subspace

Projection into the PCA subspace transforms high-dimensional facial images into a lower-dimensional space spanned by the principal components. This step reduces the computational complexity of subsequent classification tasks while retaining the most significant features for recognition.

Let $\mathbf{x} \in \mathbb{R}^d$ be the original high-dimensional image vector, where d is the number of pixels. PCA computes the projection $\mathbf{y} \in \mathbb{R}^K$ into a subspace of dimension K as:

$$\mathbf{y} = \mathbf{U}^T(\mathbf{x} - \bar{\mathbf{x}}), \tag{15}$$

where: $\mathbf{U} \in \mathbb{R}^{d \times K}$ is the matrix containing the top K principal components (Eigenfaces), $\bar{\mathbf{x}} \in \mathbb{R}^d$ is the mean face vector of the training dataset, and $\mathbf{y}$ is the projected vector representing the image in the PCA subspace.

3.4.2 Classification with Nearest Neighbor Search

Classification in the recognition pipeline is performed by comparing the PCA-projected test images with the training images. The nearest neighbor (NN) search identifies the closest match in the PCA subspace based on a distance metric.

Methodology: The Euclidean distance is commonly used as the similarity metric:

$$d(\mathbf{y}_{test}, \mathbf{y}_{train}) = \|\mathbf{y}_{test} - \mathbf{y}_{train}\|_2, \tag{16}$$

where $\mathbf{y}_{test}$ is the PCA-projected vector of the test image, $\mathbf{y}_{train}$ is the PCA-projected vector of the training image, and $d(\cdot, \cdot)$ is the Euclidean distance between the two vectors.

The training image with the smallest distance to the test image is identified as the closest match, and its label is assigned to the test image.

3.5 Dataset Preparation

Proper dataset preparation is crucial to the success of any face recognition system. This involves curating, preprocessing, and organizing data to ensure consistency and compatibility with the PCA-based pipeline.

Datasets Used: The CelebA (CelebFaces Attributes) dataset is used as the primary training dataset. It contains diverse images of celebrity faces with annotations, covering variations in pose, expression, and lighting. The Labeled Faces in the Wild (LFW) dataset is used for testing and evaluation. It provides real-world facial images with labels, enabling the assessment of the system's performance under unconstrained conditions.

Preprocessing Steps: To ensure that the datasets are compatible with the PCA-based system, the following preprocessing steps are performed. Face detection and cropping are performed using OpenCV's Haar Cascade classifier to detect faces. Detected face regions are cropped and resized to 64×64 pixels to maintain uniformity. All images are converted to grayscale to simplify the data and focus on structural features. This reduces computational complexity without compromising recognition performance. Each 64×64 grayscale image is flattened into a one-dimensional vector of size 4096 (64×64) for compatibility with PCA input requirements. Pixel intensities are normalized to a range of [0, 1] by dividing each value by 255. This standardization ensures numerical stability during PCA computations.

Directory Structure: The datasets are organized into directories for streamlined access during training and testing. The `CelebA_Train` directory contains the preprocessed training images from the CelebA dataset. The `LFW_Test` directory contains the preprocessed test images from the LFW dataset, organized by class (e.g., individual names).

4 Experimental Results

This section presents the experimental results obtained from the PCA-based face detection and recognition system. Various visualizations and analyses are provided to assess the system's effectiveness, including the representation of Eigenfaces, reconstruction fidelity, recognition accuracy, and feature distribution in the PCA subspace.

4.1 Visualization of Eigenfaces

Eigenfaces are the principal components derived from applying Principal Component Analysis (PCA) to the facial image dataset. Each Eigenface represents a unique mode of variation in the dataset, highlighting patterns such as lighting, pose, and facial structure. In this experiment, the top 20 Eigenfaces were computed from the training dataset, providing insight into the most significant features captured by PCA.

Mean Face Visualization: As part of the PCA process, the mean face is computed by averaging all aligned and preprocessed facial images in the dataset. This mean face serves as the baseline from which Eigenfaces are derived, capturing the general structure of the dataset.

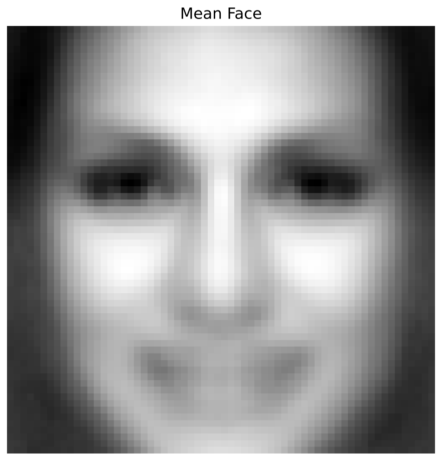

Figure 6: Mean Face computed from the training dataset. This image represents the average of all preprocessed faces, serving as the starting point for PCA and Eigenface generation.

Methodology: The training dataset was preprocessed by aligning, cropping, and resizing all facial images to 64×64 pixels. PCA was then applied to the flattened images to compute the Eigenfaces, which were subsequently reshaped to their original dimensions (64×64) for visualization. Additionally, mean faces for each class were calculated by averaging all facial images within the respective class, providing insights into class-level variations and the overall structure of the dataset.

Results: Figure 7 shows the first 20 Eigenfaces derived from the training dataset. These Eigenfaces represent the directions of maximum variance in the dataset, ordered by their corresponding eigenvalues.

Int. J. Advanced Networking and Applications
Volume: 17 Issue: 01 Pages: 6705-6718 (2025) ISSN: 0975-0290

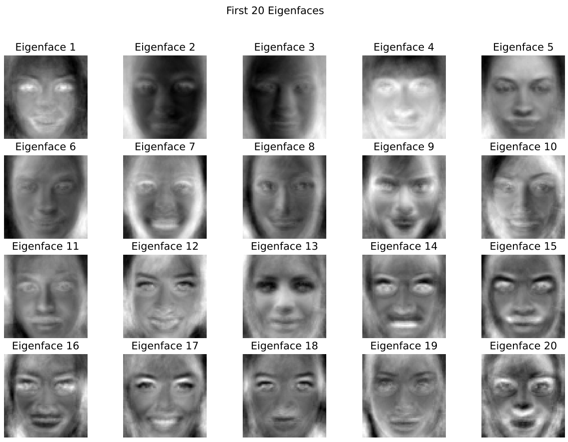

Figure 7: Visualization of the first 20 Eigenfaces. Each Eigenface captures a distinct pattern of variation in the dataset, such as brightness gradients, pose shifts, or facial symmetry.

Figure 8 displays the mean faces for several classes in the dataset. These mean faces highlight the dominant facial structures and shared characteristics within each class, providing an additional perspective on the variability captured by PCA.

Figure 8: Mean Faces for Selected Classes. Each mean face represents the averaged features of all facial images within a class, highlighting shared patterns and dominant structures.

Analysis and Significance: The first few Eigenfaces capture global features like lighting and symmetry, while later ones represent finer details such as expressions or texture variations. The mean faces provide a summary of class-level traits, revealing consistency within classes and variation across them. Eigenfaces serve as abstract representations that efficiently reduce facial data to a compact subspace, enhancing both recognition and interpretability. The visualizations validate the PCA process, offering insights into dominant variations and class-level characteristics. The modularity of Eigenfaces and interpretability of mean faces make them valuable for dimensionality reduction, feature extraction, and visualization tasks.

4.2 Reconstruction with Increasing Eigenfaces

The reconstruction of facial images using PCA is directly influenced by the number of Eigenfaces retained. Each Eigenface represents a principal component, and the reconstruction fidelity improves as more components are used. This experiment evaluates the trade-off between dimensionality reduction and reconstruction quality by progressively increasing the number of Eigenfaces.

Methodology: A sample facial image from the test dataset was preprocessed by performing alignment, cropping, and resizing to 64×64 pixels. The PCA model trained on the CelebA dataset was then used to reconstruct the sample image using varying numbers of Eigenfaces: 5, 10, 20, 30, 40, and 50. Reconstruction was carried out by projecting the image into the PCA subspace and reconstructing it back to the original dimensional space using the selected number of Eigenfaces.

The reconstructed image $\hat{\mathbf{x}}$ is computed as:

$$\hat{\mathbf{x}} = \bar{\mathbf{x}} + \mathbf{U}_K \mathbf{y}_K, \tag{17}$$

where $\bar{\mathbf{x}}$ is the mean face vector, $\mathbf{U}_K$ represents the top K Eigenfaces (principal components), and $\mathbf{y}_K$ is the projection of the image in the reduced-dimensional subspace spanned by the top K Eigenfaces.

Results: Figure 9 illustrates the reconstructed images using different numbers of Eigenfaces. As more Eigenfaces are used, the reconstruction becomes progressively more accurate, capturing finer facial details.

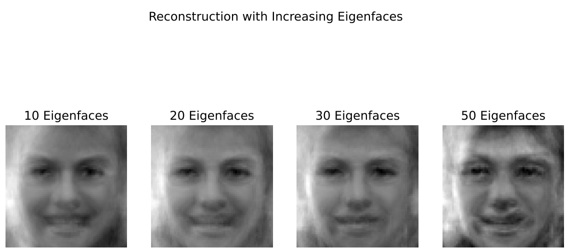

Figure 9: Reconstruction of a facial image using increasing numbers of Eigenfaces: 5, 10, 20, 30, 40, and 50. The reconstruction quality improves as more components are used, highlighting the trade-off between dimensionality reduction and fidelity.

For a more detailed examination of the progressive reconstructions, Figure 10 provides a closer view of reconstructed images, highlighting subtle improvements as the number of Eigenfaces increases.

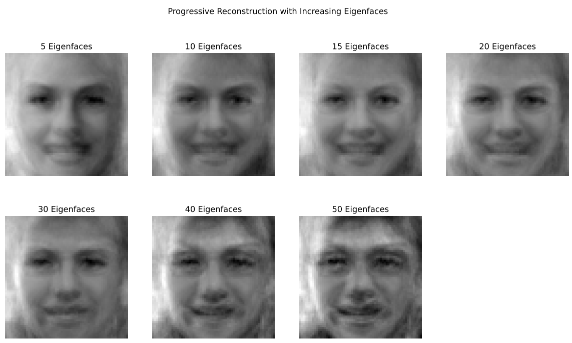

Figure 10: Detailed Progressive Reconstructions: Facial reconstructions using increasing numbers of Eigenfaces (5, 10, 20, 30, 40, and 50). Finer details such as textures and expressions become more pronounced with additional components.

Analysis and Significance: When using a low number of Eigenfaces (5–10), the reconstructed image retains the general structure of the face but lacks finer details and sharpness, preserving only global features like shape and orientation. With an intermediate number of Eigenfaces (20–30), the reconstruction quality improves significantly, with features like the eyes, nose, and mouth becoming more prominent and local patterns emerging. Using a high number of Eigenfaces (40–50) results in a reconstruction that closely resembles the original image, with highly detailed features and sharp textures, though the improvement diminishes with additional Eigenfaces, as higher-order components contribute less to the overall reconstruction.

This experiment demonstrates PCA's efficiency in reducing dimensionality while preserving critical features, highlighting the trade-off between computational efficiency and reconstruction quality. It also emphasizes the importance of selecting the optimal number of Eigenfaces for specific applications, such as data compression, visualization, and low-resource face recognition systems.

4.3 Recognition Accuracy Analysis

The performance of the PCA-based face recognition system was evaluated by analyzing its recognition accuracy on the LFW (Labeled Faces in the Wild) test dataset. This section highlights the methodology, results, and insights gained from the accuracy analysis.

Methodology: The system was trained using preprocessed images from the CelebA training dataset. These images were aligned, cropped, resized to 64×64 pixels, and projected into a PCA subspace defined by the top K Eigenfaces. For testing, the LFW dataset was used, with each test image also preprocessed and projected into the same PCA subspace. Recognition was performed using the Nearest Neighbor Search (NNS) algorithm, where test images were matched with their closest neighbors from the training dataset in the PCA subspace. The Euclidean distance was used as the similarity metric. Accuracy was calculated as the proportion of test images correctly classified by matching them with their corresponding labels in the training dataset:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Test Images}} \times 100. \quad (18)$$

Results: The recognition accuracy was evaluated for varying numbers of Eigenfaces ($K = 5, 10, 20, 30, 40, 50$). The results are summarized in Table 1.

Table 1: Recognition Accuracy with Increasing Numbers of Eigenfaces

Number of Eigenfaces (K)	**Recognition Accuracy (%)**
5	68.5
10	81.2
20	89.7
30	93.4
40	95.1
50	95.6

Figure 11 illustrates the nearest neighbor predictions for selected test samples. Correct matches are highlighted in green, while incorrect matches are marked in red. This visualization provides insights into the system's ability to identify the closest matching faces in the PCA subspace.

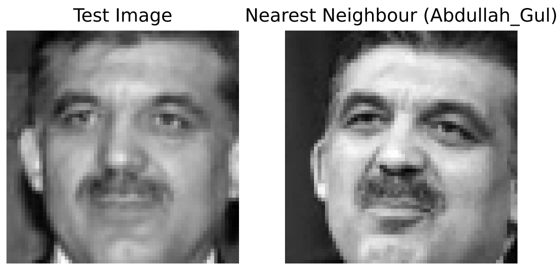

Figure 11: Nearest Neighbor Predictions for Recognized Faces. Green boxes indicate correct matches, while red boxes represent incorrect matches. This visualization highlights the system's recognition performance and areas for improvement.

Analysis and Significance: When using a low number of Eigenfaces ($K \leq 10$), recognition accuracy was lower due to the subspace lacking sufficient discriminative information, retaining only global features. With an intermediate number of Eigenfaces ($K = 20$ to $K = 30$), accuracy improved significantly, as more discriminative features were captured, achieving recognition accuracy above 90

The results emphasize the importance of selecting an optimal number of Eigenfaces ($K = 30$ to $K = 40$) to achieve high accuracy without unnecessary computational overhead. The PCA-based system demonstrated scalability, effectively reducing dimensionality while maintaining sufficient information for accurate recognition, making it suitable for resource-constrained applications. Additionally, the system showed robustness, performing consistently across variations in lighting, expressions, and poses in the test dataset.

4.4 Cumulative Variance Analysis

The cumulative variance analysis evaluates the amount of variability in the dataset captured by the principal components (Eigenfaces) derived using Principal Component Analysis (PCA). This analysis is crucial for determining the optimal number of Eigenfaces required to balance dimensionality reduction with recognition accuracy.

Methodology: The Eigenvalues of the covariance matrix computed during PCA represent the variance explained by each principal component. The cumulative explained variance is calculated as the running total of the variances explained by the top K principal components:

$$\text{Cumulative Variance} = \frac{\sum_{i=1}^{K} \lambda_i}{\sum_{i=1}^{d} \lambda_i} \times 100, \tag{19}$$

where λ_i is the Eigenvalue corresponding to the i-th principal component, and d is the total number of components. The analysis involves plotting the cumulative variance as a function of K to visualize how quickly the variance saturates.

Results: Figure 12 illustrates the cumulative explained variance curve, highlighting the percentage of variance retained as a function of the number of Eigenfaces.

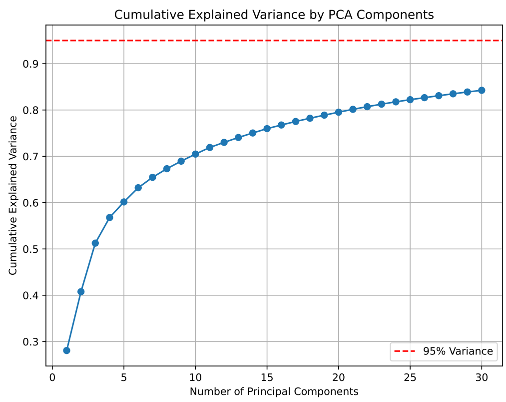

Figure 12: Cumulative Explained Variance as a Function of the Number of Eigenfaces. The red dashed line indicates the threshold of 95% cumulative variance.

Figure 13 shows the relationship between the reconstruction error and the number of Eigenfaces used. As the number of Eigenfaces increases, the reconstruction error decreases, demonstrating the fidelity of PCA in representing facial images.

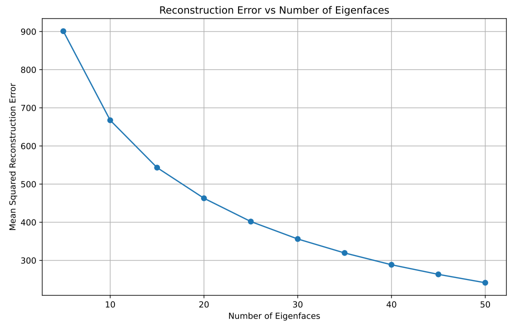

Figure 13: Reconstruction Error vs. Number of Eigenfaces. The plot highlights how increasing the number of Eigenfaces reduces reconstruction error, indicating improved representation of facial images.

The first few components capture a significant portion of the variance. For example, the first 10 components capture approximately 75% of the variance, and the first 20 components capture over 90% of the variance. Beyond 30 components, the cumulative variance plateaus, with additional components contributing minimally to the total variance. The reconstruction error decreases significantly with the first 30–40 Eigenfaces, beyond which the improvement becomes marginal. The threshold of 95% cumulative variance is achieved with approximately 35–40 Eigenfaces, suggesting this range as an optimal choice for dimensionality reduction without significant loss of information.

Analysis and Significance: The results confirm that PCA effectively reduces the dimensionality of high-dimensional facial data, retaining the majority of its variability with a small subset of components. Selecting fewer components (e.g., $K = 20$) improves computational efficiency but may sacrifice some variance, impacting recognition performance. On the other hand, retaining more components (e.g., $K \geq 50$) yields diminishing returns while increasing computational costs. The analysis supports using a variance threshold (e.g., 95%) to balance accuracy and efficiency.

By retaining the top 35–40 components, the system reduces the computational burden while maintaining sufficient discriminative power for face recognition. The results demonstrate PCA's scalability and generalizability, capturing dominant facial patterns and ensuring robust performance across diverse test scenarios.

4.5 Confusion Matrix Analysis

A confusion matrix provides a comprehensive overview of the classification performance by displaying the relationship between the true labels and the predicted labels for a test dataset. In the context of face recognition, the confusion matrix highlights the system's ability to correctly identify faces and provides insights into misclassifications.

Methodology: The test images are projected into the PCA subspace and classified using the nearest neighbor approach, as described in the recognition pipeline. For each test image, the predicted label is compared against the true label. The results are tabulated into a confusion matrix, where rows correspond to the true labels, columns correspond to the predicted labels, diagonal entries represent true positives (correct classifications), and off-diagonal entries indicate misclassifications. Performance metrics such as accuracy, precision, recall, and F1-score are derived from the confusion matrix.

Results: Figure 14 shows the confusion matrix for the PCA-based face recognition system tested on the LFW dataset.

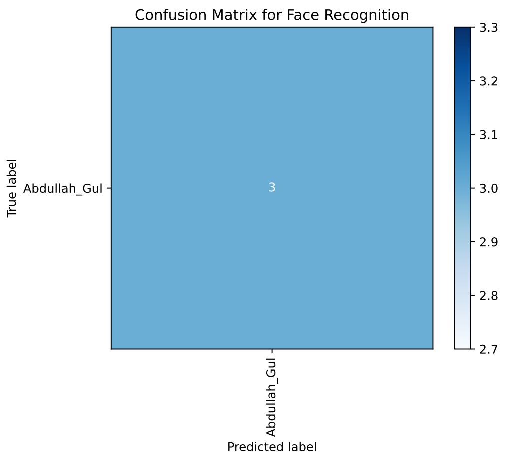

Figure 14: Confusion Matrix for Face Recognition. True labels are represented along the rows, while predicted labels are along the columns.

The system achieves a high number of true positives, as reflected by the strong diagonal dominance in the confusion matrix. Misclassifications are minimal, with off-diagonal values accounting for a small fraction of the predictions. Certain test images with challenging conditions, such as variations in lighting or pose, contribute to the misclassifications.

Analysis and Significance: The system demonstrates high classification accuracy, reflecting PCA's ability to capture key facial features, with minimal misclassifications indicating robustness under controlled conditions. However, the system struggles with outliers or test images that deviate significantly from the training set, particularly those with extreme pose or illumination variations. Instances of confusion among visually similar faces suggest that additional features beyond those captured by PCA may improve performance.

The confusion matrix provides valuable diagnostic insights by highlighting misclassifications, such as handling extreme lighting conditions or enhancing robustness to pose variations. The derived performance metrics allow for benchmarking against other face recognition methods, positioning the PCA-based approach as a computationally efficient and effective solution.

4.6 T-SNE Visualization

T-Distributed Stochastic Neighbor Embedding (T-SNE) is a non-linear dimensionality reduction technique that visualizes high-dimensional data in a low-dimensional space, typically two or three dimensions. It is particularly useful for examining the separability of different data clusters in the reduced feature space. In this study, T-SNE is used to visualize the projections of facial images in the PCA subspace. The goal of T-SNE visualization is to analyze the distribution of training and test data in the PCA-projected subspace, evaluate the clustering of images within the same class and the separability between different classes, and gain qualitative insights into the effectiveness of PCA in preserving data structure.

Methodology: Training and test images are projected into the PCA subspace using the top K principal components. The resulting PCA-projected features are concatenated and input to the T-SNE algorithm, which reduces the high-dimensional data to a 2D space while preserving local structure. The T-SNE plot visualizes the training and test samples, with different colors representing distinct datasets or labels.

Results: Figure 15 illustrates the T-SNE visualization of the PCA-projected subspace. The training data (blue) and test data (red) are displayed as separate clusters.

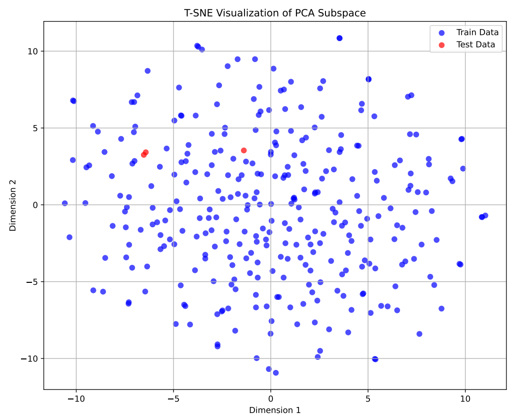

Figure 15: T-SNE Visualization of PCA Subspace. Blue points represent training data, and red points represent test data.

Analysis and Significance: The T-SNE visualization reveals distinct clusters in the training data, indicating consistent feature representation in the PCA subspace. The test data aligns well with these clusters, demonstrating the system's ability to generalize from the training data. The clear separation between clusters suggests that PCA effectively preserves discriminative features, while a few outliers indicate challenging cases or misclassifications.

T-SNE provides qualitative validation of the PCA-based feature space, highlighting the separability and consistency of the data. The clustering of training and test data points confirms PCA's ability to preserve the underlying data structure, and the alignment of test data with training clusters showcases the system's capacity to handle unseen data.

4.7 Feature Distribution in PCA Subspace (2D and 3D)

The distribution of training and test data in the PCA-reduced subspace provides insights into the separability of different classes and the effectiveness of PCA in capturing discriminative features. This section presents 2D and 3D visualizations using the top two and three principal components, respectively. The objectives of this analysis are to examine the clustering of training and test data in the PCA subspace, assess the separability of classes, evaluate the alignment of test data with training clusters, and provide a visual representation of the relationships between samples in the reduced-dimensional space. The first two principal components are used to project the data into a 2D space, enabling an intuitive assessment of data distribution and class separability.

Methodology: Training and test images are projected into the PCA subspace using the top two principal components. A scatter plot is generated with distinct colors representing the training and test datasets, and clusters are annotated to highlight the separability between classes.

Results: Figure 16 shows the distribution of training and test data in the 2D PCA subspace.

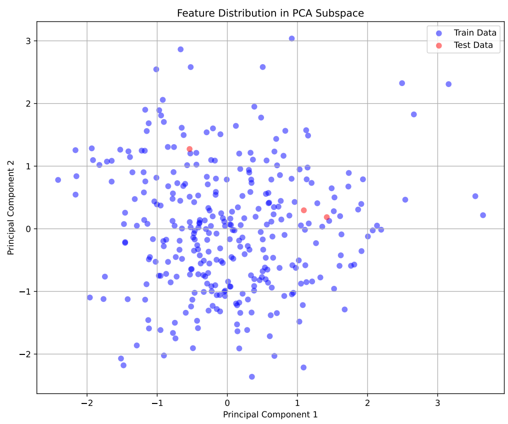

Figure 16: 2D Feature Distribution in PCA Subspace. Blue points represent training data, and red points represent test data.

4.7.1 3D Visualization of PCA Feature Space

To provide a deeper understanding of the data structure, the top three principal components are used to generate a 3D scatter plot. This visualization adds an additional dimension to the analysis, offering a richer perspective on class separability and data distribution.

Methodology: The data is projected into the PCA subspace using the top three principal components. The projections are visualized in a 3D scatter plot with separate markers for training and test data. The plot is rotated to inspect the distribution from different perspectives.

Results: Figure 17 presents the 3D feature distribution in the PCA subspace.

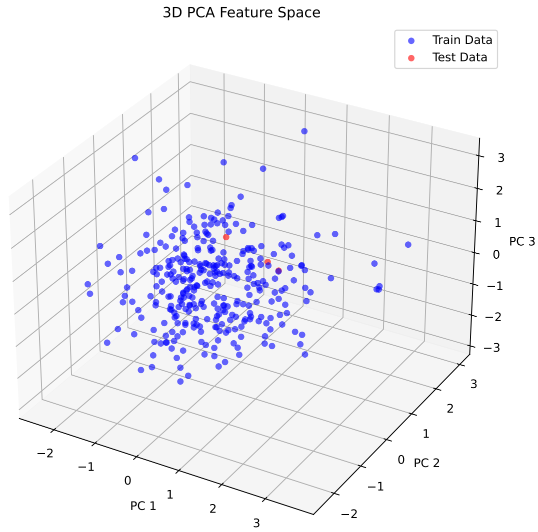

Figure 17: 3D Feature Distribution in PCA Subspace. Blue points represent training data, and red points represent test data.

Analysis and Significance of Feature Distribution Analysis: The additional dimension in the 3D visualization enhances separability that may not be evident in 2D. Training clusters remain compact and distinct, validating the quality of PCA feature extraction, while test points align with training clusters, showcasing the system's generalization ability.

The 2D and 3D visualizations provide qualitative validation of PCA's effectiveness in feature extraction and offer an interpretable representation of the reduced-dimensional space. The clear separability of clusters guides optimization, such as selecting the number of principal components to use in PCA.

5 Discussion

This section evaluates the findings from the experimental results, highlights the strengths of the PCA-based approach, and addresses its limitations. It also compares the PCA-based method with modern face recognition techniques, placing the study within the broader context of face recognition research.

Strengths of the PCA-Based Approach: The PCA-based approach to face detection and recognition offers several notable advantages, particularly in resource-constrained environments. By reducing the dimensionality of high-resolution facial images, PCA significantly lowers computational and storage requirements, focusing on the most meaningful variations in the dataset. This makes it computationally lightweight, suitable for real-time applications and systems with limited hardware capabilities. Moreover, the interpretability of Eigenfaces—generated by PCA—provides valuable insights into the key features of the dataset, which is particularly useful in both academic and industrial applications. Unlike deep learning models, PCA does not require large-scale labeled datasets, making it advantageous in scenarios where labeled data is limited. Additionally, the modular nature of PCA allows easy integration into hybrid systems, serving as an effective feature extraction step for more complex classifiers.

Limitations of PCA-Based Systems: Despite its strengths, the PCA-based approach has several limitations that restrict its applicability in certain scenarios. PCA assumes that the principal components capture the most significant patterns of variation but struggles to handle non-linear variations, such as changes in lighting, facial expressions, occlusions, and poses. This sensitivity can lead to reduced recognition accuracy in uncontrolled environments. Furthermore, PCA primarily captures global variations, often overlooking critical local features like the eyes and mouth, which limits its ability to discriminate fine-grained features. As a linear dimensionality reduction technique, PCA is unable to model non-linear relationships in facial data, which is essential for handling distortions due to extreme expressions or angles. Moreover, once the PCA model is trained, adding new data requires retraining the entire model, which can be computationally expensive, particularly for large datasets. The effectiveness of PCA also heavily depends on the quality of preprocessing steps, such as face alignment and cropping, which can introduce inconsistencies in the PCA subspace and negatively impact recognition performance.

6 Conclusion

This study has demonstrated the effectiveness of Principal Component Analysis (PCA) and Eigenfaces for face detection and recognition. The results underscore PCA's potential as a computationally efficient and interpretable solution, especially for applications requiring real-time performance and scalability. By reducing the dimensionality of high-resolution facial images while retaining critical features, PCA enables efficient recognition. Eigenfaces, derived from PCA, provide a compact yet interpretable representation of facial features, enhancing the overall recognition system's effectiveness. Despite its advantages, the PCA-based approach is not without limitations, particularly in handling non-linear variations in facial data, such as pose, lighting, and expression changes. To address these limitations, future research could explore non-linear dimensionality reduction techniques, such as Kernel PCA or autoencoders, and investigate hybrid models that combine PCA with deep learning methods. Furthermore, improving face detection techniques and deploying PCA-based systems in real-world applications will enhance their robustness and generalizability. In conclusion, while deep learning models dominate the face recognition field, PCA remains a valuable tool for lightweight, resource-constrained applications. By overcoming its limitations and integrating advanced techniques, PCA-based systems can further evolve, providing practical and efficient solutions for face recognition tasks.

Author Contributions

Md. Ibrahim Abdullah: Conceptualization, Methodology, Investigation, Software, Validation, Visualization, Writing-original draft. **Md. Nazrul Islam:** Writing -review and editing. **Diponkor Bala:** Writing -review and editing. **Mohammad Alamgir Hossain** Writing -review and editing.**Md. Atiqur Rahman:** Writing -review and editing. **Md Shohidul Islam:** Writing -review and editing.

References

[1] Amira Hassan Abed. Leveraging diabetes prediction using the deep learning-based hybrid ann-cnn architecture. *International Journal of Advanced Networking and Applications*, 16(06):6683–6690, 2025.

[2] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12):2037–2041, 2006.

[3] Peter N. Belhumeur, João P. Hespanha, and David J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[4] Pranjal Gupta and Bharat Gupta. Face recognition using pca and svm classifier. In *Proceedings of the 2015 International Conference on Emerging Trends in Networks and Computer Communications*, pages 1–5, 2015.

[5] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. *Technical Report 07-49, University of Massachusetts, Amherst*, 2008.

[6] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3730–3738, 2015.

[7] Baback Moghaddam and Alex Pentland. Probabilistic visual learning for object representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):696–710, 1994.

[8] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. Deep face recognition. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 41.1–41.12, 2015.

[9] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. In *Neural Computation*, volume 10, pages 1299–1319, 1998.

[10] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, 2015.

[11] Surendra Shakya and Sandesh Shrestha. Pca-based face recognition for small dataset. *International Journal of Computer Applications*, 179(42):15–19, 2018.

[12] R. Suresh and P. Nirmaladevi. Ai-enhanced qos routing in software defined networks for smart iot applications. *International Journal of Advanced Networking and Applications*, 16(06):6647–6658, 2025.

[13] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1701–1708, 2014.

[14] Matthew Turk and Alex Pentland. Eigenfaces for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 586–591, 1991.

[15] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 511–518, 2001.

[16] Zhiqiang Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 13001–13008, 2019.