Journal of Engineering and Sustainable Development
Vol. 29, No. 01, January 2025
https://jeasd.uomustansiriyah.edu.iq/index.php/jeasd
ISSN 2520-0917
Research Article
https://doi.org/10.31272/jeasd.2633

Enhancing Surveillance with Machine and Deep Learning-Based Facial Recognition Model: A Proposed Approach for Identification

Husham Salam Saeed[1]*, Muhammad Hassan Fares[2]

[1]Department of Water Resources Engineering, College of Engineering, Mustansiriyah University, Baghdad, Iraq
[2]Department of Computer and Communication/Communication, College of Engineering, Islamic University, Wardenrie Lebanon

*Email: Hushamsalam@uomustansiriyah.edu.iq

Article Info

Received 29/04/2024
Revised 28/11/2024
Accepted 30/11/2024

Abstract

There is limited understanding and utilization of facial recognition models in surveillance. This work addresses the underutilization of facial recognition Models in surveillance contexts. A Model that leverages its proposed facial recognition technology to monitor and locate individuals in real-time video streams and dataset images. The model begins with an initial dataset containing images of specific individuals, such as university professors, missing persons, or criminals. These images extract essential facial attributes for training models capable of identifying individuals in live video recordings. Upon a successful match, the model identifies individuals and tracks their movements using surveillance cameras. A primary objective of this work is to integrate the proposed model seamlessly with the current surveillance infrastructure, minimizing operational costs and disruptions. The work employs two main artificial intelligence approaches: Support Vector Machine achieved an accuracy of 85.33%, demonstrating effective facial recognition compared to the Multilayer Perceptron with 89.0% accuracy. Additionally, Linear Discriminant Analysis achieved the highest classification accuracy at 87.66%. Furthermore, our custom deep learning model demonstrated exceptional accuracy, ranging between 99.5% and 99.8%, showcasing significant advancements over existing methodologies.

Keywords: Custom Deep Learning; Facial Recognition; Linear Discriminant; Multilayer Perceptron; Support Vector Machine

1. Introduction

In recent years, machine learning (ML) and deep learning (DL) advancements have revolutionized various fields, including surveillance models. One of the most significant applications of these technologies is in facial recognition models, which have garnered substantial attention due to their potential to enhance security measures and streamline identification processes[1]. Facial recognition technology holds immense promise in surveillance by offering real-time identification capabilities that can bolster security measures. Traditional surveillance methods often rely on human operators to monitor video feeds, which can be prone to errors and fatigue. In contrast, automated facial recognition models powered by ML and DL algorithms offer consistent and reliable identification of individuals, even in challenging conditions such as low light or crowded environments. Recent advancements in ML and DL techniques have propelled facial recognition models to new heights of accuracy and speed. Algorithms such as convolutional neural networks (CNNs) have demonstrated remarkable capabilities in extracting and analyzing facial features, enabling robust identification and verification tasks. Moreover [2], integrating DL frameworks with large-scale datasets has facilitated training more sophisticated models capable of handling diverse facial appearances and expressions. Despite its promise, facial recognition technology also faces significant challenges and criticisms. Concerns regarding privacy invasion, bias in algorithmic decision-making, and potential misuse of facial data underscore the importance of developing ethically sound and transparent frameworks for deploying these models in surveillance contexts. Addressing these challenges is crucial to fostering public trust and ensuring responsible use of facial recognition technology. This paper proposes a comprehensive approach to enhance surveillance capabilities by integrating ML and DL techniques in facial recognition models [3]. By leveraging state-of-the-art algorithms and methodologies, our objective is to improve accuracy in identifying individuals from live video feeds and archived footage. Key components of our proposed approach include feature extraction, facial embedding techniques, and real-time processing optimizations, which collectively contribute to a more efficient and reliable surveillance infrastructure.

2. Related Works

The authors [4] have introduced a new face recognition method by suggesting four different methods that could work together: independent component analysis, principal component analysis, support vector machine (SVM), and linear discriminate analysis. The dataset has been used as a benchmark. Four hundred images of faces with 40 features, each image taken in different conditions and under different shades. All images' measurements were (112*92) and grayscale. The result was very satisfactory as it achieved 96% by implementing two methods: hybrids based on the principal component analysis and discreet wavelet transform or linear discernment analysis, which are used for data dimensionality reduction, while for classification of the faces, they used SVM.

The authors [5] used three ML methods, Multi Linear Perceptron (MLP), SVM, and Convolutional Neural Network (CNN), for the attendance model. Then, they used the Linear Discriminant (LDA) and Principal component analysis (PCA) algorithms as feature extraction for MLP and SVM. At the same time, the image feeds directly to the CNN as a feature vector. The results were 87%,86.5%, and 98% for the SVM, MLP, and CNN, respectively. Authors in [6] have presented three widely used face recognition methods: hair-like cascade oriented gradient histogram with linear binary pattern cascade, oriented gradient histogram, and SVM. These methods have been recommended for use in OpenCV and Dlib library in Python. The experimental results showed that the Histogram of Oriented Gradients HOG+SVM is more efficient and robot than the Haar and LBP approach as it gets 92% of the total score.

In [7], three machine learning algorithms, SVM, MLP, and Naive Bayes classifiers, were compared to classify the faces in the image based on the face geometry using distance measurements. The results show that the Naive Bays classifier achieved better than other classifiers, with 93.16% accuracy. Authors in [8] have designed a new face recognition approach using machine learning algorithms and principal component analysis. Then, they used the multilayer perceptron, linear discriminate analysis, SVM, and Naïve Bayes. The results show that the PCA achieved 88%, while the linear discrimination achieved 92%.

In [9], the authors discuss the importance of gender recognition from facial images in robotics and interactive models, highlighting its applications and the need for improved accuracy. A novel method is proposed, utilizing minimal facial characteristics and various feature extraction techniques. Evaluation of two datasets demonstrates enhanced accuracy compared to existing processes, with promising results achieved: a gender recognition precision of 98.75% on the FERET database, a runtime performance of 0.4 seconds, and a gender recognition accuracy of 97.43% with a runtime performance of 0.5 seconds on the UTK-Face database.3. Materials and Methods.

3. Methods and Materials

The process begins with an input image that undergoes preprocessing to enhance its quality. This preprocessing involves applying median blur, bilateral filter, histogram equalization, mean filter, and Gaussian filter. Once the image is preprocessed, key features are extracted using techniques like Local Binary Patterns (LBP) and HOG. The extracted features are then split into training and testing datasets. The training data trains various ML models, such as SVM, MLP, and LDA. Alternatively, DL methods like CNN can also be employed for training. After the models are trained, their performance is evaluated using metrics like Mean Squared Error (MSE), Absolute Squared Error (ASE), and accuracy. Fig.1 illustrates the flowchart of the overall work.

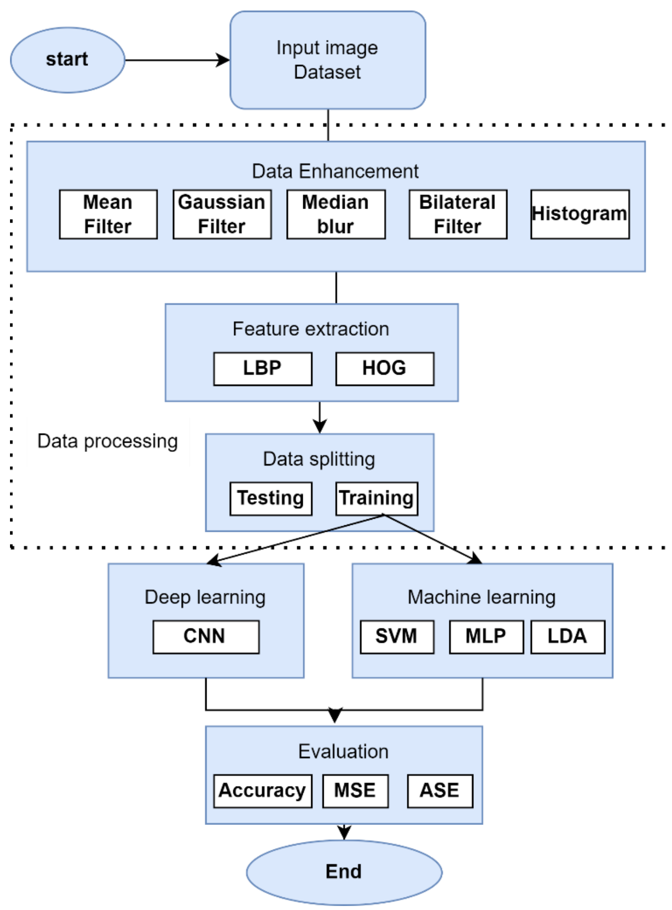

Figure 1. The Flowchart of The Overall Work

3.1. Data Enhancement

For this investigation, image preprocessing involved applying five different filters before training the module. Before applying these filters, histogram equalization was used to enhance the contrast of grayscale images. Histogram equalization is a technique that aims to uniformly distribute pixels' intensity values across the entire range of possible intensities. This improves the visual quality and contrast of the image [10].

The upcoming sections of this paper will provide detailed explanations of the previously mentioned filters. Image processing employs histogram equalization to enhance contrast and visual appeal by redistributing pixel intensity values evenly throughout the image. The primary focus of this work is on noise reduction. Images captured by cameras or generated digitally often exhibit unwanted noise, characterized by random fluctuations in pixel values.

3.1.1 Mean filter

The mean filter, sometimes referred to as the average filter, is a method of digital image processing that smooths and removes noise from images to enhance their quality. The operational approach involves substituting individual pixels inside the image with the mean value determined by squaring the pixels closest to each other. The mean filter efficiently mitigates random fluctuations in pixel intensities by finding the average value, resulting in a more uniform image [11] as in equation (1).

$$\boldsymbol{F'(x,y) = 1/(m*n) * \sum[i = -a \ to \ a] \sum[j = -b \ to \ b] F(x+i, y+j)}$$
(1)

where: F'(x, y) represents the filtered output pixel at coordinates (x, y).

F (x+i, y+j) represents the pixel intensity of the input image at coordinates (x+i, y+j), where i and j iterate over the range of the filter kernel.

m and n represent the dimensions of the filter kernel.

a = floor(m/2) and b = floor(n/2) determine the extent of the filter kernel.

3.1.2 Gaussian Filter

Gaussian blur is a frequently employed image filter in computer graphics and image processing. "Gaussian distribution" derives from the mathematical function widely recognized as the normal distribution, characterized by a bell-shaped curve. The Gaussian kernel is a matrix representation of Gaussian blur. Calculating the blurred value for each pixel involves multiplying the pixel value by the corresponding kernel value in the image [12].

3.1.3 Median blur

A digital image filtering technique known as median blur is employed to mitigate noise and achieve image smoothing [13] effectively. The non-linear filter is characterized by its ability to replace the value of each pixel inside a predetermined kernel or window with the median value of its neighboring pixels. The functioning of the median blur can be described as follows: The determination of the region surrounding each pixel, known as the kernel or window size, which is considered in the computation of the median value, is contingent upon the size of the kernel. Generally, a square-shaped kernel of odd size (such as 3x3, 5x5, etc.) is utilized.

3.1.4 Bilateral Filter

The bilateral filter is a technique commonly used in image processing to reduce noise and make images appear smoother, all while preserving the sharpness of edges. It works well in tasks like removing noise from images and enhancing textures, where maintaining the details of edges is vital. The primary objective of the filter is to blur the image while minimizing blurring around the edges [14]. It achieves this by considering both the variations in intensity and their spatial proximity. By evaluating neighboring pixels, the filter assigns weights based on their similarity in intensity and spatial distance [15].

The bilateral filter consists of two components: the component and the range component.

In this work, the filter has been implemented using the following methods for each pixel in the image: a window or kernel is defined, which includes neighboring pixels. The size of this window, sometimes adjustable as a parameter, determines how far the filtering will be.

3.1.5 Histogram Equalization

When attempting to enhance contrast and features in an image, the contrast-limited Adaptive Histogram Equalization (CLAHE) method is utilized [16]. This technique's operational mechanism involves modifying brightness levels in various image portions to maximize the visibility of distinguishing characteristics. The operational process can be described in the following manner at its most fundamental level: In the context of image processing, the term "image segmentation" refers to the process of dividing or partitioning an image into distinct portions or components, with the application of particular.

3.2 Feature Extraction using Local Binary Patterns

The LBP texture descriptor is commonly employed in computer vision and image processing. In 1994, Timo Ojala, Matti Pietikäinen, and Topi Mäenpää gave a presentation on the previously described work. Applications for LBP include texture classification, face analysis, object recognition, and other related fields. LBP has proven to be an effective method for identifying local texture patterns. The Gary level of the central pixel is used to calculate a threshold value for neighboring pixels in the LBP method. The following illustrates this process [17] as in equation (2).

$$alLBP(x_c, y_c) = \sum_{i=0}^{7} s(g_i - g_c)2^i \qquad (2)$$

$$(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$$

The arrangement of the pixels in the immediate region of the central pixel is circular. The size of the neighborhood can be adjusted based on the desired level of detail. The available radius options are 3 × 3, 5 x 5, and 8 x 8. The method of three-holding entails LBP's primary objective to analyze an image's texture by contrasting the intensity values of a core pixel with those of its surrounding pixels. At the center of a cell, each pixel has a neighboring, comparing the density values of adjacent pixels with the density value of the core pixel. If the density value of the neighboring pixels is more than or equal to the value of the central pixel, a binary value of 1 is assigned. Otherwise, a binary value of 0 is assigned. The binary pattern produced by the adjacent element at the top left position is created by combining the binary values of each neighboring component in either a clockwise or counterclockwise orientation.

3.3 Machine learning

3.3.1 Multilayer Perceptron

An MLP, often called a network, comprises layers of perceptron units. This network operates feedforward, moving information from the input layer through the layers to reach the output layer. Each perceptron unit, in a layer, takes input signals that undergo a linear transformation before being processed by an activation function. The results from each perceptron unit are then sent to the layer by adding layers between the input and output layers. This network can understand aspects of the input data. The setup of the MLP can be adjusted to handle varying numbers of layers and perceptron units within each layer [18]. The choice of activation function, such as sigmoid or Rectified Linear Unit (ReLU), also impacts how the MLP functions and what it can achieve. Throughout the training process, the weights and biases of the MLP are refined using methods like backpropagation. Backpropagation utilizes a descent method to reduce the gap between the network's expected outcomes and the intended results. The MLPs are used in various fields, such as classification, regression, and pattern identification. They have proven helpful in time series analysis, natural language understanding, and image recognition. (Taud, Hind Mas, J F).

3.3.2 Linear Discriminant

Utilizing pattern recognition and ML involves using analysis, a method that helps simplify data and tackle classification challenges. The objective is to ascertain a linear combination of features that maximizes the separation between different classes while reducing the dispersion within each class. The LDA algorithm accomplishes this objective by preserving class-discriminatory information while projecting high-dimensional data onto a lower-dimensional space. Identifying the discriminant axes involves the computation of class statistics, including means and scatter matrices, and addressing the eigenvalue issue [19]. The data is transformed into a novel subspace delineated by these axes. LDA is commonly employed in several applications, such as face recognition, character recognition, and document categorization.

3.3.3 Support Vector Machine

A supervised ML approach is used to tackle classification and regression challenges. It effectively sorts data points into two groups, especially when dealing with binary classification problems [20]. To tackle tasks related to classifying, SVM can be expanded. The primary objective of SVM is to discover the hyperplane that maximizes the separation between data points from classes. Using a hyperplane as a dividing line, the feature space is split into two areas, each corresponding to a class. The margin is the distance between the hyperplane and the nearest data points from each class. The main objective of the SVM algorithm is to optimize this margin. SVM achieves this by expanding the input data's dimensionality using a kernel function, creating a feature space to determine the hyperplane accurately. Kernel functions such as basis function (RBF) aid SVM in recognizing relationships among input variables [21]. Sigmoid kernels and similar functions are commonly employed as kernel functions in applications. When training SVM, a crucial step involves solving an optimization problem to minimize classification errors and maximize the margin by identifying support vectors near the decision boundary. These support vectors are vital for predicting data points. Contribute to constructing the hyperplane, enabling the classification of data points based on their proximity to the decision boundary – assigning them to classes depending on which side of the hyperplane they lie on. The methods used in ML will be compared; this table summarizes the characteristics of three classification algorithms: SVM, MLP, and LDA. It outlines key properties of each algorithm, such as the type of algorithm, whether it's supervised or unsupervised, whether it's linear or non-linear, the number of classes it can classify into, whether it handles missing data, and if it has a method for generalizing across multiple models (Ensemble Method). This information helps compare algorithms and select the most appropriate one for solving a specific classification problem, as shown in Table 1.

Table 1. Comparison between the SVM, LDA, and MLP methods.

Algorithm	**SVM**	**MLP**	**LDA**
Type	Classification	Classification	Classification
Supervised/ Unsupervised	Supervised	Supervised	Supervised
Linear/Non-linear	Both	Non-linear	Linear
Number of Classes	Binary/Multi	Binary/Multi	Multi
Handles Missing Data	No	No	No
Ensemble Method	No	No	No

It has been used (MSE) because it sensitively penalizes significant errors, is differentiable for optimization, and provides interpretable results in squared error units [22] as in equation (3).

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{Yi} - Y^{\wedge}\mathrm{i})^2 \tag{3}$$

(ASE) offers benefits by being robust to outliers, directly interpretable in original units, and more straightforward to compute and understand than MSE.

(ASE) offers benefits by being robust to outliers, directly interpretable in original units, and more straightforward to compute and understand than MSE, as in equation (4).

$$ASE = \frac{1}{n}\sum_{i=1}^{n}|\mathrm{Yi} - Y^{\wedge}\mathrm{i}| \tag{4}$$

The paper uses the accuracy equation to ensure clarity and reproducibility as in equation (5).

$$\text{Accuracy}= \frac{No.of\ Correct\ Predictions}{Total\ No.of\ Predictions} * 100\% \tag{5}$$

3.4 Deep learning

3.4.1 Custom Deep Learning

The program employs CNN, a type of (DL) technique. CNNs have become widely popular and successful in computer vision assignments. These models are specifically designed for computer vision and image recognition tasks, leading to advancements in object recognition, image classification, and image segmentation fields.

The functional structure of CNN: The convolutional layer plays a role in a CNN. It involves using filters, also known as kernels, that move across the input image to perform multiplication and addition operations at each element, creating feature maps [23]. These filters learn to identify patterns, edges, and textures in the image as they act as feature detectors.

Following the convolution step, an activation function is applied to introduce linear behavior into the model. Typically, a Rectified Linear Unit (ReLU) is employed as the activation function to enable the CNN to grasp patterns.

The pooling layer decreases the size of dimensions and compresses the feature maps. Usually, a Max-Pooling operation follows each layer. This method simplifies and improves the model's ability to handle variations in input data [24].

After a series of convolutional and pooling layers, the feature maps undergo a flattening process, resulting in a one-dimensional vector. By employing this approach, the spatial arrangement is reduced while retaining the acquired characteristics. Fully connected layers are incorporated into the network after the process of flattening when thick layers are utilized to establish complete connectivity. To provide predictions, these layers make use of the acquired features. The code has two dense layers, one consisting of 128 neurons and the other consisting of 2 neurons, which address the binary classification task of detecting or not detecting a certain phenomenon. At the end of a CNN, the SoftMax activation function converts the network output into a probability distribution [25]. This code applies the SoftMax function to convert scores into probabilities for classifications that determine whether a specific class is detected. The model undergoes training using the entropy loss function specifically designed for handling binary classification tasks within a multi-class framework. The Adam optimizer adjusts the model's weights throughout training to reduce the loss function. The CNN fine-tunes its weights throughout the training phase using the gradients calculated with the backpropagation algorithm. The objective is to minimize the loss by pinpointing and capturing the features from the input data. Upon completing training, the model can correctly categorize recently taken photos into their categories. The SoftMax layers results offer the likelihoods for each category. The predicted category is chosen based on the one, with the probability that CNNs are created to recognize patterns in input data without requiring feature engineering. Through the utilization of filters to identify patterns and connected layers for categorization, CNNs have shown success in a range of computer vision assignments, marking progress in the realm of DL, as shown in Fig.2. The description of the hyper-parameters of DL is shown in Table 2.

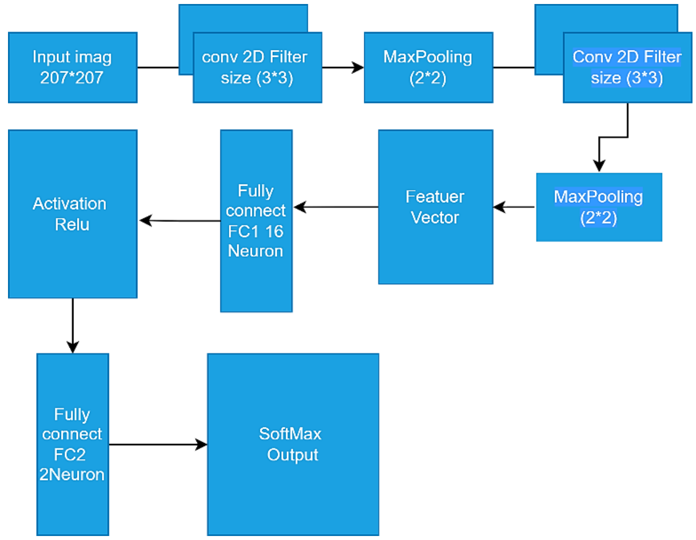

Figure 2. The custom deep learning

Table 2. Hyper-parameters for deep learning

Hyper-parameter	Description
Input Image Size	The dimensions of the input images. In this case, it's 207×207 pixels.
Conv 2D Filter Size	The size of the convolutional filters. Here, the filter size is 3×3 for both convolutional layers.
Max-Pooling Size	The dimensions of the pooling operation. Here, the pooling size is 2×2, which reduces the spatial dimensions by half.
Activation Function	The activation function is used after the convolutional layers. Here, it's ReLU (Rectified Linear Unit).
Fully Connected Layers	The number of neurons in the fully connected layers. In this case, FC1 has 16 neurons, and FC2 has two neurons.
Output Layer	The final layer uses the SoftMax function to output the probability distribution for classification.

4. Result and discussion

The CNN architecture comprised an input layer for 207x207 pixel images, two convolutional layers with 3x3 filters, and max-pooling layers with 2x2 pool sizes to extract and down-sample features. The feature maps were then flattened into a feature vector, followed by two fully connected layers, with the final layer utilizing a SoftMax activation function to produce class probabilities. The results reported in this paper were measured using a PC with the following specifications: Intel Core i7, 16 GB RAM, and NVIDIA GTX 1650. Python was also used to implement the algorithms and conduct the experiments. It will insert samples into five filters to eliminate noise and distortion and enhance images. Next, the data will undergo ML processing before moving to the DL input step.

- **Filter**

A. Mean filter

In Fig. 3, the results of a Mean filter have been taken from the image data set, and the figure above shows the results of this filter.

Figure 3. The image is a Mean filter

B. Gaussian Filter

Fig. 4 shows the results of a Gaussian Filter taken from an image data set, and the figure above shows the results of this filter.

Figure 4. The image is a Gaussian Filter.

C. Median blur

Fig. 5 shows the results of a Median blur taken from the image data set; the figure above shows the results of this filter.

Figure 5. The image is a Median blur.

D. Bilateral Filter

Fig. 6 shows the results of a bilateral filter taken from the image data set, and the figure above shows the results of this filter.

Figure 6. The image is a bilateral filter.

E. CLAHE Histogram Equalization

Fig. 7 shows the results of a CLAHE Histogram Equalization taken from the image data set; the figure above shows the results of this filter.

Figure 7. The image is a CLAHE Histogram Equalization.

• Machine learning algorithms

Table 3 shows the accuracy of the ML algorithms. The SVM accuracy has achieved 85.33%, which means that this method has recognized the face accurately compared with the MLP method, which recorded 89.0 % overall accuracy. On the other hand, the higher classification accuracy was 87.66% by using the LDA method. As a result, the proposed model in this work has successfully detected the faces in the image based on the filters and the extracted features. In terms of MSE, the table below shows that the lower MSE value was in LDA methods, which means that the LDA has good recognition of faces in image processing, and the lower the MSE, the better the model prediction. These are the results of three ML methods: LDA, MLP, and SVM. The same data set was used to compare with DL and prove that DL is better in accuracy than ML, as in Fig. 8.

Table 3. The results of machine learning algorithms.

Methods	Accuracy	MSE	ASE
SVM	85.33%	9.80%	9.05%
MLP	89.0 %	9.52%	8.18%
LDA	87.66%	9.31%	8.09

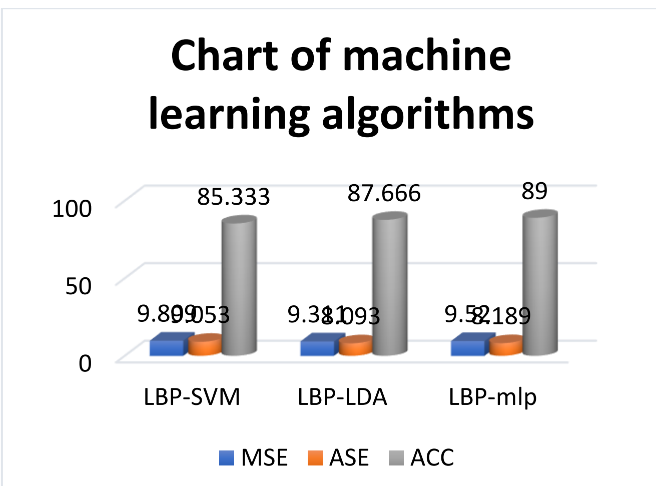

Figure 8. Chart of machine learning algorithms.

• Deep Learning

Regarding DL, the same filters were used to extract the features from the image using the same dataset. In addition, the training and the validation tests were considered, as it is very important to evaluate the model. In Fig.9, the training and validation test was conducted with one subject, subject 20; it is shown that the results were very satisfactory (0.0374 and 0.0135 in training and validation, respectively). The test loss recorded was 0.0005889, which is very low for all epochs of the proposed model. On the other hand, Fig.10 illustrates the model accuracy between 99.5% and 99.8%.

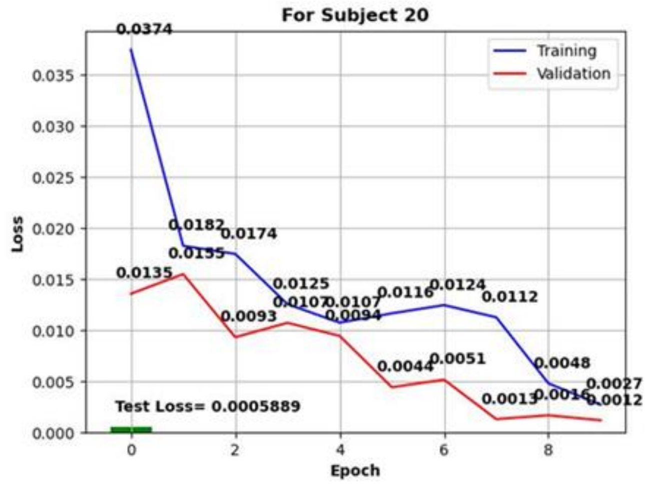

Figure 9. Models training and validation results.

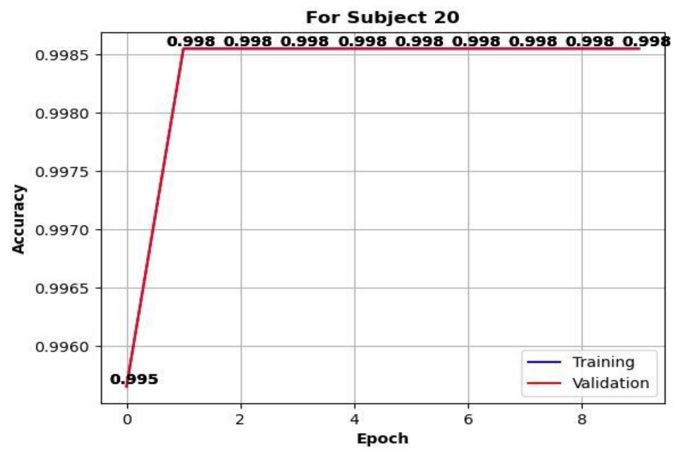

Figure 10. illustrates the model's accuracy.

In Fig.11, the results of the training and validation test have been taken for one subject, subject 430. The training was 0.0323, the validation was 0.0268, and the test loss was 0.0005889. Fig.12 demonstrates the model accuracy, which was between 99.4 % and 99.8%.

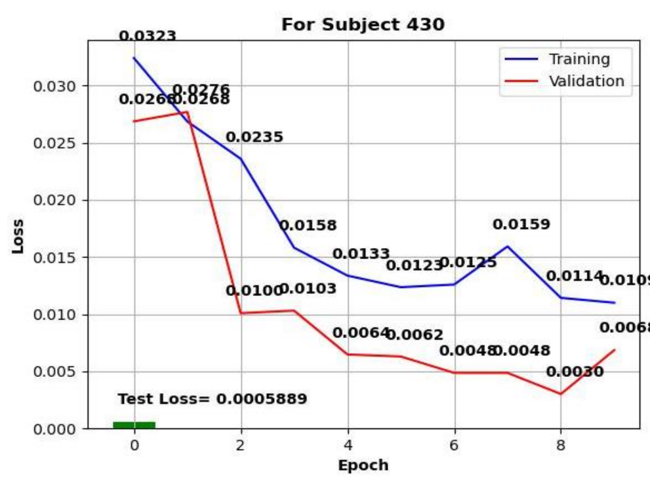

Figure 11. Model training and validation results.

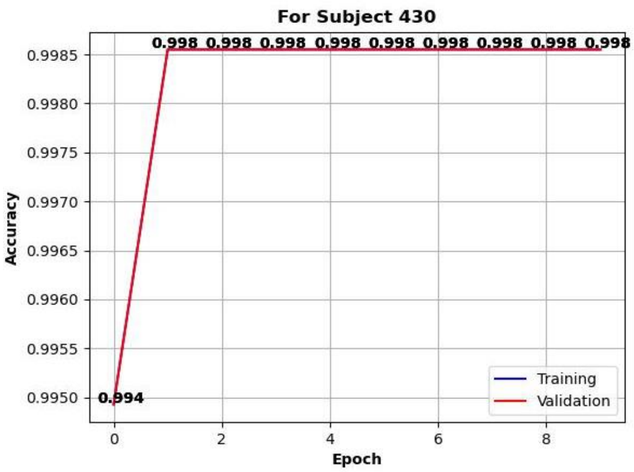

Figure 12. Illustrates the model's accuracy.

Fig.13 illustrates the results of the training and validation for different subjects: subjects 0,100, 200, 300, 400, 500, 600, 700, and 800. It shows that the model is strong in detecting the faces in the image.

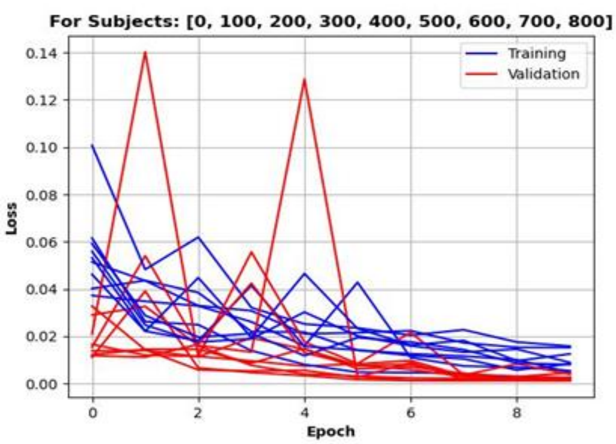

Figure13. Model training and validation results for different subjects.

Table 4. illustrates the comparisons between the proposed method and other related work methods.

The proposed model outperforms others by achieving 99.5%-99.8% accuracy in face recognition using DL, which is higher than the best results from similar work. Even in ML, my model shows competitive accuracy, making it one of the most effective approaches for face recognition under varying conditions.

Table 4. The Proposed Method and Related Work Comparison

Work	Algorithms	Results		
		SVM	MLP	LDA
Our proposed method	Machine Learning	85.3%	89%	87.6%
	Deep Learning	99.5%-99.8%		
[4]	ICA, PCA, SVM, LDA	96%		
[5]		SVM	MLP	
	Machine Learning	87%	86.5%	
	Deep Learning	CNN: 98%		
[6]	HOG+SVM	92%		
[7]	SVM, MLP, Naive Bayes	Naive Bayes: 93.16% (best performance)		
[8]	MLP, LDA, SVM, Naive Bayes	PCA	LDA	
		88%	92%	
[9]	Facial features and feature extraction techniques	FERET	UTK-Face	
		98.75%	97.43%	

5. Conclusions

In this work, a face recognition model has been applied to identify the persons in the image with different conditions based on ML and DL. The proposed model provided around 85.333%, 89%, and 87.666% accuracy for SVM, MLP, and LDA, respectively, in DL, while the accuracy was between 99.5% and 99.8% in DL. Different computer vision techniques, such as filters and feature extraction methods, achieved these results. The LFPW data set was applied, with 858 images in total. In addition, this model has been validated accurately to check if it works properly. Consequently, the proposed tracking model has achieved very satisfactory results for detecting and recognizing faces, and it could also be used for personal safety and security verification without any challenges.

Acknowledgments

The authors are grateful to Mustansiriyah University and the Islamic University of Lebanon (https://www.uomustansiriyah.edu.iq/) and (https://www.iul.edu.lb/).

Conflict of interest

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Author Contribution Statement

Husham Salam Alturaihi: Developed methodology for the model, status problem statement, collected data, and result analyses.

Muhammad Hassan Fares: Problem statement and result analyses.

Both authors supervised the findings of this work, discussed the results, and contributed to the final manuscript.

Reference

[1] Y. Yao, "Research on facial recognition system based on deep learning," *Applied and Computational Engineering*, vol. 34, no. 7, pp. 232–236, 2024, doi: https://doi.org/10.54254/2755-2721/34/20230332.

[2] L. Alzubaidi *et al.*, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *J Big Data*,

vol. 8, no. 53, pp. 1–74, 2021, doi: https://doi.org/10.1186/s40537-021-00444-8.

[3] L. Qinjun, C. Tianwei, Z. Yan, and W. Yuying, "Facial Recognition Technology: A Comprehensive Overview," *Academic Journal of Computing & Information Science*, vol. 6, no. 7, pp. 15–26, 2023, doi: https://doi.org/10.25236/AJCIS.2023.060703.

[4] Z. B. Lahaw, D. Essaidani, and H. Seddik, "Robust Face Recognition Approaches Using PCA, ICA, LDA Based on DWT, and SVM Algorithms," in *2018 41st International Conference on Telecommunications and Signal Processing, TSP 2018*, 2018, pp. 413–417. doi: https://doi.org/10.1109/TSP.2018.8441452.

[5] R. C. Damale and B. V. Pathak, "Face Recognition Based Attendance System Using Machine Learning Algorithms," in *Proceedings of the 2nd International Conference on Intelligent Computing and Control Systems, ICICCS 2018*, 2019, pp. 414–419. doi: https://doi.org/10.1109/ICCONS.2018.8662938.

[6] A. Adouani, W. M. Ben Henia, and Z. Lachiri, "Comparison of Haar-like, HOG and LBP approaches for face detection in video sequences," in *2019 16th International Multi-Conference on Systems, Signals & Devices (SSD)*, 2019, pp. 266–271. doi: https://doi.org/10.1109/SSD.2019.8893214.

[7] N. Sabri *et al.*, "A comparison of face detection classifier using facial geometry distance measure," in *2018 9th IEEE Control and System Graduate Research Colloquium, ICSGRC 2018 - Proceeding*, 2019, pp. 116–120. doi: https://doi.org/10.1109/ICSGRC.2018.8657592.

[8] S. Sharma, M. Bhatt, and P. Sharma, "Face recognition system using machine learning algorithm," *Proceedings of the 5th International Conference on Communication and Electronics Systems, ICCES 2020*, no. Icces, pp. 1162–1168, 2020, doi: https://doi.org/10.1109/ICCES48766.2020.09137850.

[9] M. J. Al_Dujaili, H. T. H. Salim ALRikabi, and I. R. Niama ALRubeei, "Gender Recognition of Human from Face Images Using Multi-Class Support Vector Machine (SVM) Classifiers.," *International Journal of Interactive Mobile Technologies*, vol. 17, no. 8, 2023, https://doi.org/10.3991/ijim.v17i08.39163.

[10] J. I.-Z. Chen, C.-Y. Lee, and W.-C. Lo, "The Pre-Processing of WT Blade Images by SS and Bilateral Filter with Machine Learning Frameworks," *J Mar Sci Technol*, vol. 31, no. 4, p. 16, 2023, doi: https://doi.org/10.51400/2709-6998.2724.

[11] S. H. Fatima, A. Munir, and S. T. Hussain, "Image denoising using difference classifier and trimmed global mean filter adaptive approach," *Vis Comput*, vol. 40, no. 8, pp. 1–13, 2023, doi: https://doi.org/10.1007/s00371-023-03106-3.

[12] H. H. Afshari, S. A. Gadsden, and S. Habibi, "Gaussian filters for parameter and state estimation: A general review of theory and recent trends," *Signal Processing*, vol. 135, no. 1, pp. 218–238, 2017, doi: https://doi.org/10.1016/j.sigpro.2017.01.001.

[13] G. George, R. M. Oommen, S. Shelly, S. S. Philipose, and A. M. Varghese, "A survey on various median filtering techniques for removal of impulse noise from a digital image," in *2018 Conference on Emerging Devices and Smart Systems (ICEDSS)*, IEEE, 2018, pp. 235–238. doi: https://doi.org/10.1109/ICEDSS.2018.8544273.

[14] R. Li, "An optimal algorithm for bilateral filtering," in *Journal of Physics: Conference Series*, IOP Publishing, 2023, p. 12030. doi: https://doi.org/10.1088/1742-6596/2634/1/012030.

[15] B.-H. Chen, Y.-S. Tseng, and J.-L. Yin, "Gaussian-adaptive bilateral filter," *IEEE Signal Process Lett*, vol. 27, no. 1, pp. 1670–1674, 2020, doi: https://doi.org/10.1109/LSP.2020.3024990.

[16] P. Musa, F. Al Rafi, and M. Lamsani, "A Review: Contrast-Limited Adaptive Histogram Equalization (CLAHE) methods to help the application of face Recognition," in *2018 Third International Conference on Informatics and Computing (ICIC)*, IEEE, 2018, pp. 1–6. doi: https://doi.org/10.1109/IAC.2018.8780492.

[17] K. Meena, "LOCAL BINARY PATTERNS AND ITS," in *IEEE-International Conference on Recent Trends in Information Technology*, 2020, pp. 3–5. doi: https://doi.org/10.1109/ICRTIT.2011.5972286.

[18] H. Taud and J. F. Mas, "Multilayer perceptron (MLP)," in *Geomatic approaches for modeling land change scenarios*, Springer, 2018, pp. 451–455.Book. https://www.springerprofessional.de/en/geomatic-approaches-for-modeling-land-change-scenarios/15171360?tocPage=1

[19] F. Zhu, J. Gao, J. Yang, and N. Ye, "Neighborhood linear discriminant analysis," *Pattern Recognit*, vol. 123, no. 1, p. 108422, 2022, doi: https://doi.org/10.1016/j.patcog.2021.108422.

[20] X. Yan and H. Zhu, "A novel robust support vector machine classifier with feature mapping," *Knowl Based Syst*, vol. 257, no. 3, p. 109928, 2022, doi: https://doi.org/10.1016/j.knosys.2022.109928.

[21] X. Ding, J. Liu, F. Yang, and J. Cao, "Random radial basis function kernel-based support vector machine," *J Franklin Inst*, vol. 358, no. 18, pp. 10121–10140, 2021, doi: https://doi.org/10.1016/j.jfranklin.2021.10.005.

[22] S. Al Ghifari and H. Studiawan, "Image Splicing Localization Using Superpixel and Wavelet Mean Squared Error," in *2023 International Conference on Information Technology (ICIT)*, IEEE, 2023, pp. 593–598. doi: https://doi.org/10.1109/ICIT58056.2023.10226015.

[23] M. Krichen, "Convolutional neural networks: A survey," *Computers*, vol. 12, no. 8, p. 151, 2023, doi: https://doi.org/10.3390/computers12080151.

[24] A. S. Mohammad, "Multi-Modal Ocular Recognition in Presence of Occlusion in Mobile Devices," 2018. https://hdl.handle.net/10355/69672

[25] R. Chauhan, K. K. Ghanshala, and R. C. Joshi, "Convolutional neural network (CNN) for image detection and recognition," in *2018 first international conference on secure cyber computing and communication (ICSCCC)*, IEEE, 2018, pp. 278–282. doi: https://doi.org/10.1109/ICSCCC.2018.8703316.