International Student Research Review
Volume 2 Issue 1 May 2025
DOI: https://doi.org/10.53378/isrr.164

An enhancement of the Eigenface algorithm using weber local descriptor applied in attendance management system

[1]Amyr Edmar Francisco, [2]Angelo Lance Seraspi, [3]Jamillah Guialil, & [4]Khatalyn Mata

Abstract

This study presents an improved face recognition system tackling the Eigenface algorithm's limitations regarding lighting variance, class separability, and classification. The proposed method incorporates Weber Local Descriptor (WLD) for illumination normalization during training and recognition. Further improvements include Kernel Principal Component Analysis (KPCA) for non-linear feature transformation, Linear Discriminant Analysis (LDA) to maximize class separability, and Ridge classification for noise-resistant recognition, replacing Euclidean distance. Testing on the extended Yale B dataset showed a significant accuracy increase from 5.63% (original Eigenface) to 99.83% (enhanced Eigenface). Evaluation on a custom dataset simulating real-world conditions (varying light, expressions) yielded 100% accuracy across feature transformation, class separability, and classification. These results demonstrate the effectiveness of the integrated WLD, KPCA, LDA, and Ridge classification techniques in developing a robust and accurate face recognition system suitable for applications like attendance management.

Keywords: *Eigenface algorithm, image processing, facial recognition, local descriptor, Weber Local Descriptor, Kernel PCA*

Article History:

Received: *March 13, 2025*
Revised: *May 22, 2025*
Accepted: *May 25, 2025*
Published online: *May 31, 2025*

Suggested Citation:
Francisco, A.E., Seraspi, A.L., Guialil, J. & Mata, K. (2025). An enhancement of the Eigenface algorithm using weber local descriptor applied in attendance management system. *International Student Research Review*, 2(1), 139-173. https://doi.org/10.53378/isrr.164

About the authors:
[1]Corresponding author. 4th year Computer Science student at Pamantasan ng Lungsod ng Maynila and former GDSC-PLM CTO with interests in data science and machine learning. Email: amyrfrancisco11@gmail.com
[2]4th year Computer Science student at Pamantasan ng Lungsod ng Maynila focused on full-stack development and AI.
[3]Professor in Computer Science at Pamantasan ng Lungsod ng Maynila.
[4]Dean of College of Information Systems and Technology Management at Pamantasan ng Lungsod ng Maynila.

1. Introduction

The process of recognizing or verifying someone's identity using their face is known as face recognition. Although the terms face detection and face recognition are often used interchangeably, they refer to two different technologies. The primary difference is that the former simply detects the presence of a face, while the latter recognizes a person's face from a collection of known faces. Despite being distinct, both technologies are often used together in systems that require facial recognition through a camera.

Face recognition has a wide range of real-world applications. It is commonly used in authentication, providing a secure and convenient way to unlock smartphones and tablets, as well as to verify user identities for secure transactions in the financial sector. It is also employed in attendance management systems, where it can accurately track and record attendance in schools, offices, and other organizations. Law enforcement and crime prevention agencies use face recognition to identify suspects and prevent crimes, aiding in investigations and enhancing public safety. These examples demonstrate the adaptability and value of face recognition technology in improving efficiency, safety, and user experience across various industries and applications.

One common problem encountered in early face recognition algorithms was how to represent the human face in a way that captures the unique features identifying an individual. Early approaches used basic face models, such as extracting features from edge images and comparing simple distances and ratios between these features. However, some facial characteristics are not intuitive to human interpretation. Turk and Pentland (1991), building on the work of Sirovich and Kirby (1987), developed an algorithm known as the Eigenface Approach—or simply the Eigenface Algorithm—in face recognition.

The Eigenface Algorithm was one of the first successful methods for real-time face recognition. The core idea is to represent a face using a small set of weights and standard features called eigenfaces. These weights differ for each individual. To recognize a new face, the algorithm calculates its weights and compares them to those of known individuals to determine whether the face is recognized or not, based on the closest match in weight patterns. To identify the eigenfaces (or the "face space"), Principal Component Analysis (PCA) is used. This method goes beyond our intuitive understanding of facial features by identifying the most distinctive characteristics that best describe individuals within a known population, rather than relying solely on familiar features like the eyes, nose, or mouth.

2. Literature Review

2.1. The concept of the Eigenface algorithm

The ability of human beings to recognize faces is remarkable. However, creating a computational model to recognize faces is quite difficult, primarily because faces are complex and multidimensional. The use of basic face models and feature descriptions, such as identifying features from an edge image and comparing simple distances and ratios, limited early attempts to enable computers to recognize faces. These attempts were based on the premise that a face is simply a combination of its individual features (Turk & Pentland, 1991).

Sirovich and Kirby (1987) proposed a method for characterizing a face image by keeping only a small set of weights for each face in a population of face images, as well as a small collection of standard pictures known as eigenpictures. In principle, this set of weights will differ between faces, and rather than storing all the pixels in an image, a face image can be reconstructed using a weighted sum of all the eigenpictures. This essentially reduces the dimensions of the features. Turk and Pentland (1991) expanded on this idea, arguing that if a number of faces can be reconstructed by weighted sums of all the eigenpictures, then any particular face can be recognized by comparing the weights required to closely reconstruct them to the weights associated with known faces. This approach to face recognition is also known as the Eigenface Approach or Eigenface Algorithm.

The Eigenface Algorithm can be divided into two components: training and recognition. During the training phase, gather a set of face images (the training set). These images are subjected to Principal Component Analysis (PCA) to calculate the eigenvectors or eigenfaces, leaving only the M eigenfaces with the highest eigenvalues. These M images define the face space in which all face images will be reconstructed. Finally, calculate the weights of each face by projecting it onto the “face space”. After initializing the system, the following steps are used for recognizing new faces. First, calculate the weights of the face image by projecting it into the "face space" or each of the eigenfaces calculated during the training phase. Then, check to see if the image is close enough to the "face space" to be considered a face. If it is a face, compare its weights to those of known faces and determine whether it is a known person or an unknown by finding the weight pattern of a known face that is closest to the weight pattern of the new face.

2.2. Limitations of the Algorithm

Issues with lighting levels. The Eigenface algorithm, despite its effectiveness in facial recognition, faces several challenges. One significant issue is the algorithm's sensitivity to variations in lighting conditions, which can affect the accuracy of facial recognition systems (Javed Mehedi Shamrat et al., 2022; Annubaha et al., 2022). Additionally, the Eigenface algorithm can encounter difficulties when faced with environmental factors like varying light intensities and face rotations, leading to decreased accuracy in facial recognition (Fei et al., 2018). Maharani Raharja et al. (2021) found that "if the light intensity on the face image object is above 1700 lux then the face image object can be recognized properly, but if the intensity ranges from 0 to 1700 lux then the face image is not recognized at all or is wrong in detecting faces due to lack of lighting." Fahmy et. al (2003) also stated that "the quality of the acquired facial image, which in turn affects the recognition/matching rate, decreases significantly when the light direction is outside the range of 60 to 120 degrees with 90 degrees as the exact frontal light angle. This is due to the fact that when the light is closer to either the right or left profile of the face, the luminance distribution across the face is unequal, which affects the feature detection". Eigenfaces are sensitive to lighting variations as they rely on global intensity patterns. Under varying lighting conditions, the eigenfaces may not capture the necessary discriminative information, leading to reduced accuracy. (Dorbi & Joshi, 2023).

Large datasets. The eigenface algorithm also struggles with large datasets. If the size of the dataset is large, it creates a huge covariance matrix that may not be computationally feasible, and might affect the performance of the algorithm in recognizing faces correctly (ElSayed et al., 2012). The computational requirements of these approaches are greatly related to the dimensionality of the original data and the number of training samples. When the face database becomes larger, the time for training and the memory requirement will significantly increase (Er et al., 2005). A feasible solution was made by dividing a large dataset into a bunch of small subsets that satisfied the assumption of conventional approaches (Park et al., 2017). Kekre et. al (2014) proposed a new approach of classification where two sets of eigenfaces are used, one for each gender. After a given test image is reconstructed with the Eigen coordinate systems of each gender, the lowest Mean Squared Error (MSE) between given image and reconstructed image, indicates the output class or gender for that image.

Inefficient computation of eigenvectors and eigenvalues. The methods for visual learning that compute a space of eigenvectors by Principal Component Analysis (PCA) traditionally require a batch computation step. Since this leads to potential problems when dealing with large sets of images, several incremental methods for the computation of the eigenvectors have been introduced. Another problem is that, in order to update the subspace of eigenvectors with another image, the whole decomposition must be recomputed from scratch (Artac et al., 2002).

Incremental Principal Component Analysis (IPCA) is another kind of feature extraction algorithm that is used for reducing the dimensionality of input feature vectors. IPCA is recommended to be used in place of conventional PCA when the size of the dataset to be analyzed is too large to fit in memory (Rehman et al., 2020). Given that retraining data in a frequent manner and increasing training data is to be expected. Recomputing an already existing eigenvector in order to add training data into it would be inefficient and will consume more memory. A research by Hallgren and Northrop (2018) supports this by stating "Incremental algorithms, where a solution is updated for additional data examples, are often desirable" and "Incremental algorithms often have a lower memory footprint than their batch counterparts."

2.3. Weber Local Descriptor

The Weber Local Descriptor (WLD) constitutes a robust local image descriptor. Its conceptual basis lies in Weber's Law, a principle in psychology positing that the just noticeable difference in a stimulus exhibits proportionality to the magnitude of the initial stimulus. WLD is engineered to extract locally significant patterns from an image (Jie Chen et al., 2010).

WLD is robust to illumination changes and noise. The computation of differential excitation and orientation relies on ratios, which helps to mitigate the effects of both multiplicative noise and variations in brightness and contrast (Jie Chen et al., 2010). In contrast to SIFT, which is often computed over a larger neighborhood (e.g., 16x16), the Weber Local Descriptor (WLD) is typically computed over a smaller neighborhood (e.g., 3x3), thereby enabling the capture of more granular local details (Jie Chen et al., 2010). The Local Binary Pattern (LBP) exhibits computational simplicity and greater speed than WLD in certain aspects and demonstrates tolerance to monotonic variations in illumination (Dahmouni et al., 2024). However, LBP generally displays heightened sensitivity to noise and localized variations in lighting. Furthermore, it can generate feature vectors of potentially high dimensionality (Singh & Chhabra, 2018). While the Histogram of Oriented Gradients (HOG) excels at capturing information pertaining to edges and shapes, demonstrating efficacy in facial expression recognition and face recognition, it can exhibit higher computational demands compared to WLD. Furthermore, the performance of HOG is notably contingent upon careful parameter tuning (Carcagnì et al., 2015).

The application of the WLD resulted in a 5.27% increase in recognition accuracy on the Yale database compared to the application of Local Ternary Patterns (LTP). Superior performance was also observed on the ORL database. Empirical findings indicated that WLD attained an accuracy of 99.25% on the ORL database and 96.97% on the Yale database, outperforming LBP, which achieved accuracies of 96% and 90.30% on the same datasets, respectively (Gong et al., 2011).

A proposed algorithm by Kong and Zhang (2015) integrating the WLD and Laplacian-of-Gaussian (LoG) to achieve illumination invariance, followed by Complete Linear Discriminant Analysis (CLDA) for feature extraction (WLD+LoG+CLDA), demonstrated the most effective recognition performance on the Yale and Extended Yale Database B. The average recognition rate attained on the Yale database was 96%. On the Extended Yale Database B, Kong and Zhang (2015) WLD+LoG+CLDA algorithm achieved a recognition rate of 99.74%, surpassing the performance of methods employing FastPCA, Fisherface, and CLDA independently, as well as LN+CLDA (Local Normalization + CLDA).

The WLD demonstrates considerable efficacy as a feature extraction technique for face recognition, frequently exhibiting superior performance compared to conventional global methods such as PCA and ICA, as well as the local descriptor LBP, particularly under variable conditions including illumination changes and noise. An evaluation was conducted comparing the WLD, LBP, and Scale-Invariant Feature Transform (SIFT) on the Brodatz texture dataset corrupted by additive white Gaussian noise. The results indicate that while the performance of all three descriptors degrades significantly with noise levels exceeding 5%, WLD demonstrates comparatively greater robustness than LBP and SIFT under these challenging conditions (Jie Chen et al., 2010).

2.4. Kernel PCA

Kernel Principal Component Analysis (KPCA) is a non-linear extension of the standard PCA (Kim et al., 2002). PCA, while efficacious for linear dimensionality reduction, encounters limitations when applied to datasets characterized by non-linear relationships. KPCA addresses this constraint through the implicit mapping of input data into a higher-dimensional feature space by employing a kernel function, subsequently performing linear PCA within that transformed feature space (Kim et al., 2002; Wang & Zhang, 2010; Peter et al., 2018).

KPCA is particularly helpful with non-linear patterns in facial images because it extends the capabilities of the linear PCA method to handle more complex data structures. Peter et al. (2015) stated that the fundamental concept of KPCA involves the utilization of a kernel function to effect an implicit mapping of input facial images into a higher-dimensional feature space. Facial images frequently exhibit non-linear configurations arising from variations in facial expressions, pose, and illumination. The employment of non-linear kernel functions, such as polynomial kernels or Gaussian kernels (Peter et al., 2018), enables KPCA to capture higher-order correlations among the pixels of a facial image (Kim et al., 2002). By introducing the kernel function, which avoids the calculation inconvenience of inner product in high dimensional feature space. Some kernel functions such as the polynomial kernel, Gaussian kernel and sigmoid kernel have been commonly used in many practical applications of kernel methods (Kim et al., 2002).

KPCA has several specific applications within the domain of facial recognition, primarily centred around enhancing recognition performance by addressing the limitations of linear methods like traditional PCA (Peter et al., 2018). Facial images frequently exhibit nonlinear variations attributable to alterations in facial expressions (e.g., smiling, frowning), illumination conditions, pose variations, and other appearance deformations (Zhao et al., 2012). KPCA, through the mapping of input facial images into a higher-dimensional feature space via a kernel function, demonstrates a superior capacity to capture these intricate, non-linear relationships in comparison to linear PCA (Peter et al., 2018). This process yields a feature set that is more appropriate for categorization purposes (Zhao et al., 2012).

Despite operating within a high-dimensional space, KPCA continues to perform dimensionality reduction through the extraction of principal components, defined as the directions of maximum variance within that space (Kim et al., 2002). This process contributes to a reduction in computational complexity and storage demands relative to operating with the original high-dimensional image data (Ebied, 2012). The features extracted by KPCA are often used as input for various classification algorithms to perform the actual face recognition. KPCA serves as an effective preprocessing step that transforms the raw image data into a more discriminative feature representation (Prajapati & Navamani, 2023).

KPCA is considered a non-linear extension of PCA (Kim et al., 2002). It uses a kernel function to map the input data into a higher dimensional feature space before performing linear PCA in that space, whereas PCA is a linear approach that determines projection directions of maximum variance in the original data (Peter et al., 2018). It can demonstrate superior performance compared to standard PCA in the context of face recognition, yielding elevated recognition rates and diminished error rates, particularly when addressing variations such as facial expressions (Peter et al., 2018). The efficacy of PCA may be limited when confronted with intricate non-linear structures (Zhou et al., 2007). Furthermore, KPCA is recognized for its capacity to extract feature sets that are more amenable to categorization than those derived from classical PCA (Wang & Zhang, 2010).

Both PCA and LDA are predicated on linear proximity features, which may exhibit reduced efficacy when applied to facial images characterized by non-linear attributes (Liliana & Setiawan, 2019). KPCA addresses non-linearities through the application of the kernel trick. This technique enables the algorithm to capture intricate, non-linear relationships among pixel values by implicitly considering high-order correlations, thereby circumventing the computational demands associated with explicit operations within the high-dimensional space wherein these correlations manifest as linear (Prajapati & Navamani, 2023).

In an empirical investigation conducted by Peter et al. (2018), the application of KPCA yielded a high recognition rate; however, it also resulted in instances of false rejection. This observation suggests an inherent limitation in achieving perfect classification across all instances, even within the training dataset. Analogous to PCA, standard KPCA operates primarily as an unsupervised dimensionality reduction methodology with the objective of maximizing variance. Consequently, it may not inherently prioritize the maximization of separability between distinct classes (individuals) within facial recognition applications. This characteristic can result in the extraction of features that are not optimally discriminative between different identities (Liu et al., 2006).

The efficacy of KPCA exhibits a significant dependency on the selection of the kernel function and the specification of its corresponding parameters. The identification of an optimal kernel (e.g., polynomial, Gaussian radial basis function, sigmoid) and the subsequent calibration of its parameters (e.g., degree 'd' for the polynomial kernel, bandwidth 'σ' for the Gaussian kernel) represent a crucial and frequently complex undertaking (Liu et al., 2016). Suboptimal parameter selection can present issues such as an ill-conditioned kernel matrix or the failure of the KPCA transformation model. Furthermore, it may contribute to overfitting or underfitting the data.

Different kernel functions possess disparate characteristics and exhibit varying degrees of suitability for diverse categories of non-linear problems. Consequently, a singular kernel may not demonstrate universal efficacy across the spectrum of variations inherent in facial data, including those arising from differences in expression, illumination conditions, and pose (Liu et al., 2016).

2.5. Nyström Method

The Nyström method represents a widely adopted and versatile technique for deriving low-rank approximations of kernel matrices, thereby facilitating enhanced scalability of kernel methods when applied to extensive datasets. The operation of conducting the eigendecomposition on this (nn) matrix incurs a computational complexity of (O(n3)) (Shen et al., 2023). Furthermore, the storage of the complete kernel matrix necessitates a memory complexity of (O(n2)) (Sterge & Sriperumbudur, 2022). These substantial computational and memory requirements render exact KPCA computationally intensive and infeasible for large-scale datasets. The reduction in computational cost enables the application of KPCA to larger datasets for which the exact method is impractical (Sterge & Sriperumbudur, 2022). The Nyström method is specifically recognized as one of the most prevalent techniques for enhancing the scalability of kernel methods (Hallgren, 2022).

Nyström approximate KPCA (NY-EKPCA) shows that it can match the statistical performance of exact KPCA with less computational complexity, provided the number of subsamples is large enough and the number of eigenfunctions used in the reconstruction is not too large (Sterge & Sriperumbudur, 2022). Random Fourier Features (RFF) is an approximation technique that directly approximates the kernel function through the construction of an explicit feature map derived from random sampling of the Fourier transform associated with the kernel's distribution. In contrast, the Nyström method approximates the kernel matrix using a low-rank matrix, commonly formulated by sampling a subset of the training instances, referred to as landmark points (May, 2018).

2.6. Ridge Classifier

Ridge regression is a regularized linear regression technique employed to mitigate issues inherent in ordinary least squares (OLS) regression, notably in scenarios involving multicollinearity or high-dimensional datasets (Hastie, 2020).

Standard OLS regression seeks to minimize the sum of squared errors between the predicted and observed values. However, in the presence of high correlation among predictor variables (multicollinearity), OLS estimates can exhibit instability and high variance (An et al., 2007). Ridge regression counters this by adding an $\ell 2$ penalty term to the OLS cost function (Hastie, 2020). This procedure aids in managing the bias-variance trade-off, stabilizing coefficient estimates when multicollinearity or limited data are present, and providing a unique solution in high-dimensional contexts where OLS is underdetermined.

Cross-validation enables the testing of models using the entirety of the training set through iterative resampling, thereby maximizing the total number of data points employed for evaluation and potentially mitigating the risk of overfitting (Rao et al., 2008).

5-fold cross-validation exhibits a reduced susceptibility to the pronounced negative bias observed with leave-one-out (LOO) cross-validation for the c-statistic, particularly when employing model estimators that apply shrinkage to estimated probabilities, such as ridge regression (Geroldinger et al., 2023).

V-fold cross-validation, including 5-fold, generally has a smaller computational cost compared to methods like leave-one-out cross-validation (where $V = n$) (Arlot & Lerasle, 2015). They have also observed that performance often increases substantially when transitioning from 2-fold to 5-fold (or 10-fold) cross-validation, with progressively smaller gains for larger values of (V), indicating that 5-fold cross-validation presents a favorable balance.

5-fold cross-validation appears to be a robust and generally recommended technique for the internal validation of ridge classifiers, offering a favorable balance of bias reduction, variance control (particularly with repetitions), and computational efficiency (Geroldinger et al., 2023).

2.7. Eigenface and Other Techniques

The eigenface algorithm addresses varying face poses by utilizing techniques such as feature extraction, background removal, image registration, and dimensionality reduction. It extracts Eigen features and Histogram of Oriented Gradient (HOG) features from faces in different poses (Rosnelly et al., 2020), removes background using masking, registers images through manual landmark detection and affine transformation, and applies log-polar transformation for scale and rotation changes (Ranganatha & Gowramma, 2022). Despite these challenges, combining Eigenface with other techniques like Support Vector Machines (SVM) can enhance accuracy levels, as seen in studies achieving up to 84.29% correct identification rates (Maw et al., 2020).

Eye alignment, also referred to as geometric normalization, constitutes a critical preprocessing stage in numerous face recognition systems. This procedure ensures the consistent spatial localization of facial features, such as the eyes, across disparate images (Dutta et al., 2015).

Li et al. (2010) further underscore the critical role of face alignment in face recognition systems, observing that the majority of approaches exhibit significant sensitivity to facial pose and scale, and that inaccurately aligned faces can substantially diminish recognition accuracy. Onaran et al. (2024) observed that misalignment consistently diminishes face image quality. While image quality does not directly equate to recognition accuracy, the sensitivity of these quality assessment methods to misalignment indirectly underscores the importance of precise alignment for achieving optimal outcomes in face recognition tasks.

3. Methodology

3.1. Existing Eigenface Algorithm

Training Phase

Step 1: Load the face images dataset.

Step 2: Resize the images to be uniform in resolution $w \times h$ where w is the width and h is the height of the image.

Step 3: Flatten each image into a tall vector of size wh.

Step 4: Get the mean face by averaging the pixel values of all the images in the dataset.

Step 5: Normalize each image in the dataset by subtracting the mean face to each image in the dataset.

Step 6: Find the eigenvalues and eigenvectors by performing Principal Component Analysis (PCA) on the dataset.

Step 7: Keep only the m eigenvectors (eigenfaces) where $m << n$.

Step 8: For each image in the dataset, calculate their weight vector of size m by projecting them to each of the m eigenfaces.

Step 9: For each image belonging to the same face class (i.e. same person) in the dataset, average the weights computed.

Step 10: Set a threshold t for maximum allowable distance from each face class. Distances greater than t will be classified as 'unknown'.

Recognition Phase

Step 11: Load the unknown image.

Step 12: Resize the images to resolution $w \times h$.

Step 13: Flatten each image into a tall vector of size wh.

Step 14: Normalize the unknown image by subtracting it to the mean face.

Step 15: Project the unknown image to each eigenfaces to get the m-dimensional weight vector.

Step 16: Get the Euclidean distance of the weights of the unknown image and the averaged weights of each face class.

Step 17: Get the minimum distance and its index.

Step 18: If the minimum distance exceeds the maximum allowable distance for all face classes, classify the unknown image as 'unknown'. Otherwise, classify the image as belonging to the face class corresponding to the index.

3.2. Proposed Eigenface Algorithm

Training Phase

Step 1: Load the face images dataset.

Step 2: Resize the images to be uniform in resolution $w \times h$ where w is the width and h is the height of the image.

Step 3: Get the illumination invariant image representations using the Weber Local Descriptor.

Step 4: Flatten each image into a tall vector of size wh.

Step 5: Get the mean face by averaging the pixel values of all the images in the dataset.

Step 6: Normalize each image in the dataset by subtracting the mean face to each image in the dataset.

Step 7: Approximate the kernel matrix via the Nyström method for Kernel Principal Component Analysis (KPCA) with a cosine kernel, then reduce dimensionality by retaining the top m principal components from eigendecomposition, where $m << n$, with n representing the total image count.

Step 8: Project the images to the m principal components in the kernel space reducing the dimension from wh to m.

Step 9: Use Linear Discriminant Analysis (LDA) to the m features obtained from KPCA, further reducing the dimensionality to d, where $d \leq c - 1$, with c representing the total number of face classes in the dataset.

Step 10: Train a CalibratedClassifierCV with RidgeClassifier to the d features from LDA using 5-fold cross validation and alpha α to avoid overfitting.

Step 11: Define a threshold t $(0 \leq t \leq 1)$ to reject faces outside the training data, classifying outputs from the Ridge Classifier with probabilities less than t as "unknown".

Step 12: Combine the KPCA, LDA, and Ridge Classifier into a single pipeline to streamline the feature extraction and classification process.

Recognition Phase

Step 13: Load the unknown image.

Step 14: Resize the image to $w \times h$.

Step 15: Get the illumination invariant image representations using the Weber Local Descriptor.

Step 16: Flatten the image into a tall vector of size wh.

Step 17: Normalize the image by subtracting the mean face.

Step 18: Feed the normalized image through the pipeline to generate class-specific probability outputs.

Step 19: Get the maximum probability and its index.

Step 20: If the maximum probability is below the threshold t, classify the image as 'unknown'; otherwise, assign the image to the face class corresponding to the identified index.

3.3. Implementation

Weber Local Descriptor. A common challenge for both traditional and modern face recognition algorithms is the variability in illumination across face images. Significant changes in lighting conditions negatively impact recognition performance. For instance, the Eigenface algorithm exhibits compromised accuracy under low ambient light (0-1700 lux), often resulting in incorrect identifications (Maharani Raharja et al., 2021). Furthermore, the Eigenface algorithm's accuracy decreases when training images are acquired in well-lit settings, but recognition is performed in dark environments, leading to misclassifications. This underscores the sensitivity of face recognition to inconsistencies in illumination between training and testing phases. To address the challenges posed by varying illumination conditions in recognition tasks, researchers utilized the WLD. The WLD operator characterizes local image patterns by analyzing two key components: *differential excitation* and *gradient orientation*.

Differential excitation captures the relative intensity changes between a central pixel and its surrounding neighbors. This mechanism focuses on the local variations in pixel values, making it less sensitive to uniform changes in illumination across the image. Specifically, if a constant brightness value is added to all pixels, the differences between neighboring pixels remain unaffected, thus preserving the differential excitation values. In addition to differential excitation, WLD computes the gradient orientation at each pixel. This component provides information about the local directional changes in intensity, further contributing to the descriptor's robustness against illumination variations. The computed differential excitation and gradient orientation values for all pixels within an image or a specific region are then aggregated to form a WLD histogram. This histogram serves as a compact representation of the image's local texture characteristics.

WLD is employed as a preprocessing step prior to feature extraction. By applying WLD to the input images, the influence of illumination is significantly reduced. This ensures that subsequent feature extraction methods primarily capture intrinsic facial texture information rather than variations caused by lighting conditions.

The inherent design of the WLD operator makes it resilient to common illumination variations. As mentioned earlier, a uniform change in brightness (additive constant) does not alter the differential excitation values. Furthermore, while a change in image contrast (multiplicative constant) scales the differences between neighboring pixels, this effect is normalized out during the WLD calculation due to an internal division operation (Jie Chen et al., 2010). This normalization step enhances the descriptor's invariance to contrast changes.

Figure 1 shows the original images of subject “Amyr” vs. its transformation after WLD. The texture of the face is retained even if the pictures were taken in different environments.

Figure 1

Original images of subject "Amyr" vs. its transformation after WLD

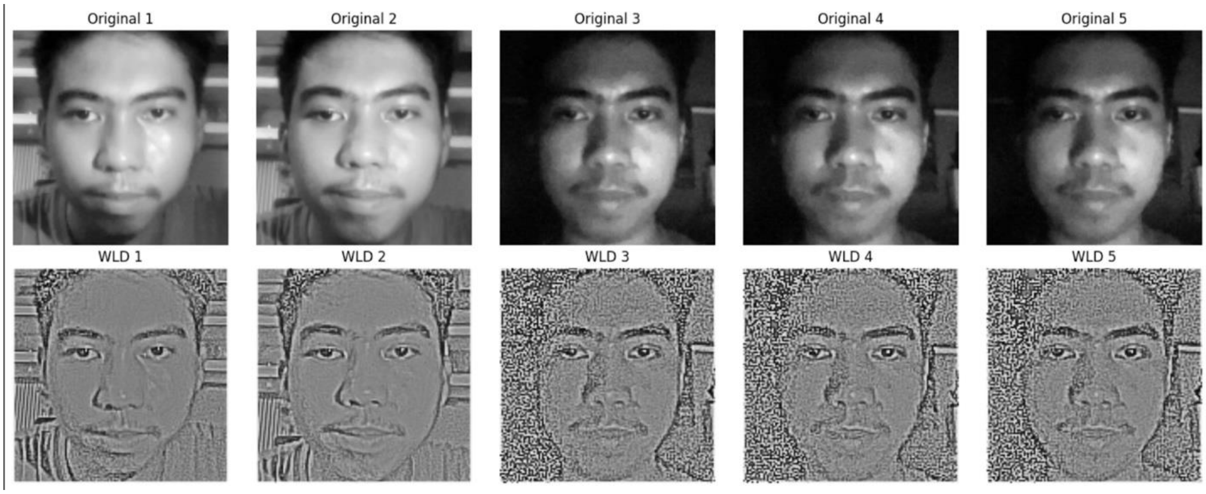
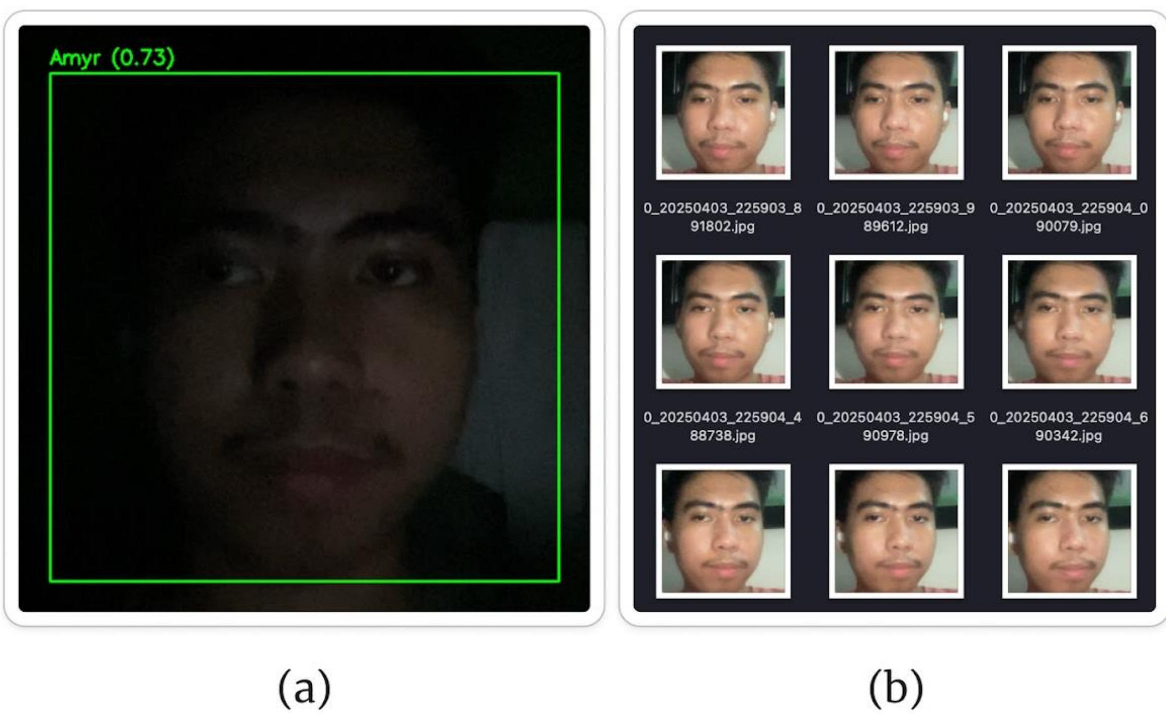


Figure 2 then shows that by using WLD as a preprocessing technique, a person with training images on a well-lit environment can still be classified correctly under dark environments.

Figure 2

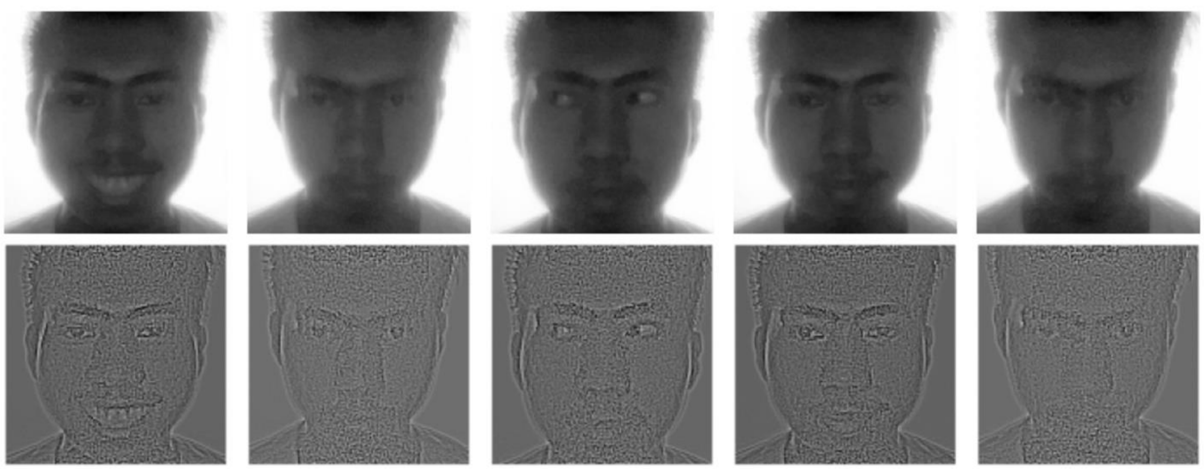
Correct classification of subject "Amyr" under low-light conditions (a) and training images of subject "Amyr" acquired under well-lit conditions (b)



Another notable limitation occurs when processing images captured under high contrast 'against-the-light' scenarios, where the background illumination significantly exceeds that of the foreground subject. Cameras possessing limited dynamic range capabilities, which is common in lower-end devices, often attempt to manage the intense background brightness by reducing the overall exposure of the image. While this prevents saturation in the background, it detrimentally darkens the foreground further, exacerbating underexposure in this region and consequently increasing the prominence of image noise. Noise artifacts can corrupt texture patterns, hindering accurate surface characterization. As illustrated in Figure 3, textures derived from these significantly darkened images exhibit considerable noise, consequently leading to degraded feature extraction and potentially impacting the overall performance of recognition.

Figure 3

'Against-the-light' images of subject "Amyr" vs. its transformation after WLD.



Nyström method for Kernel PCA and Linear Discriminant Analysis. KPCA extends the capabilities of traditional PCA by initially employing a kernel function to map the raw data into a higher-dimensional feature space. This transformation aims to achieve linear separability within the new space, a condition that might not be met in the original data. KPCA, compared to traditional PCA, stands out as a robust nonlinear feature extraction technique, demonstrating its utility as an effective preprocessing step for various classification algorithms (Prajapati & Navamani, 2023).

A kernel function, $k(x_i, x_j)$, is a function that takes two data points, x_i and x_j, as input and returns a scalar value representing their similarity or inner product in a potentially high-dimensional feature space. Common examples of kernel functions include the linear kernel, polynomial kernel, and radial basis function (RBF) kernel. In this paper, the researchers used the cosine kernel as it was the optimal kernel found after grid search with 5-fold cross validation on the olivetti face dataset.

The formula for the cosine kernel according to Mushtaq et al. (2023) is:

$$k(x_i, x_j) = \frac{x_i \cdot x_j}{\|x_i\| \cdot \|x_j\|}$$

Given a dataset with n data points $\{x_1, x_2, \ldots, x_n\}$, the Gram matrix K is an $n \times n$ matrix where the element at the i-th row and j-th column, K_{ij}, is computed as the kernel function applied to the i-th and j-th data points: $K_{ij} = k(x_i, x_j)$. The Gram matrix K plays a central role in many kernel-based algorithms including KPCA. Instead of directly working with the data points in the original input space (or their potentially high-dimensional feature space representations), KPCA operates on the Gram matrix. Using eigendecomposition, the principal components of the Gram matrix are used to reduce the dimensionality of the original face image while making it linearly separable.

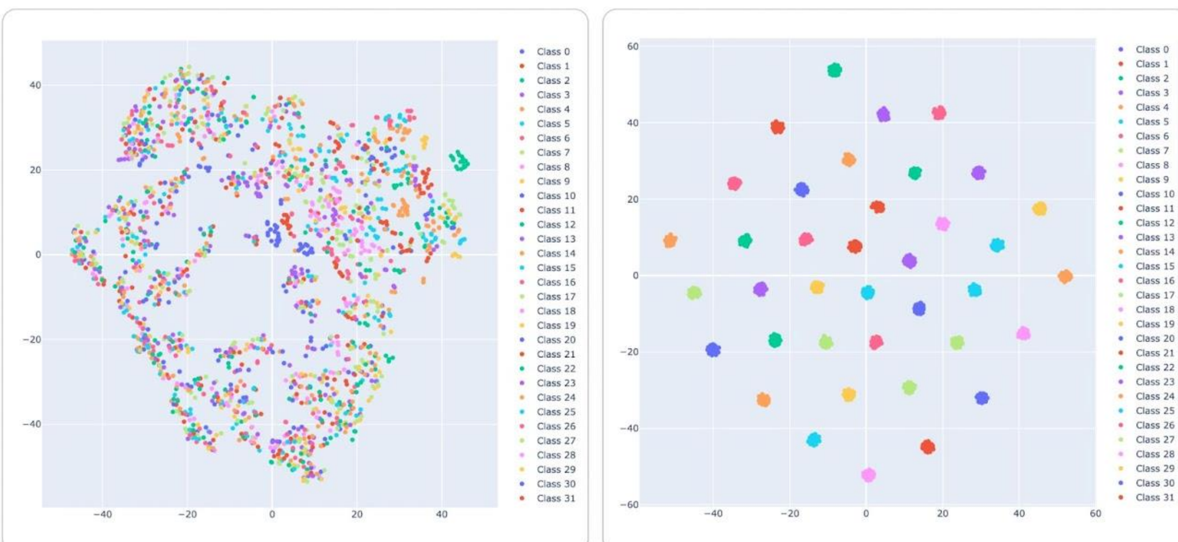
However, for large datasets (where n is large), constructing and manipulating the Gram matrix can become computationally expensive. The storage requirement for K is $O(n^2)$, and operations like eigenvalue decomposition typically have a complexity of $O(n^3)$. Nyström sampling offers a significant advantage by substantially enhancing the computational efficiency of KPCA without compromising its statistical accuracy. The Nyström method's main concept is to create a simpler, lower-rank version of the Gram matrix K. This simplified version then takes the place of the original K in kernel-based calculations, which helps speed things up computationally (Sterge et al., 2020).

To further develop effective and discriminative projected features that account for feature variability, KPCA coefficients can be projected to Linear Discriminant Analysis (LDA) projection axis (Alam et al., 2017). LDA identifies a direction (or axis) in the data's feature space that maximizes the between-class variance while minimizing the within-class variance. By doing this, it enhances the separation between different classes, making it easier to distinguish between them for classification purposes.

Figure 4(a) presents the t-distributed Stochastic Neighbor Embedding (t-SNE) projections, displaying the two-component output of t-SNE when applied to the Yale B dataset images, after an initial dimensionality reduction using PCA on the original Eigenface algorithm. Due to the large variation in illumination on the Yale B dataset, PCA fails to separate classes together, resulting in overlaps on the feature space which may lead to high misclassification even with an optimal threshold for rejecting low confidence predictions. While figure 4(b) illustrates the t-SNE projections of the same dataset following the application of KPCA and LDA on the enhanced algorithm. The resulting cluster formations exhibit enhanced clarity and separation between distinct facial classes. This improved separation suggests a greater tolerance for unseen images that might exhibit larger variations compared to the training data.

Figure 4

t-SNE of Yale B dataset using PCA (a) and t-SNE of Yale B dataset using KPCA+LDA (b)

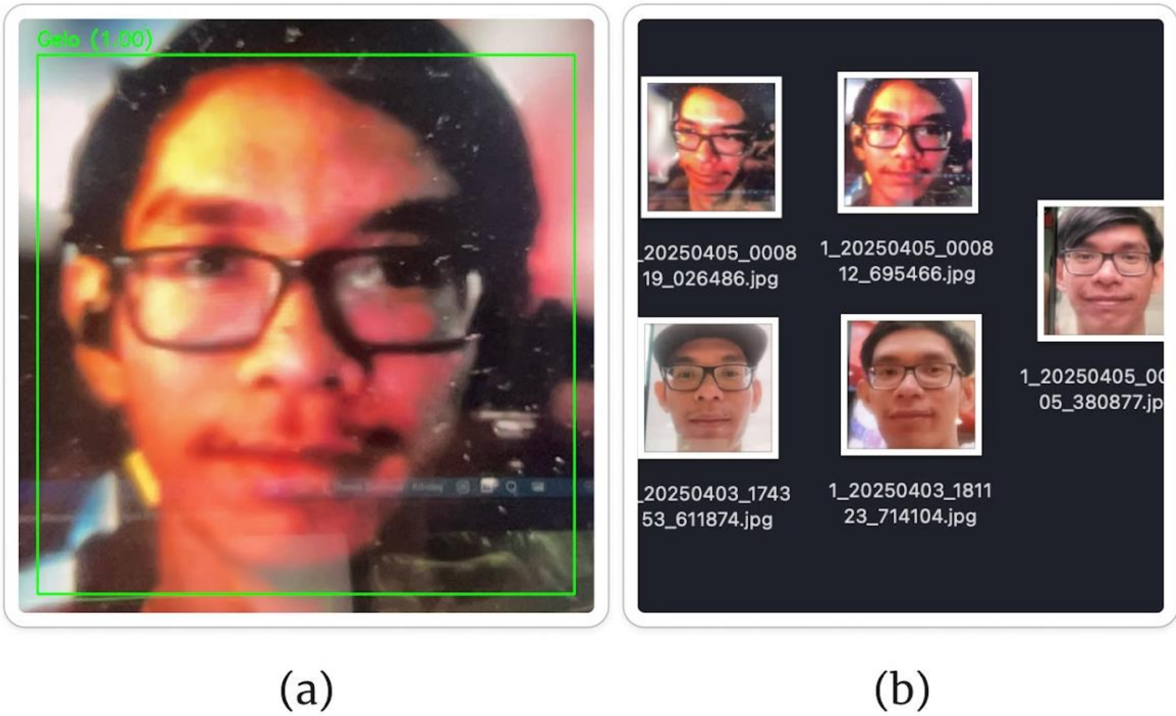


(a) (b)

Figure 5 illustrates the practical benefit of well-separated feature vectors for each face class. The enhanced algorithm's improved accuracy and precision in predicting the correct label for novel images can be attributed to this characteristic. This also strengthens the algorithm's generalization capabilities after it has been exposed to varied images of the same individual captured under diverse conditions such as lighting, pose, or image quality, enabling accurate classification of new, unseen instances.

Figure 5

Correct classification of subject "Gelo" (a) and training images of subject "Gelo" taken from various conditions and angles (b)



Ridge Classifier. The original Eigenface algorithm relies on Euclidean distance on feature vectors for classification. However, intra-class variations and inter-class similarities can compromise accuracy. As illustrated on figure 5, substantial variations and large distances can exist within the feature vectors of images belonging to the same class. Concurrently, the feature vectors of different face classes may exhibit overlap. Thus, relying on Euclidean distance even with the most optimal threshold (maximum allowable distance) may result in a high number of misclassifications.

Ridge Regression, a linear least squares method with L2 regularization, offers a more robust approach by minimizing errors and penalizing large weights, enhancing generalization for better feature discrimination. The Ridge Classifier handles multiclass problems by treating them as multi-output regression. For C classes, it learns C regression functions. Given a new sample, the predicted class is the one corresponding to the regression function with the highest output value. While other classification losses exist, the penalized least squares loss in Ridge Classifier can yield similar performance. Notably, its ability to compute the projection matrix

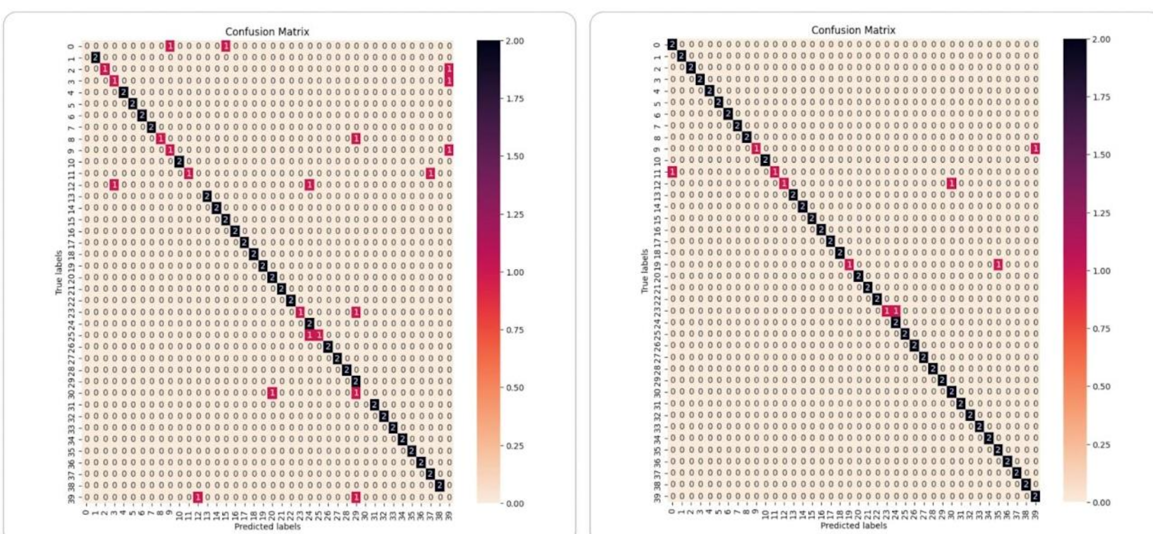
$(X^TX)^{-1}X^T$ only once provides computational efficiency, especially with many classes, making it a faster alternative in certain scenarios.

Ridge Classifiers initially output raw decision scores, not probabilities. To obtain meaningful probabilities (0-1), a calibration step is required. This involves training a secondary model (calibrator) to map the classifier's scores to calibrated probabilities using observed outcomes (Scikit-learn developers, n.d.). To avoid biased calibration due to the classifier's performance on its training data, the calibrator must be trained on unseen data. Cross-validation is commonly employed: the data is split into folds, the classifier is trained on some folds and predicts on a held-out fold, and the calibrator is trained on these out-of-sample predictions and true labels. This ensures a more realistic score-to-probability mapping, leading to reliable probability estimates on new data.

The researchers utilized a 5-fold cross-validation strategy with scikit-learn's Calibrated Classifier CV model, which internally employs a Ridge Classifier to directly output calibrated probabilities. This methodology necessitates a minimum of 5 images per face class to ensure that each fold in the cross-validation process contains at least one sample for training and validation of the classifier and subsequent calibration.

Figure 6

Confusion matrices for face classification on the Olivetti Faces dataset: (a) PCA features with Euclidean distance; (b) PCA features with Ridge Classifier.



(a) (b)

Figure 6 presents a comparative analysis of confusion matrices. The left matrix depicts the performance of the original algorithm, which employed PCA for dimensionality reduction and Euclidean distance for face classification. In contrast, the right matrix illustrates the performance of the same algorithm but with the classification step performed by a Ridge Classifier instead of Euclidean distance. For comparative evaluation, the Olivetti Faces dataset was utilized. The original algorithm, relying on Euclidean distance for classification, attained an accuracy of 81%. In contrast, the one incorporating a Ridge Classifier for the classification stage demonstrated a significant improvement, achieving an accuracy of 94%. It is important to note that the classification stage was the sole modification implemented in the original algorithm for this comparison. KPCA and LDA were intentionally excluded for the purpose of isolating the impact of the classification method, simplifying the comparison.

4. Findings and Discussion

This section details the outcomes of experiments designed to assess the performance of the enhanced Eigenface Algorithm in face recognition. The evaluation utilized two distinct datasets: the Yale B dataset and a custom dataset acquired via the researcher's laptop webcam. The enhanced Eigenface Algorithm incorporates several key modifications. Firstly, a preprocessing step employing the WLD was introduced to achieve illuminance-invariant image representations. Secondly, the traditional PCA for feature extraction was substituted with the Nyström method for Kernel PCA, utilizing a cosine kernel, and Linear Discriminant Analysis to promote more clustered feature vectors for each identity. Finally, the Euclidean distance metric used for classification in the original Eigenface approach was replaced with a Ridge Classifier to enhance classification robustness.

4.1. Improved Performance Under Varying Lighting Conditions Using WLD

The Yale B dataset, containing 2,414 images of 38 individuals under extreme lighting variations, was used to evaluate the enhanced Eigenface Algorithm's performance under diverse illumination. Employing a 50/50 train/test split, the results illustrated in figure 8 demonstrate a substantial improvement across all four evaluation metrics, most notably the increase in accuracy from the original algorithm's 5.6% to the enhanced algorithm's 99.83%.

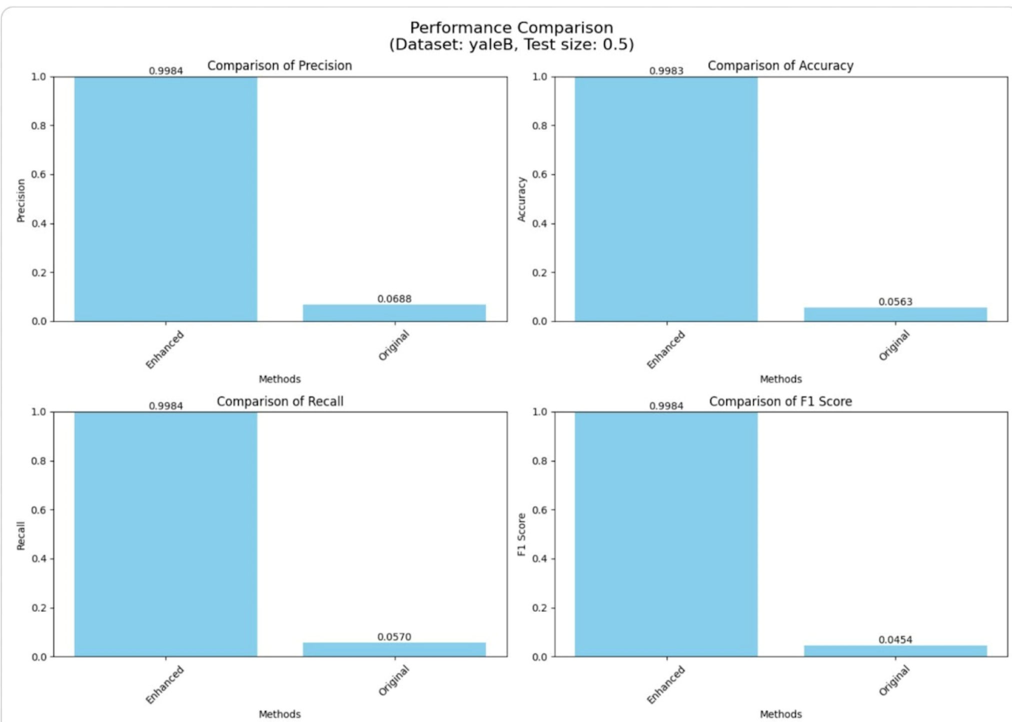
This indicates that the enhanced algorithm exhibits robust performance even when presented with face images captured under varying lighting conditions.

Figure 7

The first 5 images in the Yale B dataset

Figure 8

Performance of the original and enhanced Eigenface algorithm on the Yale B dataset



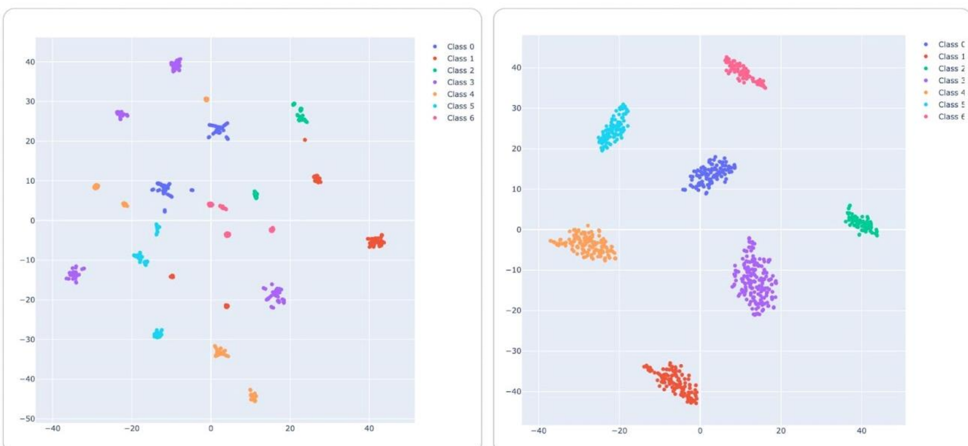
4.2. Improved Separation of Classes Using KPCA and LDA

To illustrate the enhanced separation of classes, the researchers compiled a custom dataset. This dataset, while featuring only 7 individuals, includes images of each subject captured under diverse real-world conditions encompassing variations in lighting, pose, and angle. The intentional variability in image acquisition was designed to demonstrate the scattering of feature vectors for each face class across the feature space, mirroring the challenges encountered in unconstrained face recognition scenarios.

Figure 9 displays the t-SNE projection of the feature vectors derived from the custom dataset. For the original algorithm, these feature vectors were obtained after applying PCA. In contrast, for the enhanced algorithm, the feature vectors visualized were generated after the application of KPCA followed by LDA. This shows the extreme attempt of the enhanced algorithm to separate the feature vector of each face class which results in a more stable classification.

Figure 9

t-SNE of custom dataset using PCA (a) and t-SNE of custom dataset using KPCA+LDA (b)



(a) (b)

Figure 10

The first 5 images in the custom dataset

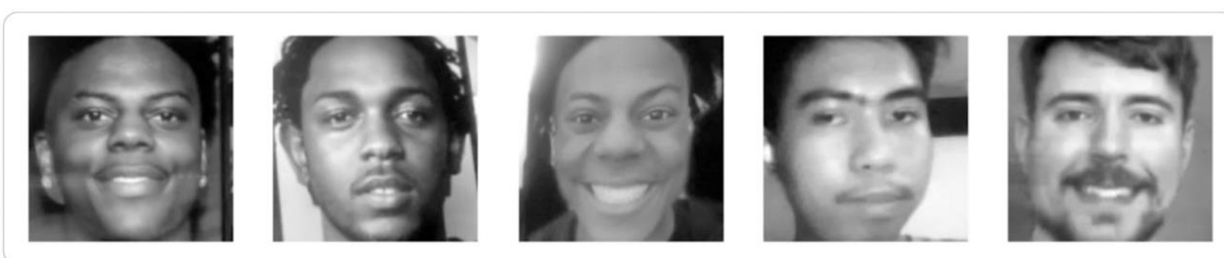


Figure 11 shows the evaluation of the enhanced Eigenface algorithm vs. the original Eigenface algorithm on the custom dataset. While the original algorithm achieved a high accuracy rate of 92%, the enhanced algorithm achieved 100% performance on all four metrics. It also generalizes well when the algorithm is used in real-time recognition using only a laptop webcam.

Figure 11

Performance of the original and enhanced Eigenface algorithm on the custom dataset

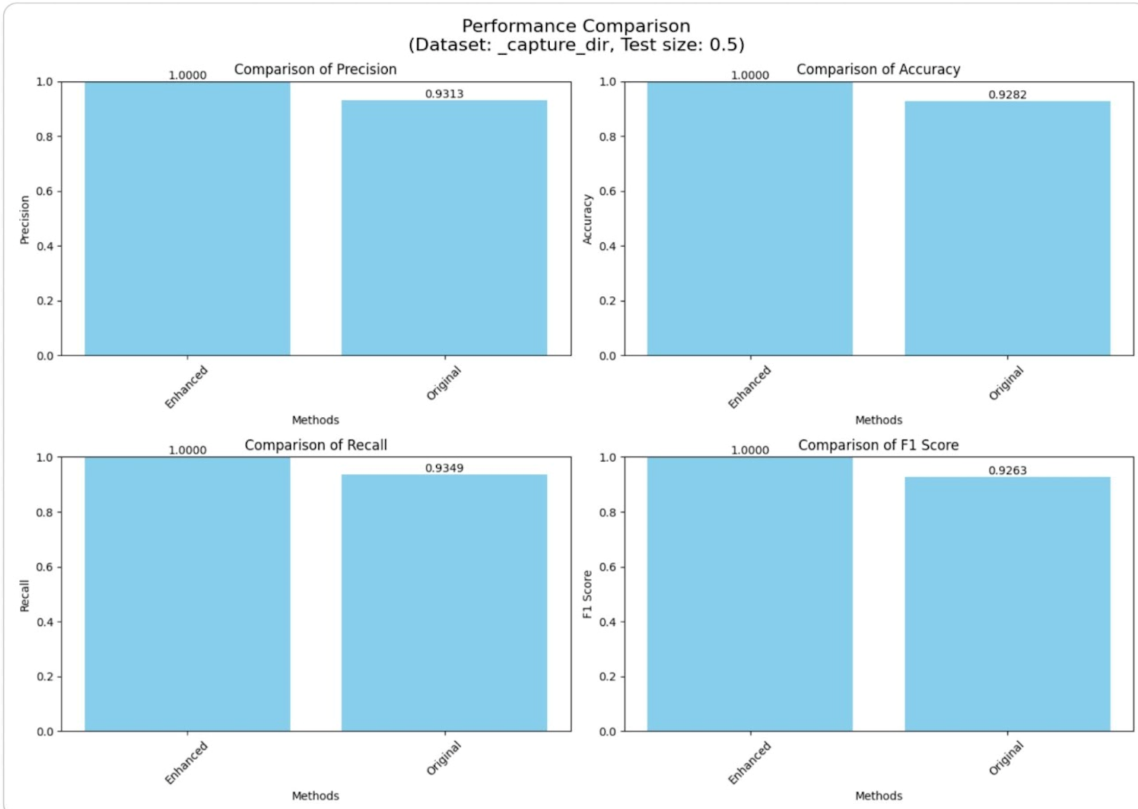
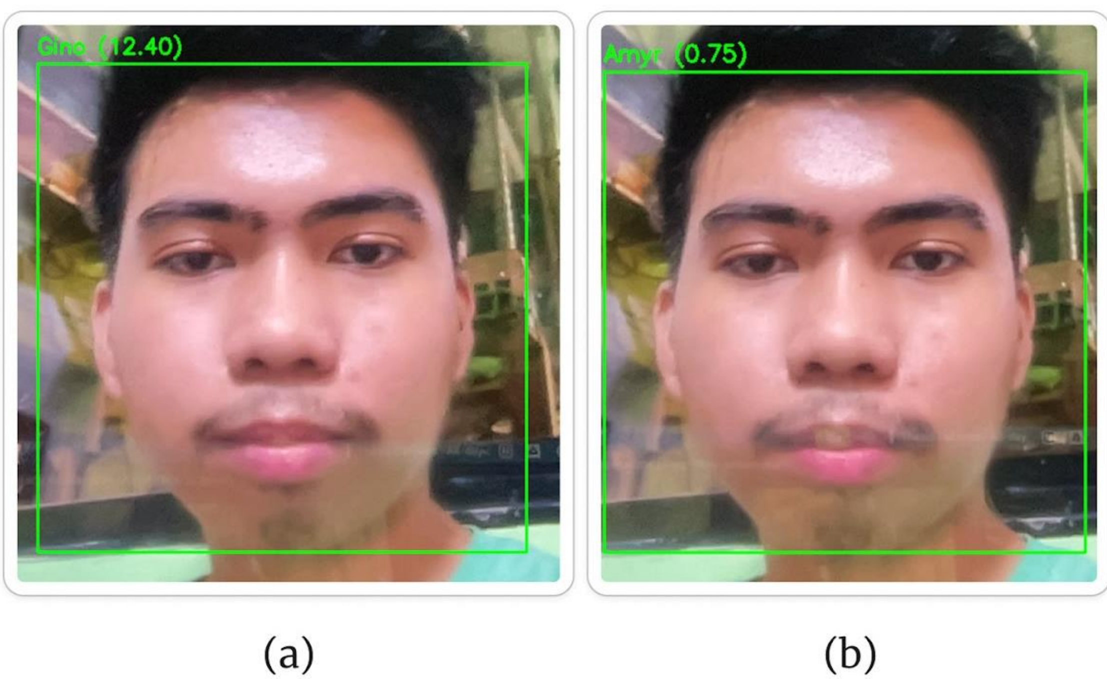


Figure 12

Recognition of an unseen image of subject "Amyr" with Original algorithm (a) and Enhanced algorithm (b)



To prove that the enhanced algorithm generalizes well in the real world, an unseen face image of subject "Amyr" was fed into the algorithm, both the original and enhanced. The original algorithm misclassified the face as belonging to subject "Gino" while the enhanced algorithm still classified it correctly as "Amyr".

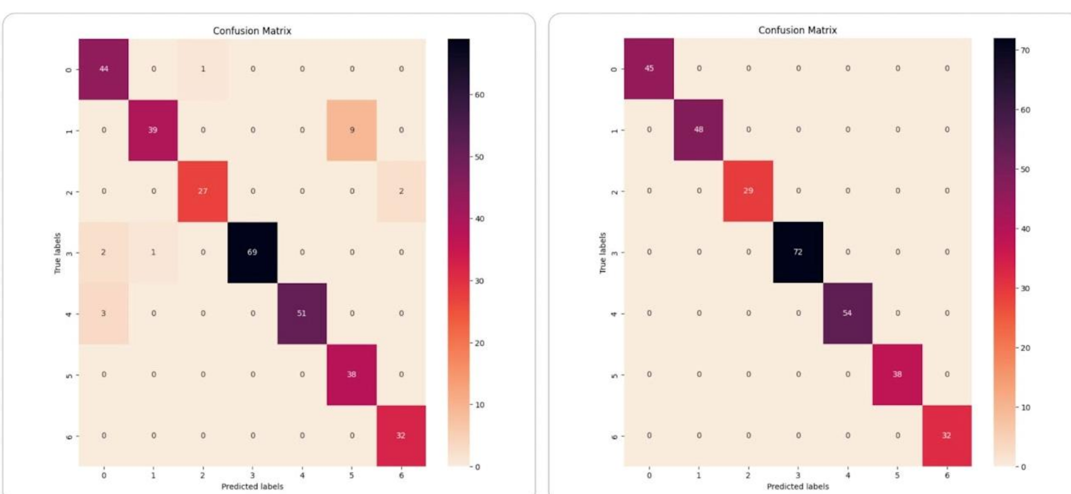
4.3. Improved classification precision using Ridge Classifier

To specifically examine the impact of employing a Ridge Classifier for the classification stage in face recognition, the original algorithm was modified solely by replacing the Euclidean distance metric with a Ridge Classifier. The deliberate exclusion of KPCA and LDA in this modified version was intended to isolate and assess the effect of the Ridge Classifier without the confounding influence of different feature extraction techniques.

Figure 13 presents a comparative analysis of confusion matrices obtained using the custom dataset, which features scattered feature vectors for each face class. Subfigure (a) illustrates the confusion matrix of the original, unmodified algorithm. Subfigure (b) displays the confusion matrix of the original algorithm where the classification step was modified to utilize a Ridge Classifier instead of Euclidean distance. The original algorithm achieved an accuracy of 94% on this dataset, while the modified algorithm incorporating the Ridge Classifier demonstrated a perfect accuracy of 100%.

Figure 13

Confusion matrices for face classification on the custom dataset: PCA features with Euclidean distance (a) and PCA features with Ridge Classifier (b)



(a) (b)

4.4. Improved Performance Against Larger Dataset

To assess the effectiveness of the enhanced algorithm, we conducted experiments on the wild_lfw dataset as seen in figure 14. This dataset contains 26,480 face images of 1,324 distinct individuals, with each image resized to a resolution of 64x64.

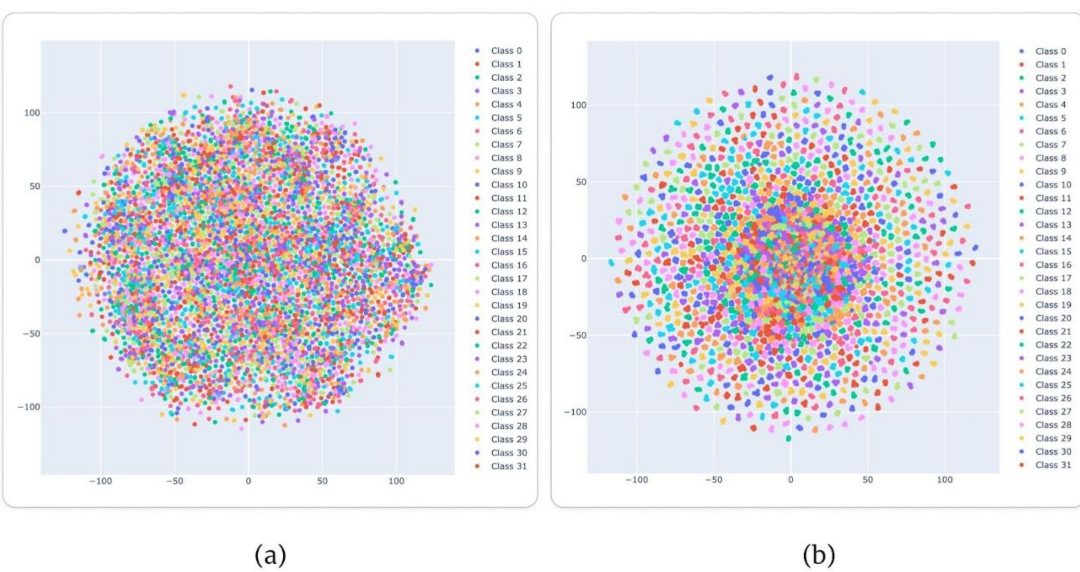
Figure 14

The first 10 images of the wild_lfw dataset.

The enhanced algorithm's improved classification performance on the Wild_LFW dataset is clearly demonstrated in figure 15. The feature space exhibits a substantially more refined structure, indicating a more effective embedding of facial identities and a diminished ambiguity in differentiating between individuals when compared to the original approach.

Figure 15

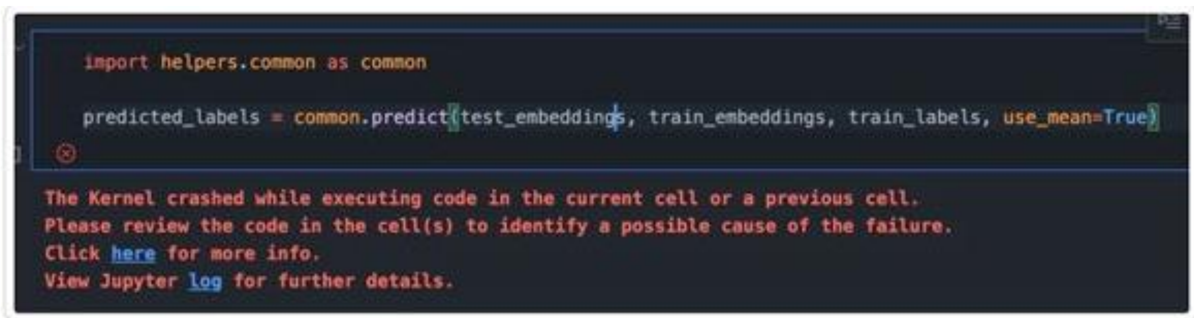
t-SNE of feature vectors on wild_lfw with original algorithm (a) and enhanced algorithm (b)



During the evaluation on the wild_lfw dataset, a critical limitation of the original algorithm was observed. As depicted in Figure 16, the original algorithm encountered a system failure and was unable to process the entirety of the dataset, presumably due to the substantial volume of image data inherent in wild_lfw.

Figure 16

The original algorithm crashed when tested against the wild_lfw dataset



```
import helpers.common as common

predicted_labels = common.predict(test_embeddings, train_embeddings, train_labels, use_mean=True)

The Kernel crashed while executing code in the current cell or a previous cell.
Please review the code in the cell(s) to identify a possible cause of the failure.
Click here for more info.
View Jupyter log for further details.
```

In contrast, the enhanced algorithm demonstrated robust performance on the same dataset, achieving a recognition accuracy of 82.46%, as detailed in table 1. This stark contrast underscores the improved stability and scalability of the enhanced algorithm in handling large-scale image datasets for facial recognition tasks.

Table 1

Comparison of metrics between the original and enhanced algorithm on the wild_lfw dataset

Metric	Original	Enhanced
Presicion	-	0.8372
Accuracy	-	0.8246
Recall	-	0.8246
F1 Score	-	0.8115

Across the spectrum of datasets employed for evaluation, the enhanced algorithm consistently outperformed its original counterpart. This superiority was particularly pronounced on specialized datasets designed to assess robustness against specific challenges. As evidenced in table 2 the enhanced algorithm demonstrated a significant advantage on the Yale B dataset, which is characterized by substantial variations in illumination.

Table 2

Comparison of metrics between the original and enhanced algorithm on the Yale B dataset

Metric	Original	Enhanced
Presicion	0.0688	0.9984
Accuracy	0.0561	0.9993
Recall	0.0370	0.9984
F1 Score	0.0454	0.9984

Similarly, table 3 reveals a notable performance gain on the Olivetti dataset, known for its inherent pose variations. Furthermore, the enhanced algorithm exhibited superior results on the custom dataset curated by the researchers, as presented in table 4. These findings collectively underscore the enhanced algorithm's improved generalization capabilities and its efficacy in mitigating the adverse effects of challenging conditions such as varying lighting and pose, as well as its adaptability to novel data distributions.

Table 3

Comparison of metrics between the original and enhanced algorithm on the Olivetti dataset

Metric	Original	Enhanced
Presicion	0.8208	0.9187
Accuracy	0.8125	0.8750
Recall	0.8125	0.8750
F1 Score	0.7975	0.8775

Table 4

Comparison of metrics between the original and enhanced algorithm on the custom dataset

Metric	Original	Enhanced
Presicion	0.9313	1.0000
Accuracy	0.9262	1.0000
Recall	0.9349	1.0000
F1 Score	0.9263	1.0000

5. Conclusion

The enhanced Eigenface algorithm presents significant improvements over the original algorithm by addressing three primary limitations. First, the enhanced algorithm tackles the original algorithm's sensitivity to varying lighting conditions by incorporating the Weber Local

Descriptor (WLD) as a preprocessing step. WLD effectively normalizes light levels, thus enhancing the algorithm's robustness under different illumination scenarios. Second, to improve class separability and dimensionality reduction, the enhanced algorithm replaces Principal Component Analysis (PCA) with Kernel PCA (KPCA) and Linear Discriminant Analysis (LDA). KPCA maps the data into a higher-dimensional space for better linear separability, while LDA optimizes the separation between different classes. For large datasets, the Nyström method is employed to approximate the kernel matrix in KPCA, increasing computational efficiency. Third, the enhanced algorithm replaces the Euclidean distance classifier with a Ridge Classifier to enhance classification accuracy and robustness. The Ridge Classifier uses L2 regularization to prevent overfitting, leading to improved generalization and performance. Overall, these enhancements contribute to a more accurate and robust face recognition system suitable for real-world applications.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was not supported by any funding.

AI Declaration

The author declares the use of Artificial Intelligence (AI) in writing this paper. In particular, the author used ChatGPT and Gemini in enhancing the grammar and writing of this paper as well as help in interpreting some of the literatures and other materials. The author takes full responsibility in ensuring that research idea, analysis and interpretations are original work.

ORCID

Amyr Edmar L. Francisco - https://orcid.org/0000-0003-2775-9351

References

Alam, S., Kwon, G., & The Alzheimer's Disease Neuroimaging Initiative (2017). Alzheimer disease classification using KPCA, LDA, and multi-kernel learning SVM. *International Journal of Imaging Systems and Technology*, *27*(2), 133–143. https://doi.org/10.1002/ima.22217

An, S., Liu, W., & Venkatesh, S. (2007). Face recognition using kernel ridge regression. *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-7. https://doi.org/10.1109/CVPR.2007.383105

Annubaha, C., Widodo, A. P., & Adi, K. (2022). Implementation of eigenface method and support vector machine for face recognition absence information system. *Indonesian Journal of Electrical Engineering and Computer Science*, *26*(3), 1624. https://doi.org/10.11591/ijeecs.v26.i3.pp1624-1633

Arlot, S., & Lerasle, M. (2015). Choice of V for V-Fold Cross-Validation in Least-Squares Density Estimation. *arXiv*. https://doi.org/10.48550/arXiv.1210.5830

Artac, M., Jogan, M., & Leonardis, A. (2002). Incremental PCA for on-line visual learning and recognition. *Object Recognition Supported by User Interaction for Service Robots*, *3*, 781–784. https://doi.org/10.1109/ICPR.2002.1048133

Carcagnì, P., Del Coco, M., Leo, M., & Distante, C. (2015). Facial expression recognition and histograms of oriented gradients: A comprehensive study. *SpringerPlus*, *4*(1), 645. https://doi.org/10.1186/s40064-015-1427-3

Chen, S., Luo, L., Yang, J., Gong, C., Li, J., & Huang, H. (n.d.). *Curvilinear Distance Metric Learning*. https://gcatnjust.github.io/ChenGong/paper/chen_nips19.pdf

Dahmouni, A., Abdelouahad, A. A., & Silkan, H. (2024). A noise-robust descriptor: applications to face recognition. *Journal of Theoretical and Applied Information Technology*, 102(10), 4361-4373.

Gong, D., Li, S. & Xiang, Y. (2011). Face recognition using the Weber Local Descriptor. *The First Asian Conference on Pattern Recognition*, 589–592. https://doi.org/10.1109/ACPR.2011.6166675

Dorbi, H., & Joshi, P. (2023). Face Recognition Algorithms: A Comparative Study. *International Research Journal of Modernization in Engineering Technology and Science*. https://doi.org/10.56726/IRJMETS41255

Dutta, A., Günther, M., El Shafey, L., Marcel, S., Veldhuis, R., & Spreeuwers, L. (2015). Impact of eye detection error on face recognition performance. *IET Biometrics*, *4*(3), 137–150. https://doi.org/10.1049/iet-bmt.2014.0037

Ebied, H. M. (2012). Kernel-PCA for face recognition in different color spaces. *2012 Seventh International Conference on Computer Engineering & Systems (ICCES)*, 201–206. https://doi.org/10.1109/ICCES.2012.6408513

Er, M. J., Chen, W., & Wu, S. (2005). High-speed face recognition based on discrete cosine transform and RBF neural networks. *IEEE Transactions on Neural Networks*, *16*(3), 679–691. https://doi.org/10.1109/TNN.2005.844909

Fahmy, G., El-Sherbeeny, A., Mandala, S., Abdel-Mottaleb, M., & Ammar, H. (2006). *The effect of lighting direction/condition on the performance of face recognition algorithms*, 62020J. https://doi.org/10.1117/12.666527

Fei, Z., Yang, E., Li, D., Butler, S., Ijomah, W., & Mackin, N. (2018). Eigenface algorithm-based facial expression recognition in conversations—an experimental study. In J. Ren, A. Hussain, J. Zheng, C.-L. Liu, B. Luo, H. Zhao, & X. Zhao (Eds.), *Advances in Brain Inspired Cognitive Systems* (Vol. 10989, pp. 342–351). Springer International Publishing. https://doi.org/10.1007/978-3-030-00563-4_33

Geroldinger, A., Lusa, L., Nold, M., & Heinze, G. (2023). Leave-one-out cross-validation, penalization, and differential bias of some prediction model performance measures—A simulation study. *Diagnostic and Prognostic Research*, *7*(1), 9. https://doi.org/10.1186/s41512-023-00146-0

Hallgren, F. (2022). Kernel PCA with the Nyström method (No. arXiv:2109.05578). *arXiv*. https://doi.org/10.48550/arXiv.2109.05578

Hastie, T. (2020). Ridge regularization: An essential concept in data science. *Technometrics*, *62*(4), 426–433. https://doi.org/10.1080/00401706.2020.1791959

Javed Mehedi Shamrat, F. M., Ghosh, P., Tasnim, Z., Khan, A. A., Uddin, Md. S., & Chowdhury, T. R. (2022). Human face recognition using eigenface, SURF Method. In G. Ranganathan, R. Bestak, R. Palanisamy, & Á. Rocha (Eds.), *Pervasive Computing and Social Networking* (Vol. 317, pp. 73–88). Springer Nature Singapore. https://doi.org/10.1007/978-981-16-5640-8_7

Jie Chen, Shiguang Shan, Chu He, Guoying Zhao, Pietikäinen, M., Xilin Chen, & Wen Gao. (2010). WLD: A robust local image descriptor. *IEEE Transactions on Pattern Analysis* *and Machine Intelligence*, *32*(9), 1705–1720. https://doi.org/10.1109/TPAMI.2009.155

Kekre, H. B., Sarode, T. K., & Save, J. K. (2014). Gender classification of human faces using class based PCA. *International Journal of Scientific and Research Publications*, 4(2), 1-9.

Kong, R., & Zhang, B. (2015). An effective new algorithm for face recognition. *International Conference on Computer Science and Intelligent Communication*, Zhengzhou, China. https://doi.org/10.2991/csic-15.2015.94

Kim, K., Jung, K. & Kim, H.J. (2002). Face recognition using kernel principal component analysis. *IEEE Signal Processing Letters*, *9*(2), 40–42. https://doi.org/10.1109/97.991133

Li, H., Wang, P., & Shen, C. (2010). Robust face recognition via accurate face alignment and sparse representation. *International Conference on Digital Image Computing: Techniques and Applications*, 262–269. https://doi.org/10.1109/DICTA.2010.54

Liliana, D.Y., & Setiawan, A.I.M. (2019). Face recognition with kernel principal component analysis and support vector machine. *International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS)*, 175–180. https://doi.org/10.1109/ICIMCIS48181.2019.8985199

Liu, C., Zhang, T., Ding, D., & Lv, C. (2016). Design and application of Compound Kernel-PCA algorithm in face recognition. *35th Chinese Control Conference (CCC)*, 4122–4126. https://doi.org/10.1109/ChiCC.2016.7553997

Liu, N., Wang, H., & Yau, W.-Y. (2006). Face recognition with weighted kernel principal component analysis. *9th International Conference on Control, Automation, Robotics and Vision*, 1–5. https://doi.org/10.1109/ICARCV.2006.345161

Maharani Raharja, N., Arief Fathansyah, M., & Nur Nazilah Chamim, A. (2021). Vehicle parking security system with face recognition detection based on eigenface algorithm. *Journal of Robotics and Control (JRC)*, *3*(1), 78–85. https://doi.org/10.18196/jrc.v3i1.12681

Maw, H.M., Thu, S.M., & Mon, M.T. (2020). Eigenface based facial expression recognition. *International Conference on Advanced Information Technologies (ICAIT)*, 117–122. https://doi.org/10.1109/ICAIT51105.2020.9261777

May, A. (2018). Kernel approximation methods for speech recognition. *Journal of Machine Learning Research*, 20, 1-36.

Mushtaq, Z., Qureshi, M. F., Abbass, M. J., & Al-Fakih, S. M. Q. (2023). Effective kernel-principal component analysis based approach for wisconsin breast cancer diagnosis. *Electronics Letters*, *59*(2), e212706. https://doi.org/10.1049/ell2.12706

Onaran, E., Sarıtaş, E., & Ekenel, H. K. (2024). Impact of face alignment on face image quality (No. arXiv:2412.11779). *arXiv*. https://doi.org/10.48550/arXiv.2412.11779

Park, J.-K., Park, H.-H., & Park, J. (2017). Distributed eigenfaces for massive face image data. *Multimedia Tools and Applications*, *76*(24), 25983–26000. https://doi.org/10.1007/s11042-017-4823-6

Peter, M., Minoi, J.-L., & Hipiny, I. H. M. (2019). 3D face recognition using kernel-based PCA Approach. In R. Alfred, Y. Lim, A. A. A. Ibrahim, & P. Anthony (Eds.), *Computational Science and Technology* (Vol. 481, pp. 77–86). Springer Singapore. https://doi.org/10.1007/978-981-13-2622-6_8

Rao, R. B., Fung, G., & Rosales, R. (2008). On the dangers of cross-validation. An experimental evaluation. *Proceedings of the 2008 SIAM International Conference on Data Mining*, 588–596. https://doi.org/10.1137/1.9781611972788.54

Rehman, A., Khan, A., Ali, M. A., Khan, M. U., Khan, S. U., & Ali, L. (2020). Performance analysis of PCA, Sparse PCA, Kernel PCA and Incremental PCA Algorithms for Heart Failure Prediction. *International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, 1–5. https://doi.org/10.1109/ICECCE49384.2020.9179199

Rosnelly, R., Simanjuntak, M. S., Clinton Sitepu, A., Azhari, M., Kosasi, S., & Husen. (2020). Face recognition using eigenface algorithm on laptop camera. *8th International Conference on Cyber and IT Service Management (CITSM)*, 1–4. https://doi.org/10.1109/CITSM50537.2020.9268907

Scikit-learn developers (n.d.). *1.1. Linear Models*. (n.d.). https://scikit-learn/stable/modules/linear_model.html

Scikit-learn developers (n.d.). *1.16. Probability calibration*. https://scikit-learn/stable/modules/calibration.html

Scikit-learn developers (n.d.). *6.7. Kernel Approximation*. https://scikit-learn/stable/modules/kernel_approximation.html

Ranganatha, S. & Gowramma, Y. P. (2022). Eigen and HOG features based algorithm for human face tracking in different background challenging video sequences. *International Journal of Image, Graphics and Signal Processing*, *14*(4), 70–83. https://doi.org/10.5815/ijigsp.2022.04.06

Shen, K., Wang, H., Chaudhuri, A., Asgharzadeh, Z. (2023). Automatic Gaussian Bandwidth Selection for Kernel Principal Component Analysis. In: Jin, Z., Jiang, Y., Buchmann, R.A., Bi, Y., Ghiran, AM., Ma, W. (eds) *Knowledge Science, Engineering and Management*. KSEM 2023. Lecture Notes in Computer Science(), vol 14117. Springer, Cham. https://doi.org/10.1007/978-3-031-40283-8_2

Singh, G., & Chhabra, I. (2018). Effective and fast face recognition system using complementary OC-LBP and HOG feature descriptors with SVM classifier. *Journal of Information Technology Research (JITR)*, *11*(1), 91–110. https://doi.org/10.4018/JITR.2018010106

Sirovich, L., & Kirby, M. (1987). Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A*, *4*(3), 519. https://doi.org/10.1364/JOSAA.4.000519

Sterge, N., & Sriperumbudur, B. (2021). Statistical optimality and computational efficiency of nyström kernel PCA (No. arXiv:2105.08875). *arXiv*. https://doi.org/10.48550/arXiv.2105.08875

Sterge, N., & Sriperumbudur, B. K. (2022). Statistical optimality and computational efficiency of nyström kernel PCA. *Journal of Machine Learning Research*, 23, 1-32.

Sterge, N., Sriperumbudur, B., Rosasco, L., & Rudi, A. (2020). Gain with no Pain: Efficiency of Kernel-PCA by Nyström sampling. *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, PMLR 108:3642-3652.

Prajapati, S.K. & Navamani, C. (2023). Effective approach for face recognition and active shape 3D models using kernel principal component analysis. *International Journal of Engineering Technology and Management Sciences*, *7*(2), 787–797. https://doi.org/10.46647/ijetms.2023.v07i02.085

Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, *3*(1), 71–86. https://doi.org/10.1162/jocn.1991.3.1.71

Wang, Y., & Zhang, Y. (2010). Facial recognition based on kernel PCA. *Third International Conference on Intelligent Networks and Intelligent Systems*, 88–91. https://doi.org/10.1109/ICINIS.2010.88

Zhang, Y.J. (Ed.). (2011). *Advances in face image analysis: techniques and technologies*. IGI Global. https://doi.org/10.4018/978-1-61520-991-0

Zhao, L., Dong, J., & Li, X. (2012). Research on KPCA and NS-LDA Combined Face Recognition. *Fifth International Symposium on Computational Intelligence and Design*, 140–143. https://doi.org/10.1109/ISCID.2012.43

Zhou, J., Liu, Y., & Chen, Y. (2007). Face recognition using kernel PCA and hierarchical RBF network. *6th International Conference on Computer Information Systems and Industrial Management Applications (CISIM'07)*, 239–244. https://doi.org/10.1109/CISIM.2007.28