Journal of Scientific & Industrial Research
Vol. 83, April 2024, pp. 414-423
DOI: 10.56042/jsir.v83i4.3167

A Hybrid Approach based on Haar Cascade, Softmax, and CNN for Human Face Recognition

Pancham Singh[1], Mrignainy Kansal[1], Rajeev Singh[2], Sushil Kumar[3]* & Chelsi Sen[1]

[1]Department of Information Technology, Ajay Kumar Garg Engineering College, Ghaziabad 201 015, Uttar Pradesh, India
[2]Department of Information Technology, KIET Group of Institutions, Delhi-NCR, Ghaziabad 201 206, Uttar Pradesh, India
[3]Department of Computer Science and Engineering, KIET Group of Institutions, Delhi-NCR, Ghaziabad 201 206, Uttar Pradesh, India

Received 24 June 2023; revised 13 January 2024; accepted 09 March 2024

Face recognition has been studied long but it is still an important and current research field in deep learning, computer vision, and forensics. There are several applications such as group action systems, human-machine interaction, and security systems, where face recognition is of vital importance. It is noticed that the algorithms based on Deep Learning (DL) have shown higher performances, stipulation of accuracy, and processing speed as compared to traditional machine learning algorithms. With its dominant methodology in deep learning, the Convolutional Neural Network (CNN) has contributed immensely to face recognition. In this paper, a novel hybrid version of the deep learning algorithm containing Haar Cascade, SoftMax, and CNN components is proposed. It provides promising results for applications based on the recognition of human faces. In the experiments, the accuracy of this hybrid algorithm is achieved at 99.95%, which is significantly higher than existing Viola-Jonas and Principal Component Analysis (PCA), which have accuracy rates of 74.38% and 81.81% respectively. However, the accuracy of our proposed algorithm close to Linear Discriminant Analysis (LDA) at 95.45%, and SoftMax and CNN at 94%. In this paper, the proposed hybrid deep learning algorithm improves the result performance and is compared with some existing techniques for face recognition.

Keywords: Biometric system, Computer vision, Linear discriminant analysis, Principal component analysis, Viola-Jonas

Introduction

The field of face recognition research poses significant challenges due to the variability in facial appearances, the effects of illumination, and the complexities of image backgrounds.[1] These challenges are present in both real-world and captured images, as well as those obtained from sensing elements and digital sources. Despite these challenges, there is a growing demand for efficient and automatic face recognition systems in various applications such as security, psychology, image processing, and computer vision. However, even with extensive research in this field, there are still many unresolved challenges, particularly in addressing the increased recognition time when the scale of face information increases. Despite these challenges, face recognition remains a highly effective method, characterized by its high accuracy and low invasiveness. Using this method, we analyze the attributes of an individual's face having been captured by a digital video cameras or online face capturing devices which is crucial for ensuring higher levels of security and comprehensive security solutions.

Artificial intelligence and computer vision are closely intertwined, with computer vision focused on replicating human vision through electronic means. This technology not only allows for the perception and interpretation of images but also the ability to detect, identify, and process those images like human vision. One specific application of computer vision is face recognition, which utilizes biometric methods to identify individuals based on the unique characteristics of their faces. While human eyes can recognize people, the limitations of human concentration span necessitate the development of a computerized face recognition method.[2] Identity fraud, hacking, and security breaches are all examples of the various ways that criminals can access personal information and financial accounts without authorization. These types of crimes often involve the use of stolen identification cards, keys, passwords, and PIN codes to gain access to sensitive information. However, with the advancement of technology, there are now systems in place that can verify a person's true identity through the use of biometrics. It is the process by which computer technology helps us to identify

*Author for Correspondence
E-mail: drsushil.cs@gmail.com

individuals based on their unique characteristics such as fingerprints, eye patterns, facial features, and voice recognition. One example of this technology is the use of face recognition systems which have four main components: face detection, pre-processing, feature extraction, and face recognition. These systems can greatly reduce the risk of fraud and unauthorized access to personal information. The main contributions of this paper are addressed as follows:

1) A model based on the convolutional neural network (CNN), using the FER-2013 dataset is used for the recognition of faces since it is fast and has high precision in face recognition.

2) On providing an input image the system detects the face using Haar cascade and SoftMax classifier.

3) Normalize between 0 and 1 and reshape the data before splitting the dataset.

4) Batch normalization can be used in the proposed model to improve training and validation accuracy.

5) After every max-pooling layer and fully connected layer in the proposed model, a dropout layer with a coefficient of 0.5 is added to overcome the overfitting issues.

6) Experimental results evaluate the level of accuracy when comparing different face recognition methods.

Several automatic facial recognition systems have been developed, and several facial recognition algorithms have been proposed. Linear Discriminant Analysis (LDA), Principal Component Analysis,[3] Viola Jonas, and CNN are among them. The sections below discuss the approximate accuracy and limitations of the algorithms proposed and CNN's specifically as the algorithm has gained considerable attention in recent years as the most precise algorithm available.[4]

Literature Survey

A lot of work has been pursued in the area of face recognition but still, it is a research field. Bruner and Tagiuri conducted the first research on facial recognition in psychology in the early 1950s, followed by engineering research in the 1960s.[5] In 1972, the first studies in this discipline focused on emotional facial expressions.[6,7] However, research on automatic face recognition began in 1973, and numerous ideas have been proposed by scholars in recent years using tiny datasets. Earlier in the 1970's, pattern classification techniques were used for facial recognition.[5] For the past 15–20 years research on face recognition using cameras had been done extensively. Recognizing a 3D object from a 2D images have been considered a challenging task. In 1995, a survey of FRT was given in a review paper. Initially, video-based face recognition was taken into consideration. Since then, as technology has advanced, numerous algorithms have been presented to produce considerably more efficient and accurate face recognition results. New datasets have been created to test these face recognition algorithms. Several algorithms that are used for face recognition include PCA,[8] LDA, Viola Jones, and neural network-based algorithms such as CNN, etc. Above all mentioned algorithm PCA was once considered a very popular algorithm because of its easy implementation.[9] PCA is a dimensionality reduction method that offers an accuracy of 85%. Kirby and Sirovich were the researchers who applied PCA in 1990 to represent faces and similarly, Turk and Pentland 1991 applied the same algorithm to recognize faces.[9] After extensive research, several drawbacks have been found while working with PCA. One of the major drawbacks is that it is an arduous task to evaluate the covariance matrix accurately. Also, PCA is barely sensitive to non-identical training datasets. Sometimes it is very challenging to capture even the simplest invariance. Another algorithm is LDA, which uses a set of projected vectors to produce results with 80% accuracy. LDA is also known as the Fischer face method. This algorithm has a limitation of small sample size.[10–12] Computational difficulties arise during the execution of LDA which does not make it an efficient algorithm. Like this algorithm, the other mentioned algorithms also have some limitations. Considering all these aspects the most efficient and reliable algorithm can be CNN. As we already discussed, along with CNN face recognition can be done using algorithms such as PCA, LDA, etc. These are used on a large scale in the present time also because, to the present date, the computational powers of computers are not as fast to make decisions as humans. But the CNN algorithm makes it possible. Face Recognition with CNN is much more reliable and efficient than other ML algorithms. It has some features which reduce the computational work and make it time efficient. Also, CNN has a high accuracy rate. This is the reason why CNN is popular in the present scenario of AI and ML. Some important literature reviews in this area are reported in Table 1.

In this field, different proposed techniques are compared and analyzed based on the detection of facial emotions. It reports the existing works

Table 1 — Some important literature reviews for face recognition

Methodology	Findings	Limitations
LDA[10]	Not effective for the problem having a small sample size using LDA. Therefore, a new LDA-based face recognition process has been proposed which uses a within-class scatter matrix to calculate the projection vectors in the null space.	The main problem that arises with this new proposed LDA-based algorithm is an intrinsic overfitting problem which is how to distinguish between the noise subspace and the true in-class variation subspace.
LDA[11]	This paper proposed a new feature extraction process for recognizing human faces. The new method introduced here uses a new variant of LDA and it is commonly considered as a general technique. The proposed method is DF-LDA which is a linear pattern recognition methodology that solves the overfitting problem.	The weighted scatter matrix between the classes, although classes clustered in the input space should have more weight in the input space as this is more likely to lead to miss classification. Eigenvectors of the positive Eigenvalues of the weighted between-class scatter, which typically include considerable discriminatory information, will ignore the null space of the training sets within-class scatter matrix.
Extended PCA[8]	Extended-PCA algorithms including 2DPCA, KPCA, IKPCA, ESKPCA, and incremental PCA (IPCA) have been detailed. On comparing different algorithms via face recognition experiments, it is found that 2D PCA is computationally more efficient than others.	PCA on facial images is typically time-consuming to implement. For tiny face databases, the 2D PCA requires a bit longer recognition time than any other traditional recognition model. PCA takes into consideration of a full-frontal view of the subject's face, which is uncommon in real-time situations. One of the greatest issues with the PCA approach is the within-class weak discrimination power.
PCA[3]	This study demonstrates that the PCA approach has the highest level of accuracy when all images are captured correctly and routinely, without any qualities, and with identical lighting and background conditions.	PCA demands highly labeled data for high accuracy, means every image should have the same attributeslike lighting, background, etc.
CNN[2]	In order to control the spread of the covid-19, this high precision face mask recognition model has been built which uses Sklearn, a deep learning CNN model to identify whether a person is wearing a mask or not, even in the live video, and can be easily applied at the workplaces.	It can only be used in high-definition camcorders. It is not suitable for the very high volume of the data, as with the increasing volume the efficiency of the model will decrease.
CNN and biLSTM[7]	After the augmentation of the images in the ck+ datasets, the BiLSTM-CNN model gave an accuracy about 99.43%, hence it can handle the overfitting case and improve the performance of the trained model.	The dataset consists of 981 images which is quite small and hence it is still overfitted. It requires more training time; therefore, it is slow.
Haar Cascade classifier and CNN[6]	CNN are the most efficient and also time taking model for image processing, which has been used to detect facial emotions and hence received 65.59% validation accuracy at the 105th epoch and validation losses are as 0.6 and 1.0.	It requires more training in order to provide the correct results, hence also time-consuming.
Hybrid CNN and ConvLSTM[13]	This paper proposes a novel hybrid model combining CNN and ConvLSTM for video-based facial expression recognition (VFER). The approach involves input pre-processing, alignment, scaling, and cropping of facial images before classification. It utilizes different datasets like CK+, SAVEE, and AFEW, achieving over 95% accuracy in a tenfold subject-independent cross-validation on CK+.	The limitations include challenges in recognizing emotions with limited data (e.g., contempt, fear, sadness) and confusion between certain emotions. The paper doesn't extensively discuss the computational complexity, training efficiency, or scalability of the proposed model. The methods gave lack comprehensive analysis of model efficiency.
im-cGAN[14]	The proposed model fuses global and local features, utilizing an improved conditional generative adversarial network (im-cGAN) for data augmentation. im-cGAN leverages action units (AUs) to control facial muscle movements hence generating labelled expression samples with fine-grained changes. It proposes a D-loss function to enhance feature discrimination.	Recognizing expressions with subtle muscle deformations (e.g., fear) poses challenges, and misclassifications may occur due to similarities in facial movements.
Deep CNN with bilinear pooling[15]	The paper introduces two feature representation schemes, a basic convolutional neural network (CNN) architecture and a deep CNN architecture with bi-linear pooling, matrix normalization, and dense layers. These schemes are designed to capture complex facial features and enhance the performance of the recognition system by addressing challenges like noise, illumination, blurriness, and motion-blur.	The paper mentions the implementation environment but due to lacks details on how the proposed method would perform in different hardware or software setups, limiting its generalizability. The generalization of the proposed approach to broader datasets is not extensively discussed.
PACVT[16]	The patch attention convolutional vision transformer (PACVT) is a novel FER approach in challenging scenarios with occlusions. The model effectively addresses occlusion-related challenges by dynamically weighting local features and capturing long-distance relationships between facial patches through self-attention.	The model's performance may vary based on the types and severity of occlusions, and its generalization to diverse real-world scenarios warrants further exploration. The complexity analysis indicates an increase in parameters and training time compared to certain baselines, suggesting a trade-off between performance and efficiency.
POSTER Technique[17]	The paper proposes POSTER++, an enhanced version of FER model POSTER. It enhances computational performance in POSTER by changing the two-stream design, cross-fusion, and multi-scale feature extraction techniques. The model greatly reduces the number of parameters and floating-point operations, indicating a better balance of accuracy and computing complexity.	The paper focuses on benchmark datasets, and the generalization of POSTER++ to diverse real-world scenarios or challenging conditions is not extensively explored.

comparing techniques, solutions to problems, and comparing different techniques with each other. This paper describes the method of face recognition using a deep learning algorithm known as CNN. A CNN is a type of network that forwards its output to the next layers of the network it means every output works as input for the further layers until it reaches its final state and this forwarding process is conventionally used to analyze visual images by processing data coming from layer-wise like a grid topology that is why it is also said as convolutional networking.[18,19]

- CNN has completely changed our methodology towards image identification as CNN can recognize patterns and make sense of them. CNN is the most efficient algorithm for image classification, retrieval, and identification tasks as it provides higher accuracy with less execution time.
- CNN provides high-accuracy results in identifying where in an image a particular thing exists. This aspect has made CNN the go-to method for predictions involving any image as input.
- A key feature of CNN is that this algorithm can achieve "Spatial Invariance" which means that it can recognize and extract image features without any manual intervention. CNN learns and does evaluation by itself from image/data and further performs extraction of features from image/data. It is a Deep Learning technique that produces precise results.[20–23]

The crucial objective of the pooling layer is to reduce the dimensions of pixels coming from the convolutional process. Here this layer tries to extract features that are more important to recognize an image. This layer also reduces the time and cost of the recognition process.[24] This image summarizes the main image and feeds the generated output into the fully connected layer.

Proposed Methodology

The proposed model consists of Haar Cascade, SoftMax classifier and a Convolutional Neural Network (CNN) used to detect face recognition. First, the proposed system detects faces from a given input images using Haar cascade and SoftMax classifier, then crops the faces, thereby normalizing the image to (48 × 48) pixels. The input is then sent through the CNN and a single class (0–6) is generated. Following the input and output, denoted 0-"Angry", 1-"Disgust", 2-"Fear", 3-"Happy", 4-"Sad", 5-"Surprised", 6-"Neutral". A CNN is used to preprocess and recognize visual images.

Table 2 — Details of different Haar Cascade classifiers with label, stage, and size

Haar Cascade Classifier	Label	Stage	Size
HaarCascade frontal face default	HD	25	24×24
HaarCascade frontal face alt	HT	21	20×20
HaarCascade frontal face alt2	HT2	20	20×20
Haar Cascade frontal face alt tree	HTR	46	20×20
Haar Cascade profile face	HPF	25	20×20

Haar Cascade, SoftMax, and CNN Layered Architecture:

Input Layer

The input layer is the first layer of the CNN-layered architecture. In this layer, only the user inputs their image/data to be identified.[25,26]

Haar Cascade Classifier

This cascading classifier is considered one of the most effective object detection methods that are similar to Haar to detect faces. The Haar-like features have various characteristics. The details of several Haar cascade classifiers with label, stage, and size are reported in Table 2. While the "line feature" and "rectangle feature" are used to identify the slanted line of the item, the "edge feature" is used to define edges around the object. The largest computation will be shown when using integral pictures. The specific Haar features of a face is represented by the algorithm. The algorithm detection turns the input image, which contains numerous faces, into a (24 × 24) window before pixel-by-pixel analyzing each Haar characteristic of that window. The classifier must be trained on the two features of the detection algorithm's images (positive or negative). Positive aspects of an image refer to images that include faces, whereas negative aspects of an image refer to images that do not include faces. Haar cascade frontal face is used for face detection.[6] For the Haar Cascade layer, let $h(x)$ be the Haar-like feature extraction function and $w_h$represents the weights associated with these features. The output of the HaarCascade layer can be represented as:

$$Haar_{Cascade_{Output}} = \sum_h w_h \, . \, h(x) \qquad \ldots (1)$$

SoftMax Classifier

The SoftMax function converts a vector of K real values into a vector of K real values that sums to 1. It converts the input values, which can be more than 0, less than zero, zero, or greater than one, into probabilities between 0 and 1, and identifies them as such. It converts small or negative inputs into small

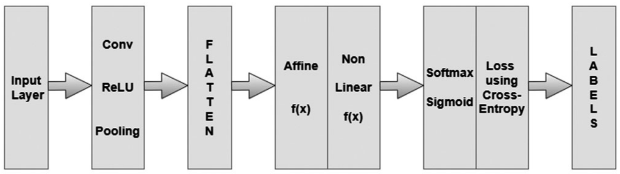

Fig. 1 — Processes of CNN algorithm

probabilities and large or positive inputs into large probabilities, but it remains in the range of 0 and 1. Assuming z_j as the input to the Softmax layer, the Softmax function for jth class is defined as:

$$P(Class_j) = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \qquad \dots (2)$$

Convolutional Layer

This is the second layer of the CNN-layered architecture. It is the most important layer of the recognition process as shown in Fig. 1. The convolutional layer is the main building element of CNN that handles the majority of the computational hard lifting. This layer extracts all the important features from an image which is important for the identification of any images. On this layer, the image is considered as a pixel matrix or we can say here all the images are converted according to their pixel matrices. CNN has multiple filters that perform convolution processes. In the convolution process, we slide the filter matrix over the image pixel matrix and calculate the dot product to find the convolved feature matrix.[20,27] Let x_i represent the input to the CNN layer, W_i denote the weight matrix, and b_i be the bias term for the ith layer. The output of the CNN layer is given by the convolution operation followed by activation function σ:

$$y = \sigma(W_i * x_i + b_i) \qquad \dots (3)$$

ReLU Layer

It is usually, for the normalization process, we use the ReLU function. ReLU stands for Rectified Linear Unit. ReLU Layer takes the input from the convolutional layer and calculates the maximum valued feature matrix.[20,21] ReLU is a non-linear operation made up of rectifier-using components. Since it is an element wise operation, each pixel is affected and all negative values in the feature map are replaced with zero. This means that all negative values are turned into zero using the ReLU function.[24–26,28]

Pooling Layer

Pooling is a concept where the pooling function reduces the dimensions of each activation map of the feature rectified image and generates a new reduced feature image or pooled image. But it does not reduce the features which are more important to identify the image.

Fully Connected Layer

A Fully Connected Layer (FCL) is the last layer of the CNN model. This layer takes input from the pooling layer and this input provided by the pooling layer is in the form of the flatten. FCL is a combination of affine functions and nonlinear functions. FCL is connected to each filter of its previous and after layers. Output coming from the pooling layer is firstly fed into a flattened layer where output is converted into one-dimensional form and then it is fed into the FCL.

Output and Error Layer

Here, CNN generates the output of the given input image/data with the accuracy and error chances or error percentage. The output layer is a combination of Softmax or sigmoid functions and cross-entropy loss functions. The sigmoid function is used to calculate image recognition probability because it consists of the value between 0 and 1 and the error is computed by the cross-entropy function.

Input Image from the Camera

This is the starting point of your facial detection process. Capturing images from a camera is a standard practice for real-time applications.

Image Detection

This step involves identifying Regions of Interest (ROIs) in the input image.[29] This can be done using techniques like object detection to locate potential faces in the image.

Haar Cascade, Softmax and CNN

Integrating Haar Cascade, Softmax, and CNN layers is a hybrid approach that combines classical feature-based methods (Haar Cascade) with deep learning techniques (CNN and Softmax). This approach can leverage the strengths of both methods for improved accuracy. For each detected ROI r_i, apply the hybrid approach of Haar Cascade, Softmax, and CNN:

$$CNN\_Output_{r_i} = \sigma\left(W_{CNN} * I_{r_i} + b_{CNN}\right)$$

$$Haar\_Cascade\_Output_{r_i} = W_{Haar} . F_{r_i}$$

$$Hybrid\_Output_{r_i} = Softmax\left(Haar_{Cascade_{Output_{r_i}}} \otimes CNN_{Output_{r_i}}\right) \qquad \dots (4)$$

where, $\otimes$represents the fusion operation.

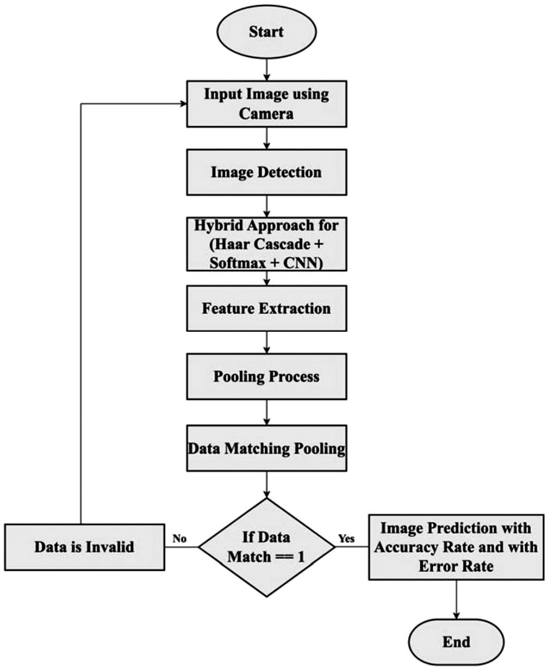

Fig. 2 — Face recognition using Hybrid approach for Haar cascade, SoftMax and CNN flowchart

Feature Extraction

Extracting relevant features from the regions identified in the previous step is crucial for building a robust representation of facial characteristics. Features may include facial landmarks, textures, or other discriminative information.[30] For each detected ROI, extract relevant features:

$$Feature_{r_i} = ExtractFeatures\left(Hybrid_{output_{r_i}}\right) \quad \ldots (5)$$

Pooling Process

Pooling is often used to aggregate features and reduce dimensionality. Common pooling techniques include max pooling or average pooling. Apply a pooling process to aggregate features:

$$Pooled_{Feature} = Pool\left(Features_{r_i}\right) \quad \ldots (6)$$

The steps as shown in Fig. 2 help to understand the processes of proposed algorithm. The pseudo code of proposed method is also presented in Algorithm "Hybrid Facial Detection"

Algorithm: Hybrid Facial Detection

Input: Real-time image stream from camera

Output: Facial detection results, accuracy, and error rate

```
procedure FacialDetection():
while True:
# Capture image from the camera
input_image = capture_image_from_camera()
# Image Detection
detected_regions = detect_regions(input_image)
# Loop over detected regions
for region in detected_regions:
# Haar Cascade + Softmax + CNN
cnn_output = apply_cnn_with_haar_softmax(region)
# Feature Extraction
features = extract_features(cnn_output)
# Pooling Process
pooled_features = pool(features)
# Data Matching Pooling
matching_result = match_data_pooling(pooled_features, dataset)
# Decision Process
if not matching_result:
# No match, data is invalid
print("Data is invalid. Capturing a new image.")
continue
else:
# Match found, predict accuracy and error rate
prediction_result = predict(matching_result)
accuracy, error_rate = compute_accuracy_error_rate(prediction_result)
# Call the FacialDetection procedure to start the process
FacialDetection()
```

Data Matching Pooling

Matching the extracted features with a pre-existing dataset is a key step for recognizing individuals. Techniques like similarity measures or machine learning classifiers can be employed for this matching process. The pooled features are compared with a pre-existing dataset to find a match using.

$$Matching_{Result} = Match(Pool_{Features}, Dataset) \quad \ldots(7)$$

Decision Process

Based on the matching results, a decision is made. If a match is found, the system proceeds to predict accuracy and error rates. If no match is found, it indicates that the captured data may not correspond to a known individual, and the system goes back to capture a new image. If no match is found, go back to capture an image. If a match is found, then the prediction is computed as:

$$Prediction_{Result} = Predict(Matching_{Result}) \quad \ldots(8)$$

Data Predict Accuracy and Error Rate

If a match is found, the system can predict accuracy and error rates based on the recognition results. This step provides quantitative measures of the model's performance. Overall, the described flow is logical and aligns with the typical steps involved in facial detection and recognition systems. Keep in mind that the success of your approach will depend on the specific implementation details, the quality of your dataset, and the performance of the chosen algorithms. The accuracy and error rates based on the prediction result and ground truth are computed as:

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \quad \ldots (9)$$

$$Error\ Rate = 1 - Accuracy \quad \ldots (10)$$

Results Analysis

Analysis of results on facial dataset has been done using various face recognition methods. Some faces are demonstrated in Fig. 3. After analysis of a facial dataset using various above-mentioned algorithms, the predicted results are shown in Fig. 4. Out of all algorithms, the highest accuracy comes out using the hybrid model Haar Cascade, Softmax, and CNN i.e., 99.95%.

Result Analysis

Results using Principal Component Analysis (PCA)

The dataset comprises a collection of facial images, explicitly consisting of 10 distinct individuals. Each individual's face is represented by 6 unique images, resulting in a total of 60 images in the training dataset. For the testing dataset, we have a total of 44 images. Among these, 8 faces are represented with 4 images each, while 2 faces have 6 images each. This dataset is the foundation for our analysis and experimentation, allowing us to investigate various aspects of facial recognition, identification, or any other relevant research objectives. According to the results depicted in Fig. 4a, the accuracy of our face recognition algorithm is determined to be 81.81% and its accuracy graph is shown in Fig. 5a.

Results using Linear Discriminant Analysis (LDA)

The dataset is composed of facial images featuring 10 distinct individuals. Each individual's face is captured by 6 unique images, resulting in a total of 60 images in the training dataset. In the testing dataset, there are a total of 44 images, with 8 faces represented by 4 images each and 2 faces represented by 6 images each. This dataset serves as the cornerstone for our analysis and experimentation, enabling us to explore various aspects of facial recognition, identification, and other pertinent research objectives. As per the results presented in Fig. 4b, LDA based face recognition algorithm attains an accuracy of 95.45%, with the corresponding accuracy graph illustrated in Fig. 5b.

Results using Softmax and CNN

The facial dataset comprises 10 distinct individuals, with each individual having 6 unique images, resulting in a total of 60 images in the training dataset. In the testing dataset, 44 images are utilized, with 8 faces represented by 4 images each,

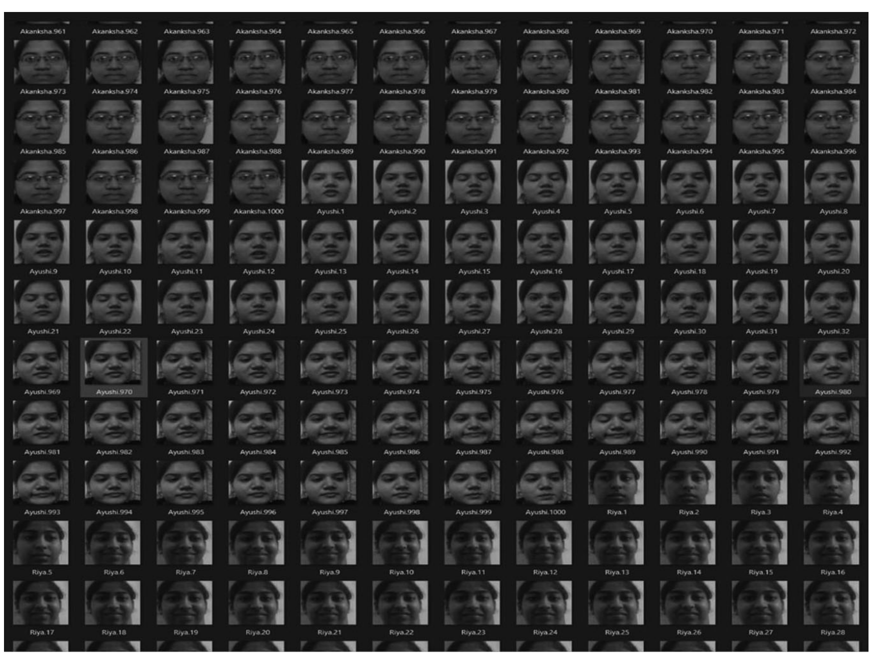

Fig. 3 — Facial dataset sample

Fig. 4 — Predicted results using: (a) PCA, (b) LDA, (c) Softmax + CNN, and (d) proposed Haar Cascade + Softmax + CNN algorithms, respectively

and 2 faces represented by 6 images each. This dataset serves as the basis for our analysis and experimentation, enabling the exploration of various facets of facial recognition, identification, and other pertinent research objectives. As illustrated in Fig. 4c, Softmax and CNN based face recognition algorithm achieves an accuracy of 94%, and the corresponding accuracy graph is presented in Fig. 5c.

Results using Haar Cascade, Softmax, and CNN

The dataset used in this algorithm includes 3000 training images and 60 testing images. We have focused on three different individuals in the dataset because of the large storage required for storing these images. For training, each individual has 1000 images, resulting in a total of 3000 training images. This extensive training dataset covers a wide range of poses, facial expressions, and lighting conditions to ensure comprehensive model training. For testing, we have allocated 20 images per individual, amounting to a total of 60 testing images. Overall, the dataset provides a solid foundation for developing and accessing the performance of our face recognition algorithm. According to the results depicted in Fig. 4d, the accuracy of hybrid (Haar Cascade, Softmax, and CNN) face recognition algorithm is determined to be nearly 99.95% and its accuracy

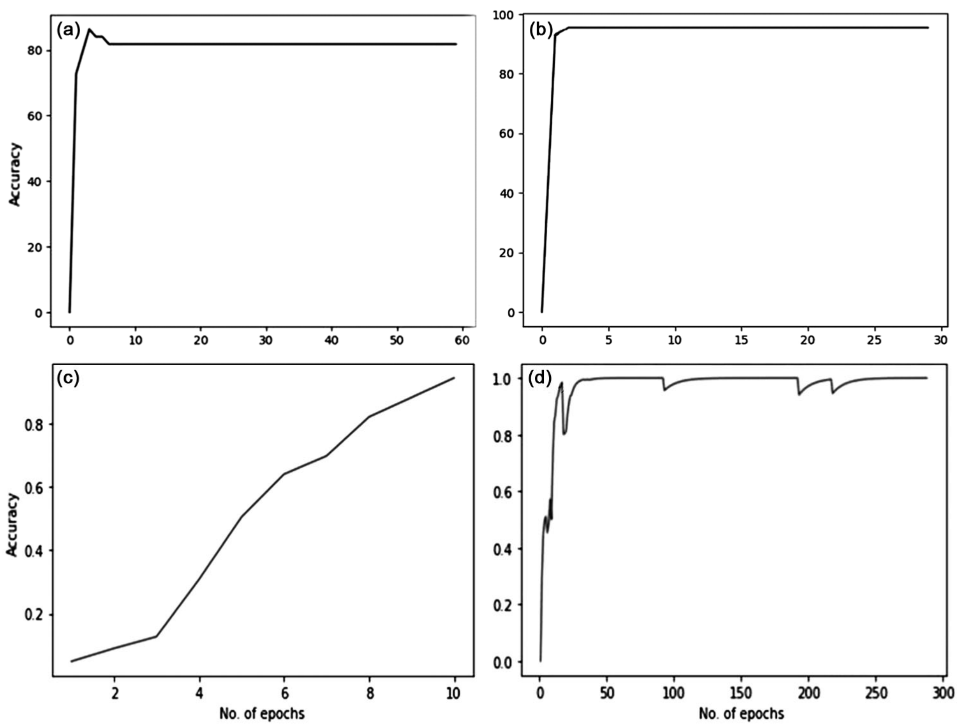

Fig. 5 — Accuracy graphs using: (a) PCA, (b) LDA (c) Softmax + CNN, and (d) proposed Haar Cascade + Softmax + CNN algorithms, respectively

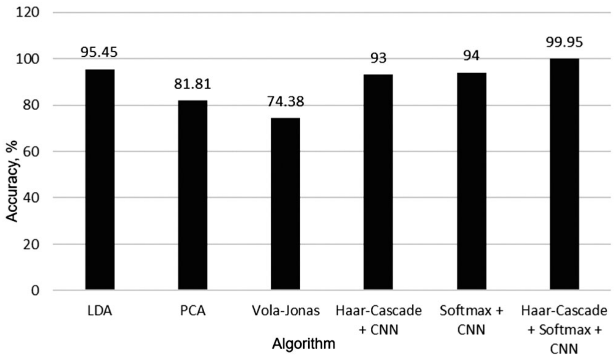

Fig. 6 — Accuracy comparisons graph of all the algorithms

graph is presented in Fig. 5d. The accuracy result of the proposed method is compared with PCA, LDA, Softmax + CNN reveals better, which is demonstrated in Fig. 6.

Conclusions

This paper proposes a face recognition system using novel hybrid CNN, Softmax, and Haar Cascade techniques. The proposed method is compared with the existing techniques and the experimental findings show that the proposed hybrid algorithm performs exceptionally well, with an overwhelming accuracy of 99.95%. The proposed technique successfully combines the strengths of Haar Cascade, Softmax, and CNN, resulting in improved feature extraction and classification capabilities. To enhance improvement of feature extraction methods, the hybrid algorithm could potentially be undertaken. Investigating novel methods for capturing intricate facial details and patterns could contribute to even higher accuracy rates. Extensive testing under diverse conditions such as varying lighting, facial expressions, and potential barrier will enhance the algorithm's robustness. Research efforts can be directed toward minimizing processing time and resource requirements, making the algorithm suitable for deployment in time-sensitive environments. In facial recognition technology, it can be used to solve ethical and privacy issues. Future research should concentrate on creating systems to guarantee accountability, preventing misuse or bias in the deployment of facial recognition systems.

References

1 Kaur P, Krishan, K, Sharma S K & Kanchan T, Facial-recognition algorithms: A literature review, *Med Sci Law*, **60(2)** (2020) 131–139, doi: 10.1177/0025802419893168.

2 Kaur G, Sinha R, Tiwari P K, Yadav S K, Pandey P, Raj R, Vashisth A & Rakhra M, Face mask recognition system using CNN model, *Neurosci Informatics*, **2(3)** (2022) 100035, https://doi.org/10.1016/j.neuri.2021.100035.

3 Maheswari V, Sari C A & Rachmawanto E H, Study analysis of human face recognition using principal component analysis, *IEEE Explore*, (2020) 55–60.

4 Garg K, Verma K, Patidar K & Tejra N, Convolutional neural network based virtual exam controller, *IEEE Explore*, (2020) 895–899.

5 Zhao W, Chellappa R, Phillips P J & Rosenfeld A, Face recognition: a literature survey, *ACM Comput Surv*, **35(4)** (2003) 399–458, https://doi.org/10.1145/954339.954342.

6 Rasheed L, Khadam U, Majeed S, Ramzan S, Bashir M S & Iqbal M M, Face recognition emotions detection using haar cascade classifier and convolutional neural network, (2022), https://doi.org/10.21203/rs.3.rs-2048290/v1.

7 Febrian R, Halim B M, Christina M, Ramdhan D & Chowanda A, Facial expression recognition using bidirectional LSTM-CNN, *Procedia Comput Sci*, **216** (2023) 39–47, https://doi.org/10.1016/j.procs.2022.12.109.

8 Li L, Liu S, Peng Y & Sun Z, Overview of principal component analysis algorithm, *Optik*, **127(9)** (2016) 3935–3944, https://doi.org/10.1016/j.ijleo.2016.01.033.

9 Moon H & Phillips P J, Computational and performance aspects of PCA-based face-recognition algorithms, *Perception*, **30(3)** (2001) 303–321.

10 Chen L F, Liao H Y M, Ko M T, Lin J C & Yu G J, A new LDA-based face recognition system which can solve the small sample size problem, *Pattern Recognit*, **33(10)** (2000) 1713–1726, https://doi.org/10.1016/S0031-3203(99)00139-9.

11 Lu J, Plataniotis K N & Venetsanopoulos A N, Face recognition using LDA-based algorithms, *IEEE Trans Neural Netw*, **14(1)** (2003) 195–200, https://doi.org/10.1109/TNN.2002.806647.

12 He Z, Wu M, Zhao X, Zhang S & Tan J, Representative null space LDA for discriminative dimensionality reduction, *Pattern Recognit*, **111** (2021) 107664, https://doi.org/10.1016/j.patcog.2020.107664.

13 Singh R, Saurav S, Kumar T, Saini R, Vohra A & Singh S, Facial expression recognition in videos using hybrid CNN & Conv LSTM, *Int J Inf Technol*, **15(4)** (2023) 1819–1830, https://doi.org/10.1007/s41870-023-01183-0.

14 Sun Z, Zhang H, Bai J, Liu M & Hu Z, A discriminatively deep fusion approach with improved conditional GAN (im-cGAN) for facial expression recognition, *Pattern Recognit*, **135** (2023) 109157, https://doi.org/10.1016/j.patcog.2022.109157.

15 Hossain S, Umer S, Rout R K & Tanveer M, Fine-grained image analysis for facial expression recognition using deep convolutional neural networks with bilinear pooling, *Appl Soft Comput*, **134** (2023) 109997, https://doi.org/10.1016/j.asoc.2023.109997.

16 Liu C, Hirota K & Dai Y, Patch attention convolutional vision transformer for facial expression recognition with occlusion, *Inf Sci*, **619** (2023) 781–794, https://doi.org/10.1016/j.ins.2022.11.068.

17 Mao J, Xu R, Yin X, Chang Y, Nie B & Huang A, POSTER V2: A simpler and stronger facial expression recognition network, *arXiv*, (2023) preprint arXiv:2301.12149, https://doi.org/10.48550/arXiv.2301.12149.

18 Tombe R & Viriri S, Effective processing of convolutional neural networks for computer vision: A tutorial and survey,

IETE Tech Rev, **39(1)** (2022) 49–62, https://doi.org/10.1080/02564602.2020.1823252.

19 Lawrence S, Giles C L, Tsoi A C & Back A D, Face recognition: A convolutional neural-network approach, *IEEE Transl Neural Netw*, **8(1)** (1997) 98–113, https://doi.org/10.1109/72.554195.

20 Indolia S, Goswami A K, Mishra S P & Asopa P, Conceptual understanding of convolutional neural network-a deep learning approach, *Procedia Comput Sci*, **132** (2018) 679–688, https://doi.org/10.1016/j.procs.2018.05.069.

21 Teoh K H, Ismail R C, Naziri S Z M, Hussin R, Isa M N M & Basir M S S M, Face recognition and identification using deep learning approach, *J Phys Conf Ser*, **1755(1)** (2021) 012006, https://doi.org/10.1088/1742-6596/1755/1/012006.

22 Vivian N I & Ise O A, Face recognition service model for student identity verification using deep neural network and support vector machine (SVM), *Int J Sci Res Comput Sci Eng Inf Technol*, **6(4)** (2020) 11–20.

23 Albawi S, Mohammed T A & Al-Zawi S, Understanding of a convolutional neural network, *IEEE Explore,* (2017) 1–6, https://doi.org/10.1109/ICEngTechnol.2017.8308186.

24 Singh P, Kansal M, Srivastava M & Gupta M, Smart agriculture resource allocation and cost optimization using ML in cloud computing environment, *Convergence Cloud Comput, AI, Agric Sci*, (2023) 152–163, https://doi.org/10.4018/979-8-3693-0200-2.ch008.

25 Kasar M M, Bhattacharyya D & Kim T, Face recognition using neural network: A review, *Int J Secur Appl*, **10(3)** (2016) 81–100, http://dx.doi.org/10.14257/ijsia.2016.10.3.08.

26 Sukmandhani A A & Sutedja I, Face recognition method for online exams, *IEEE Explore*, (2019) 175–179, https://doi.org/10.1109/ICIMTech.2019.8843831.

27 Maheswari V, Sari C A, Setiadi D R I M & Rachmawanto E H, Study analysis of human face recognition using principal component analysis, *IEEE Explore,* (2020) 55–60, https://doi.org/10.1109/iSemantic50169.

28 Zhou K, Oh S K, Pedrycz W & Qiu J, Data preprocessing strategy in constructing convolutional neural network classifier based on constrained particle swarm optimization with fuzzy penalty function, *Eng Appl Artif Intell*, **117** (2023) 105580, https://doi.org/10.1016/j.engappai.2022.105580.

29 Singh P, Kansal M, Singh M K, Kumar S & Dwivedi A, Crop disease prediction using deep learning algorithms, *Convergence Cloud Comput, AI, Agric Sci*, (2023) 290–305, https://doi.org/10.4018/979-8-3693-0200-2.ch015.

30 Yadav S P, Emotion recognition model based on facial expressions, *Multimed Tools Appl*, **80(17)** (2021), 26357–26379, https://doi.org/10.1007/s11042-021-10962-5.