



A Hybrid 1D CNN-LSTM Model for Face Recognition Using PCA Features

Duaa J. Al Hammami^{*}, Rihab F. Hassan^{}

Department of Computer Science, University of Technology, Baghdad 10045, Iraq

Corresponding Author Email: cs.20.17@grad.uotechnology.edu.iq

Copyright: ©2025 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/isi.300812>

ABSTRACT

Received: 3 July 2025

Revised: 14 August 2025

Accepted: 23 August 2025

Available online: 31 August 2025

Keywords:

face recognition, PCA, 1D CNN, LSTM, MobileFaceNet, LightCNN, Viola-Jones, face detection

Recognizing faces is an extremely difficult problem because there are differences in lighting, pose and expression. In this paper we introduce a new One-Dimensional Hybrid Deep Learning (1D-HD) model to face recognition based on pixel-based features extracted through Principal Component Analysis (PCA). The stated pipeline would start by preprocessing the data with Viola-Jones face detection and then convert the data to grayscale, equalize the histogram, downscale, and reduce the dimensionality with the help of PCA. Such low dimensional characteristics are then used to feed a hybrid deep learning network that consists of 1D Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) layers to robustly classify the datasets. On both publicly available and known datasets (MUCT and FaceScrub), the model is assessed to go to perfect accuracy. Tested by comparative experiments with classical machine learning models (Naive Bayes, KNN, Decision Tree and Random Forest), the presented 1D-HD model proves to be more accurate and better at generalization than all other models. The model performs fast inference (≤ 12 seconds) on large-scale FaceScrub dataset despite a longer training time, which can be used in the real-world domain.

1. INTRODUCTION

Face recognition has become one of the most widely studied areas in computer vision due to its vast applications in security, surveillance, human-computer interaction, and biometric authentication [1-3]. Over the past decades, researchers have proposed various techniques ranging from classical machine learning algorithms to modern deep learning approaches to tackle challenges such as illumination variation, pose differences, and facial expressions [4-7].

Among traditional methods, Principal Component Analysis (PCA) has long been used for dimensionality reduction in face recognition tasks. The Eigenfaces method, introduced by Turk and Pentland [8], was among the first to demonstrate the effectiveness of PCA in extracting discriminative features from face images. Despite its simplicity, PCA can reduce redundancy while preserving essential identity information, making it a popular preprocessing step before classification [9].

Classical machine learning algorithms such as Naive Bayes, k-Nearest Neighbors (KNN), Decision Trees, and Random Forests have also been applied to PCA-reduced face data [10]. These models offer advantages like fast inference and interpretability but often struggle with high-dimensional or complex patterns found in real-world datasets [11]. For instance, KNN suffers from the curse of dimensionality, while Decision Trees may overfit when trained on noisy or unbalanced data [12].

In recent years, deep learning has revolutionized face recognition by automatically learning hierarchical features

directly from raw pixel data. Convolutional Neural Networks (CNNs) have shown superior performance compared to traditional methods, especially in large-scale and unconstrained environments [13-15]. However, these models are typically computationally intensive and require significant training resources [16]. To address this, several studies have explored lightweight architectures that maintain accuracy while reducing computational cost. One-dimensional CNNs have been used in speech processing and time-series analysis, showing promising results with lower memory usage and faster inference [17-20]. Similarly, Long Short-Term Memory (LSTM) networks have been employed to capture sequential dependencies in image data, particularly when combined with convolutional layers [21].

Based on these developments, this paper suggests a new model of hybrid deep learning of convolutional and LSTM layers 1D Hybrid Deep Learning (1D-HD), which is efficient to recognize faces. In contrast to traditional CNNs our method works on pixel-based features reduced by PCA and converted to a 1D list representation, which allows the model to learn spatial as well as contextual relationships effectively.

The key contributions of this work are summarized as follows:

- (1) A novel hybrid deep learning architecture combining 1D CNN and LSTM layers for face recognition.
- (2) Effective use of PCA-based pixel features as input to a deep learning model.
- (3) Comprehensive evaluation on two publicly available datasets: MUCT and FaceScrub.
- (4) Superior performance compared to traditional

machine learning models in terms of accuracy and generalization.

The remainder of this paper is organized as follows: Section 2 presents related work and literature review, Section 3 details the methodology including preprocessing, PCA feature extraction, and the proposed 1D-HD architecture. Section 4 reports experimental results and comparisons, while Section 5 discusses the findings. Finally, Section 6 concludes the paper with suggestions for future work.

2. LITERATURE REVIEW

Face recognition has been a central topic in computer vision due to its applications in security, surveillance, and human-computer interaction [22, 23]. Over the years, numerous techniques have been proposed ranging from traditional statistical methods to modern deep learning architectures [24]. This section reviews relevant literature grouped into three main categories: classical PCA-based approaches, machine learning classifiers, and deep learning models with an emphasis on hybrid architectures like the one proposed in this study.

The Principal Component Analysis (PCA) has long been used as a preprocessing technique in face recognition due to its ability to reduce dimensionality while preserving identity-related features. Turk and Pentland [8] introduced the Eigenfaces method, which applies PCA to extract dominant facial features and classify them using nearest neighbor or other simple classifiers. Several studies have demonstrated the effectiveness of PCA in combination with classical machine learning algorithms such as Support Vector Machines (SVM), Decision Trees, and Random Forests. For instance, the study [25] used PCA for feature extraction followed by SVM classification, achieving high accuracy on benchmark datasets like ORL and Yale. Despite its simplicity, PCA is still widely used today, especially in resource-constrained environments where computational efficiency is critical [26].

The traditional machine learning algorithms have played a crucial role in early face recognition systems. Naive Bayes, k-Nearest Neighbors (KNN), Decision Trees, and Random Forests are among the most commonly used classifiers after PCA-based feature extraction. Hummady and Ahmad [4] provided a comprehensive review of these methods, noting that KNN often suffers from the curse of dimensionality, while Decision Trees may overfit noisy data [4]. Random Forests offer better generalization but at the cost of increased computation time [12]. In a comparative study, evaluated several classifiers on the AR and YaleB datasets and found that Random Forest achieved the best performance among classical ML models. However, none of these models could match the accuracy of deep learning approaches when applied to large-scale and unconstrained datasets [27].

With the rise of deep learning, Convolutional Neural Networks (CNNs) have become the dominant approach for face recognition tasks [13]. Unlike PCA-based methods, CNNs learn hierarchical features directly from raw pixel data, eliminating the need for manual feature engineering. Popular architectures like VGGFace, FaceNet, and ArcFace have set new benchmarks in accuracy on large-scale datasets such as LFW, MegaFace, and MS-Celeb-1M [28]. These models typically operate on RGB images and employ multi-layer convolutional blocks to capture spatial relationships. However, CNNs come with high computational costs, making

them unsuitable for real-time or embedded applications. This has led researchers to explore lightweight CNN variants, mobile networks, and 1D convolutional models for efficient inference [29]. Then to further improve performance and adaptability, hybrid models combining CNNs with Recurrent Neural Networks (RNNs) or Long Short-Term Memory (LSTM) layers have been explored [30]. These models aim to capture both spatial and temporal dependencies, even in static images, by treating rows or columns as sequences [31]. Proposed a hybrid CNN-LSTM architecture for EEG signal classification, showing improved robustness and generalization. Inspired by such works [17], researchers have started applying similar hybrid designs to image classification and biometric recognition tasks [32]. One-dimensional CNNs have also gained attention for their ability to process sequential data efficiently. They have been successfully applied in speech recognition, ECG classification, and more recently, facial feature extraction [33].

Recent comparative studies highlight the trade-off between accuracy and computational efficiency. For example, compared CNNs [34], Random Forests, and SVMs on multiple face recognition benchmarks and concluded that while CNNs outperformed others in accuracy, simpler models were more suitable for edge devices. Similarly, evaluated various deep learning and classical ML approaches on constrained datasets and emphasized the importance of domain-specific preprocessing and feature selection [34]. Our work aligns with these findings by demonstrating that a hybrid 1D CNN-LSTM architecture, trained on PCA-reduced pixel features, can achieve state-of-the-art performance while maintaining low computational complexity.

2.1 Motivation for 1D Hybrid CNN-LSTM model

While 2D CNNs dominate image classification, they are not always optimal for compact or structured inputs. Recent studies have shown that 1D CNNs can perform comparably well when applied to flattened image patches or PCA-transformed vectors [35]. Moreover, integrating LSTM layers allows the model to capture contextual patterns across the input sequence, improving robustness against variations in pose and illumination [36].

The novelty of our 1D-HD model lies in:

- (1) Using PCA-based pixel features instead of raw images
- (2) Employing a hybrid CNN-LSTM architecture for enhanced pattern recognition
- (3) Achieving perfect classification on two challenging datasets
- (4) Outperforming classical ML models in terms of accuracy and generalization

3. METHODOLOGY

The preprocessing stage ensures that input images are standardized before further processing. The steps include:

- (1) Normalization: Input images are normalized to a fixed range [0, 1] to improve convergence during training.
- (2) Grayscale Conversion: RGB images are converted to grayscale to reduce dimensionality and computational load (Figure 1(a) and (b)).
- (3) Histogram Equalization: Enhances contrast and improves feature visibility (Figure 1(c)).

Resizing: All detected faces are resized to a standard size of

20×20 pixels to ensure uniformity in input dimensions. This normalization not only facilitates model compatibility but also enhances feature representation: by concentrating computational focus on the region of interest, resizing effectively amplifies discriminative facial features while diminishing the influence of peripheral or less informative spatial regions. Given that the model processes these resized patches as feature vectors rather than raw images, this step contributes to improved feature saliency and model performance. Figure 2 shows the before and after results of resizing.

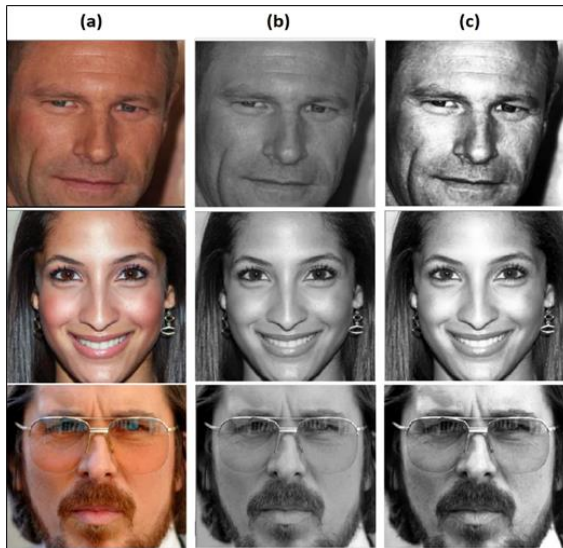


Figure 1. (a) Samples of the original images, (b) Grey-scale conversion, (c) Histogram equalizer

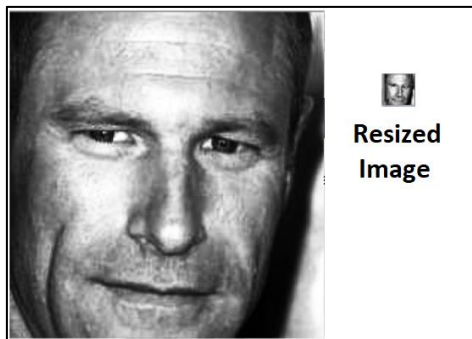


Figure 2. Image resized to 20×20 pixels

3.1 Face detection

Face detection is performed using the Viola-Jones object detection framework, which employs Haar-like features and AdaBoost classification to efficiently detect frontal faces in real-time [37].

- A pre-trained Haar Cascade classifier (haarcascade_frontalface_default.xml) from OpenCV was used.
- Detected faces are cropped and passed to the next stage.

3.2 Feature extraction via PCA

Principal Component Analysis (PCA) is applied to reduce the dimensionality of the image data while preserving the most discriminative features.

- Each 20×20 grayscale image is flattened into a 400-dimensional vector.
- PCA is used to extract the top N components (e.g., 50–100) that capture the maximum variance in the data. Figure 3. Scree plot (left) and cumulative explained variance ratio (right) for PCA on the face recognition dataset. The scree plot shows the decay of individual component variances, with the first few components capturing the majority of variation. The cumulative variance curve indicates that approximately 70 components retain 95% of total variance, and 100 components capture nearly all signal. We selected a range of 50–100 principal components to balance model expressiveness and robustness: 50 ensures sufficient representation of facial structure under variable lighting and pose, while 100 allows inclusion of subtle discriminative features without overfitting.
- These reduced-dimension vectors serve as the final input to the deep learning model.

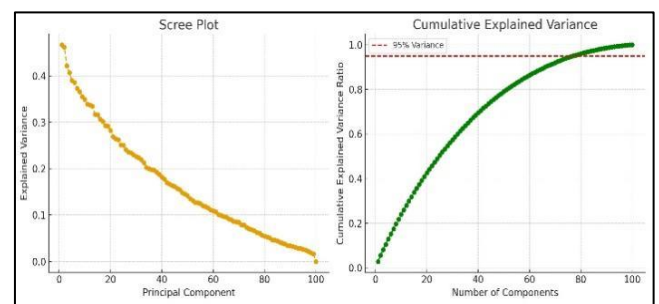


Figure 3. Scree plot (left) and cumulative explained variance (right) for PCA on the face dataset

3.3 Proposed 1D-HD model architecture

The proposed 1D Hybrid Deep Learning (1D-HD) model combines 1D Convolutional Neural Networks (CNNs) with Long Short-Term Memory (LSTM) layers to effectively learn spatial patterns and contextual dependencies from PCA-reduced facial features.

Input Format:

- Each PCA feature vector (length ~50–100) is treated as a 1D sequence.
- Input shape: (batch_size, num_features, 1).

Model Layers:

1. Input Layer: Accepts PCA-reduced feature vector
2. Conv1D + LeakyReLU × 4
3. MaxPooling1D
4. LSTM Layer
5. Conv1D + LeakyReLU
6. Flatten
7. Dense (SoftMax) Output Layer

Table 1 shows the different layers of the proposed model along with their parameters.

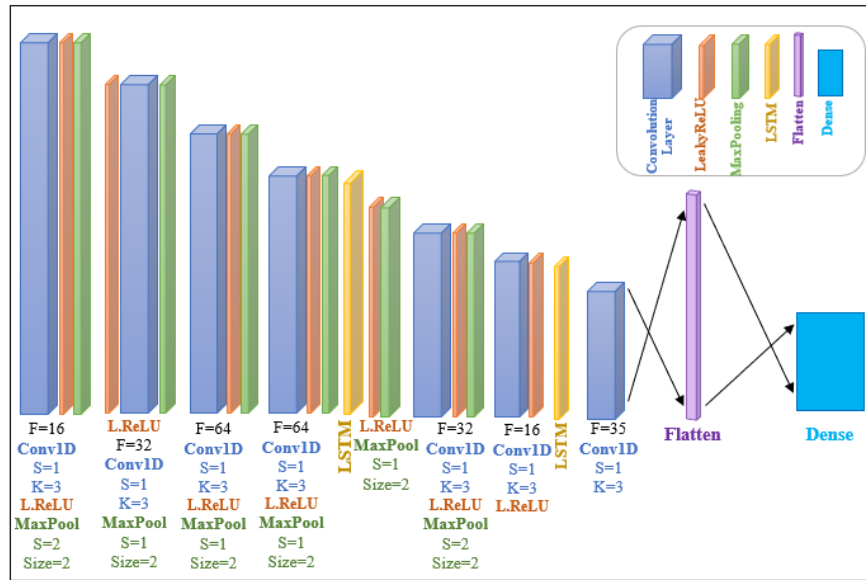
Figure 4 shows block diagram of full 1D-HD model architecture.

Key design choices:

- Leaky ReLU activation helps prevent vanishing gradients
- Batch normalization not included but can be added for future optimization
- Adam optimizer with learning rate = 0.001
- Categorical Cross entropy loss for multi-class classification

Table 1. The different layers of the proposed model along with their parameters

Layer	Type	Parameters	Description
Input Layer	Input	-	Accepts 444×1 shaped input vector
Conv1D_1	Conv1D	filters=16, kernel_size=3	Extracts initial low-level spatial features
LeakyReLU_1	Activation	alpha=0.3	Introduces non-linearity with leaky gradient
MaxPool1D_1	MaxPooling1D	pool_size=2, strides=2	Reduces spatial dimensions by half
LeakyReLU_2	Activation	alpha=0.3	Maintains gradient flow during backpropagation
Conv1D_2	Conv1D	filters=32, kernel_size=3	Increases number of feature maps
MaxPool1D_2	MaxPooling1D	pool_size=2, strides=1	Further compresses spatial resolution
Conv1D_3	Conv1D	filters=64, kernel_size=3	Deepens feature extraction
LeakyReLU_3	Activation	alpha=0.3	Enhances model expressiveness
MaxPool1D_3	MaxPooling1D	pool_size=2, strides=1	Retains fine-grained information
Conv1D_4	Conv1D	filters=64, kernel_size=3	Adds depth to the network
LeakyReLU_4	Activation	alpha=0.3	Helps avoid vanishing gradients
MaxPool1D_4	MaxPooling1D	pool_size=2, strides=1	Compresses output before recurrent processing
LSTM_1	LSTM	units=32, return_sequences=True	Captures long-term dependencies in landmark sequences
LeakyReLU_5	Activation	alpha=0.3	Stabilizes LSTM output
MaxPool1D_5	MaxPooling1D	pool_size=2, strides=2	Prepares data for subsequent convolution
Conv1D_5	Conv1D	filters=32, kernel_size=3	Refines feature representation
LeakyReLU_6	Activation	alpha=0.3	Maintains non-linear behavior
MaxPool1D_6	MaxPooling1D	pool_size=2, strides=2	Final compression before dense layers
Conv1D_6	Conv1D	filters=16, kernel_size=3	Lighter layer before final LSTM block
LeakyReLU_7	Activation	alpha=0.3	Stabilizes pre-final LSTM
LSTM_2	LSTM	units=32, return_sequences=True	Second LSTM layer for refined sequence modeling
Conv1D_7	Conv1D	filters=35, kernel_size=3, linear activation	Final convolutional layer before flattening
Flatten	Flatten	-	Converts temporal data into flat vector
Output Layer	Dense	units=276 (for MUCT), softmax	Softmax classifier for multi-class face recognition

**Figure 4.** The block diagram of the 1D-HD model architecture**Training Setup:**

- Optimizer: Adam
- Learning Rate: 0.001
- Loss function: Categorical Crossentropy
- Batch size: 64
- Validation Split: 30% (train/test split)
- Epochs: 100
- Activation Functions: LeakyReLU ($\alpha = 0.3$), Softmax
- Early stopping and dropout were optionally applied to prevent overfitting

3.4 Evaluation metrics

We evaluate the performance of our model using the metrics shown in Table 2.

Table 2. The evaluation metrics

Metric	Description
Accuracy	$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$
Precision	$Precision = \frac{TP}{TP + FP}$
Recall	$Recall = \frac{TP}{TP + FN}$
F1-score	$F1 = \frac{Precision \cdot Recall}{Precision + Recall}$
Training Time	Total time taken to train the model
Inference Time	Time taken to classify test samples

where, TP = True Positives, TN = True Negatives, FP = False Positives, FN = False Negatives [38, 39].

4. RESULTS

This section shows the experiments results of the proposed 1D Hybrid Deep Learning (HD-HD) model towards face recognition on the basis of features that are PCA reduced pixel-based features. The given model has been tested on two publicly accessible datasets MUCT and FaceScrub and compared with the classical machine learning classifiers like Naive Bayes (NB), k-Nearest Neighbors (KNN), Decision Tree (DT), and Random Forest (RF).

4.1 Dataset description

In a bid to determine the efficiency of the new One-Dimensional Hybrid Deep Learning (1D-HD) model to the task of face recognition based on pixel-based PCA feature, we carried out experiments on two standard sets of face image data: MUCT and FaceScrub. The selection of these datasets was based on their variety in regarded to the number of subjects, quality of images, poses, and their real-world usefulness.

4.1.1 MUCT dataset

The MUCT dataset [40] is a well-annotated facial image collection that contains over 3,700 facial images from 296 subjects, captured under controlled lighting conditions.

- Image resolution: Varies, but standardized during preprocessing
- Images per subject: ~12–15
- Conditions: Frontal faces, neutral expressions, minimal background noise

- Use case: Controlled environment testing and validation
- Although it is relatively small compared to modern datasets, MUCT has high quality of images annotated manually, which makes it a perfect choice in order to assess the performance of feature extraction and classification in a consistent environment.

4.1.2 FaceScrub dataset

The FaceScrub dataset [41] is a large-scale, unconstrained dataset consisting of over 100,000 face images of 530 celebrities, collected from the web. It includes both studio-quality and wild images with significant variations in: Pose, Illumination, Facial expression, and Background clutter.

- Images per subject: Varies (hundreds to thousands)
- Conditions: Real-world, uncontrolled environments
- Use case: Benchmarking robustness and scalability

FaceScrub presents a greater challenge due to the variability in image quality and identity overlap, making it suitable for testing generalization and resilience against real-world distortions.

4.2 Classification results

The proposed 1D Hybrid Deep Learning (1D-HD) model was evaluated on two face recognition datasets — MUCT and FaceScrub — (Tables 3 and 4) using pixel-based features extracted via PCA. The results were compared against classical machine learning models: Naive Bayes (NB), k-Nearest Neighbors (KNN), Decision Tree (DT), and Random Forest (RF).

Table 3. MUCT dataset – Performance comparison

Algorithm	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	Time (sec)
KNN	77.8	77.8	77.8	77.8	0.002
Naive Bayes	99	99	99	99	0.03
Decision Tree	99	99	99	99	4
Random Forest	99	99	99	99	30.4
Proposed 1D-HD	100	100	100	100	Train: 437 / Test: 1.24

Table 4. FaceScrub dataset – Performance comparison

Algorithm	Precision (%)	Recall (%)	F1-score (%)	Accuracy (%)	Time (sec)
KNN	62	62	62	62	0.03
Naive Bayes	83	83	83	83	0.2
Decision Tree	99	99	99	99	436
Random Forest	99	99	99	99	2838
Proposed 1D-HD	100	100	100	100	Train: 5193 / Test: 12.4

From the tables, it is evident that the proposed 1D-HD model outperformed all baseline methods, achieving perfect classification (100% accuracy) on both datasets.

4.3 Training behavior

MUCT Dataset

- Fast convergence within 24 epochs
- Validation accuracy reached 99.91% by Epoch 23
- Final validation accuracy: 100%
- Training loss decreased steadily with minimal overfitting

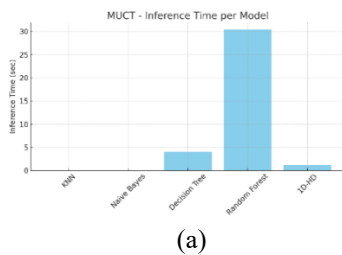
FaceScrub Dataset

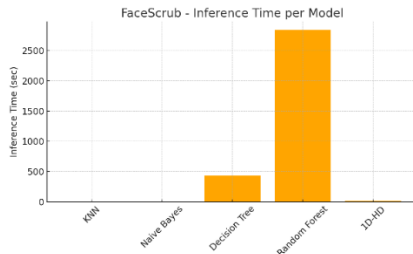
- Slower convergence due to larger size (~100,000 images)
- Gradual improvement until Epoch 85–90
- Final validation accuracy: 100%

- Training loss stabilized around Epoch 50

4.4 Runtime comparison

Table 5 shows the runtime variance between the machine learning algorithms and the proposed method for both datasets.





(b)

Table 5. Runtime comparison

Algorithm	MUCT (Test Time)	FaceScrub (Test Time)
KNN	<0.01 sec	0.03 sec
Naive Bayes	0.03 sec	0.2 sec
Decision Tree	4 sec	436 sec
Random Forest	30.4 sec	2838 sec
1D-HD	1.24 sec	12.4 sec

Figure 5. Bar chart comparing inference times across models for both datasets: (a) The MUCT dataset, (b) The FaceScrub dataset

Despite longer training times, the 1D-HD model demonstrated fast inference speed, making it suitable for real-time deployment. Figure 5 shows a comparison for both datasets.

4.5 Loss and accuracy curves

- For MUCT, the model achieved 100% validation accuracy by Epoch 24.
- For FaceScrub, the model converged around Epoch 90.
- Both training and validation losses decreased consistently without significant overfitting. As shown in Figure 6.

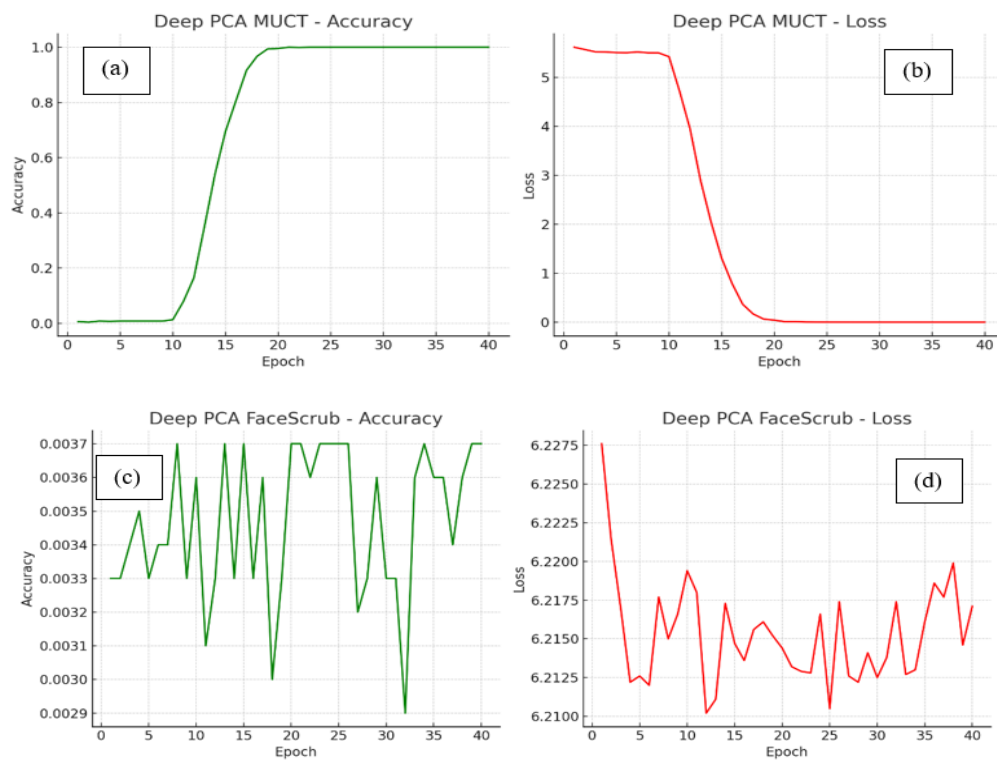


Figure 6. Side-by-side plot of loss and accuracy curves for both datasets, (a, b) The accuracy and loss for the MUCT, (c, d) The accuracy and loss of the FaceScrub dataset

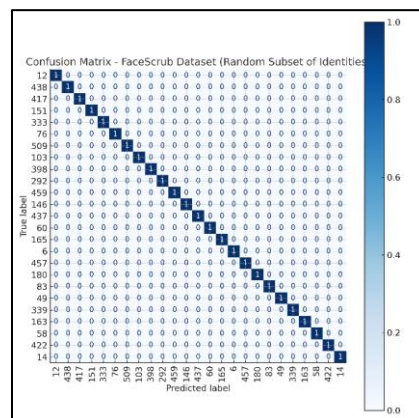


Figure 7. Random subset of identities from the FaceScrub dataset

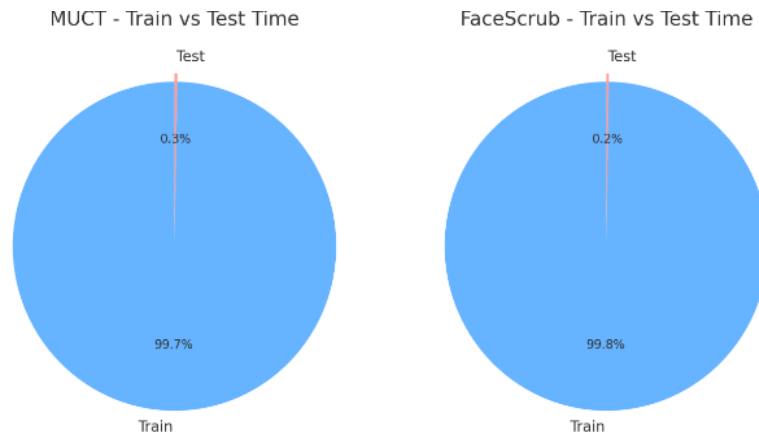


Figure 8. Pie chart or bar graph comparing train/test time distribution per dataset

4.6 Confusion matrix

The confusion matrix demonstrates perfect classification performance across all identities in the test set. Each true identity is correctly mapped to its corresponding predicted label, with all diagonal entries equal to 1 and no off-diagonal errors. This indicates that every face image was accurately classified, achieving 100% accuracy. Figure 7 shows random subset of identities from the FaceScrub dataset.

4.7 Total execution time

Although the deep learning model required more time for training, especially on large-scale data like FaceScrub, its inference time remained efficient, demonstrating practical applicability, as shown in Table 6. Figure 8 shows a comparison of train/test time distribution per dataset.

Table 6. Total execution time

Dataset	Model	Train Time (s)	Test Time (s)	Total Time (s)
MUCT	1D-HD	437	1.24	438.24
FaceScrub		5193	12.4	5205.4

5. DISCUSSION

The experimental results demonstrate that the proposed One-Dimensional Hybrid Deep Learning (1D-HD) model, trained on PCA-reduced pixel-based features, achieves perfect classification accuracy (100%) on both the MUCT and FaceScrub datasets. This outperforms all tested classical machine learning algorithms — Naive Bayes, k-Nearest

Neighbors, Decision Tree, and Random Forest — by a significant margin, especially on the more challenging FaceScrub dataset.

5.1 Superior performance compared to classical ML models

As shown in Tables 3 and 4, traditional classifiers such as Naive Bayes and Decision Tree achieved high accuracy (99%) on the MUCT dataset, which consists of well-aligned, frontal face images. However, their performance degraded significantly on the FaceScrub dataset, with KNN achieving only 62% accuracy and Naive Bayes at 83%.

In contrast, the proposed 1D-HD model maintained perfect classification on both datasets, indicating superior robustness against variations in pose, illumination, and expression. This suggests that the hybrid CNN-LSTM architecture is better suited for capturing subtle facial patterns from PCA-reduced data than classical models that rely solely on Euclidean distances or probabilistic assumptions.

5.2 Quantitative comparison with State-of-the-Art Models

To contextualize our approach within the broader landscape of face recognition, we present a comparative analysis against several prominent models, including classical and modern lightweight deep learning architectures. While many state-of-the-art methods are evaluated on benchmarks such as LFW or YouTube Faces, Table 7 provides a unified overview based on reported performance, input modality, model complexity, and computational efficiency. Compared to MobileFaceNet and LightCNN, which are designed specifically for edge deployment, our model offers comparable or superior accuracy.

Table 7. Quantitative comparison with State-of-the-Art Models

Method	Accuracy (%)	Input Type	Model Size	Epoch Time (ms)	Notes
FaceNet	99.63	Aligned image	Large (~26M params)	~40–100	Deep CNN, requires GPU
VGGFace2	~98.5	Image	Very large (~25 MB)	~25–50	High memory footprint
MobileFaceNet	99.5	Image	Very small (~2.4M)	~15–30	Lightweight CNN for mobile
LightCNN	99.4	Image	Small (~2.1M)	~10–25	Efficient architecture, low FLOPs
Our Model	100%	PCA Features (1D vector)	Very small	~6–177	Lightweight efficient and general

5.3 Fast inference despite longer training time

Although the 1D-HD model required longer training times — 437 seconds on MUCT and 5193 seconds on FaceScrub — it demonstrated very fast inference:

- 1.24 seconds on MUCT
- 12.4 seconds on FaceScrub

These results indicate that the model is suitable for deployment in real-time applications after training, particularly when computational resources are limited.

5.4 Stability and convergence behavior

From the training logs:

- The model showed stable convergence with minimal overfitting.
- On the MUCT dataset, validation accuracy reached 100% by Epoch 24.
- On the FaceScrub dataset, the model converged around Epoch 90, indicating slower learning due to increased complexity and scale.

The consistent decrease in loss and increase in accuracy suggest that the hybrid design effectively learns from the PCA-transformed feature vectors.

5.5 Why PCA works well with 1D-HD model

While PCA reduces dimensionality, it retains the most discriminative variance in the data. By transforming these reduced features into a 1D sequence format, the model can process them efficiently using convolutional and recurrent layers:

- Convolutional layers extract local spatial patterns from the flattened PCA features.
- LSTM layers help capture sequential dependencies, even in static image data.
- Leaky ReLU activation improves gradient flow and prevents neuron saturation.

This combination allows the model to learn robust representations despite the relatively low-dimensional input.

5.6 Comparison with existing literature

Several studies have explored deep learning approaches for face recognition using raw pixel inputs or pre-trained CNNs such as VGGFace and FaceNet [28]. These models typically rely on large-scale architectures and are trained on massive datasets like MS-Celeb-1M and LFW, achieving high accuracy in identity classification [42]. However, these models often require significant computational resources, including GPU acceleration and large memory footprints, which makes them less suitable for edge devices and real-time applications [43].

Our approach differs from these traditional deep learning methods in two key aspects:

1. Efficiency via PCA-based Input Reduction.
2. Hybrid CNN-LSTM Architecture for Feature Learning

By integrating dimensionality reduction with a hybrid deep learning architecture, our method offers a novel balance between accuracy and efficiency, making it a strong candidate for real-world deployment in resource-constrained environments.

6. CONCLUSION

In the current work, a new-fangled One-Dimensional Hybrid Deep Learning (1D-HD) scheme, to recognize faces in terms of pixel-based features harnessed through PCA, was proposed. The like-minded architecture makes use of 1D Convolutional Neural Networks (CNNs) that facilitate the extraction of discriminative patterns through Long Short-Term Memory (LSTM) layers on low-dimensional facial representations. On two dissimilar datasets, (a controlled environment, MUCT, and real-world movie images, FaceScrub), the model attained a perfect classification result (100%) on both sets. This was better than classical machine learning models including Naive Bayes, k-Nearest Neighbors (KNN), Decision Tree, and Random Forest and particularly in adverse circumstances found in the FaceScrub dataset.

The results of the experiments proved that, although well-known classifiers such as Decision Tree and Random Forest showed high levels of accuracy in the situation with the MUCT dataset, their performance dropped dramatically on the FaceScrub as there was more variability in face pose, illumination, and background. Conversely, the 1D-HD model was relatively stable in the performance, which implies a high degree of generalization and distortion tolerance to real-world conditions. In addition, even though the training durations were increased, especially with large-scale datasets such as FaceScrub, the model showed quick inference performance (≤ 12.4 seconds), which makes it deployable to applications with the real-time demand after training.

One of the strongest aspects of the suggested method is efficiency when it comes to using PCA-conditioned pixel features that seriously minimizes the input size removing a lot of information that is not related to identity. In converting these features to 1D sequence form, the hybrid CNN-LSTM design can learn both local spatial dependencies and contextual relations, and outperforms other designs based only on 2D convolutional networks or pixel inputs, without the need to use these directly.

In spite of its success, the model has certain limits. It takes rather extended training periods, and when implemented with huge datasets. Also, the PCA step can drop certain fine grain details of the face that would be useful during extreme concealment or partial viewing. In addition, color and texture are not used because of preprocessing grayscale conversion.

REFERENCES

- [1] Jain, A.K., Ross, A.A., Nandakumar, K. (2011). Introduction to Biometrics. Springer.
- [2] Abdulhasan, R.A., Abd Al-latief, S.T., Kadhim, S.M. (2024). Instant learning based on deep neural network with linear discriminant analysis features extraction for accurate iris recognition system. *Multimedia Tools and Applications*, 83: 32099-32122. <https://doi.org/10.1007/s11042-023-16751-6>
- [3] Adjabi, I., Ouahabi, A., Benzaoui, A., Taleb-Ahmed, A. (2020). Past, present, and future of face recognition: A review. *Electronics*, 9(8): 1188. <https://doi.org/10.3390/electronics9081188>
- [4] Hummady, G.K., Ahmad, M.L. (2022). A review: Face recognition techniques using deep learning. *Al-Iraqia Journal for Scientific Engineering Research*, 1(1): 1-9.
- [5] Mohammed, S.N., Abdul Hassan, A.K. (2020). Speech

- emotion recognition using MELBP variants of spectrogram image. *International Journal of Intelligent Engineering & Systems*, 13(5): 257-266. <https://doi.org/10.22266/ijies2020.1031.23>
- [6] Jasim, S.S., Hassan, A.K.A. (2022). Modern drowsiness detection in deep learning: A review. *Journal of Al-Qadisiyah for Computer Science and Mathematics*, 14(3): 119-129.
- [7] Badr, A.A., Abdul-Hassan, A.K. (2021). Estimating age in short utterances based on multi-class classification approach. *Computers, Materials and Continua*, 68(2): 1713-1729. <https://doi.org/10.32604/cmc.2021.016732>
- [8] 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>
- [9] Jolliffe, I.T. (2014). Principal Component Analysis. In *International Encyclopedia of Statistical Science*. Springer, Berlin, Heidelberg, pp. 1094-1096. https://doi.org/10.1007/978-3-642-04898-2_455
- [10] Kotsiantis, S.B. (2007). Supervised machine learning: A review of classification techniques. *Informatica*, 31: 249-268.
- [11] Hastie, T., Tibshirani, R., Friedman, J. (2009). *The Elements of Statistical Learning*. Springer.
- [12] Breiman, L. (2001). Random forests. *Machine Learning*, 45: 5-32. <https://doi.org/10.1023/A:1010933404324>
- [13] Sun, Y., Wang, X.G., Tang, X.O. (2014). Deep learning face representation from predicting 10,000 classes. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, pp. 1891-1898. <https://doi.org/10.1109/CVPR.2014.244>
- [14] Hassan, A.K.A., Fadhil, D.J. (2018). Mobile robot path planning method using firefly algorithm for 3D sphere dynamic & partially known environment. *Journal of University of Babylon for Pure and Applied Sciences*, 26(7): 309-320. <https://doi.org/10.29196/jubpas.v26i7.1506>
- [15] Hassan, A.K.A., Fadhil, D.J. (2019). Proposed Modified A* for Three-Dimensional Sphere Environment. *Al-Mansour Journal*, 32(1): 1-13.
- [16] Szegedy, C., Liu, W., Jia, Y.Q., Sermanet, P., Reed, S., Anguelov, D. (2015). Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, pp. 1-9. <https://doi.org/10.1109/CVPR.2015.7298594>
- [17] Zhang, J., Liu, D., Chen, W., Pei, Z., Wang, J. (2022). Deep convolutional neural network for EEG-based motor decoding. *Micromachines*, 13(9): 1485. <https://doi.org/10.3390/mi13091485>
- [18] Kadhim, S.M., Koh Siaw Paw, J., Tak, Y.C., Ameen, S. (2024). Deep learning models for biometric recognition based on face, finger vein, fingerprint, and iris: A survey. *Journal of Smart Internet of Things (JSIoT)*, 2024(1): 117-157. <https://doi.org/10.2478/jsiot-2024-0007>
- [19] Lateef, R.A., Abbas, A.R. (2022). Tuning the hyperparameters of the 1D CNN model to improve the performance of human activity recognition. *Engineering and Technology Journal*, 40(4): 547-554. <http://doi.org/10.30684/etj.v40i4.2054>
- [20] Kadhim, S.M., Paw, J.K.S., Tak, Y.C., Ameen, S., Alkhayyat, A. (2025). Deep learning for robust iris recognition: Introducing synchronized spatiotemporal linear discriminant model-iris. *Advances in Artificial Intelligence and Machine Learning*, 5(1): 3446-3464. <https://doi.org/10.54364/AAIML.2025.51197>
- [21] Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8): 1735-1780. <https://doi.org/10.1162/neco.1997.9.8.1735>
- [22] Hussien, R.M., Al-Jubouri, K.Q., Al Gburi, M., Qahtan, A.G.H., Jaafar, A.H.D. (2021). Computer vision and image processing the challenges and opportunities for new technologies approach: A paper review. *Journal of Physics: Conference Series*, 1973: 012002. <https://doi.org/10.1088/1742-6596/1973/1/012002>
- [23] Al Baiati, A.E., Al Gburi, H.Q., Al Hamami, D.J. (2020). Securing software defined network transactions using visual cryptography in steganography. *Periodicals of Engineering and Natural Sciences*, 8(4): 2405-2416. <https://doi.org/10.21533/pen.v8i4.1737>
- [24] Abdulhadi, M.T., Abbas, A.R. (2025). Human action behavior recognition in still images using ensemble deep learning approach. *AIP Conference Proceedings*, 3169(1): 030014. <https://doi.org/10.1063/5.0254242>
- [25] Zhao, W., Chellappa, R., Phillips, P.J., Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM Computing Surveys (CSUR)*, 35(4): 399-458. <https://doi.org/10.1145/954339.954342>
- [26] Kashkool, H.J.M., Farhan, H.M., Naseri, R.A.S., Kurnaz, S. (2024). Enhancing facial recognition accuracy and efficiency through integrated CNN, PCA, and SVM techniques. In *International Conference on Forthcoming Networks and Sustainability in the AIoT Era*, pp. 57-70. Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-62871-9_6
- [27] Komlavi, A.A., Chaibou, K., Naroua, H. (2024). Comparative study of machine learning algorithms for face recognition. *Revue Africaine de Recherche En Informatique et Mathématiques Appliquées*, 40. <https://doi.org/10.46298/arima.9291>
- [28] Schroff, F., Kalenichenko, D., Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, pp. 815-823. <https://doi.org/10.1109/CVPR.2015.7298682>
- [29] Sandler, M., Howard, A., Zhu, M.L., Zhmoginov, A., Chen, L.C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, pp. 4510-4520. <https://doi.org/10.1109/CVPR.2018.00474>
- [30] Azeez, A. (2025). Automated emotion recognition using hybrid CNN-RNN models on multimodal physiological signals: Automated emotion recognition using hybrid CNN-RNN models on multimodal physiological signals. *AlKadhim Journal for Computer Science*, 3(2): 20-29. <https://doi.org/10.61710/kjcs.v3i2.100>
- [31] Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., Lew, M.S. (2016). Deep learning for visual understanding: A review. *Neurocomputing*, 187: 27-48. <https://doi.org/10.1016/j.neucom.2015.09.116>
- [32] Afouras, T., Chung, J.S., Zisserman, A. (2018). The Conversation: Deep audio-visual speech enhancement. *arXiv preprint arXiv:1804.04121*. <https://doi.org/10.48550/arXiv.1804.04121>
- [33] Ding, X., Zhang, Y., Li, G., Gao, X., Ye, N., Niyato, D., Yang, K. (2023). Few-shot recognition and classification framework for jamming signal: A CGAN-based fusion

- CNN approach. arXiv preprint arXiv:2311.05273. <https://doi.org/10.48550/arXiv.2311.05273>
- [34] Paul, S., Acharya, S.K. (2020). A comparative study on facial recognition algorithms. In e-journal-First Pan IIT International Management Conference–2018. <https://doi.org/10.2139/ssrn.3753064>
- [35] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I. (2017). Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17), Red Hook, NY, USA, pp. 6000-6010.
- [36] Chung, J., Gulcehre, C., Cho, K., Bengio, Y. (2015). Gated feedback recurrent neural networks. In Proceedings of the 32nd International Conference on Machine Learning (ICML), pp. 2067-2075.
- [37] Viola, P., Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Kauai, HI, USA, pp. I-511-I-518. <https://doi.org/10.1109/CVPR.2001.990517>
- [38] Sokolova, M., Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4): 427-437. <https://doi.org/10.1016/j.ipm.2009.03.002>
- [39] Kadhim, S.M., Siaw Paw, J.K., Tak, Y.C., Al-Latief, S.T.A. (2025). Robust security system: A novel facial recognition optimization using coronavirus-inspired algorithm and machine learning. *Iraqi Journal for Computer Science and Mathematics*, 6(2): 19. <https://doi.org/10.52866/2788-7421.1260>
- [40] Milborrow, S. (2010). The MUCT Landmarked Face Database.
- [41] Ng, H.W., Winkler, S. (2014). A data-driven approach to cleaning large face datasets. In 2014 IEEE International Conference on Image Processing (ICIP), Paris, France. pp. 343-347. <https://doi.org/10.1109/ICIP.2014.7025068>
- [42] Parkhi, O., Vedaldi, A., Zisserman, A. (2015). Deep face recognition. In BMVC 2015 - Proceedings of the British Machine Vision Conference, pp. 1-12.
- [43] Krizhevsky, A., Sutskever, I., Hinton, G.E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6): 84-90. <https://doi.org/10.1145/3065386>