

An Improved Convolutional Neural Network (Icnn) Model And U-Net Segmentation Of Facial Features For 2d-3d Facial Image Analysis

Aswathy.R¹, Dr. A. Sherin²

¹Research Scholar, Nehru Arts and Science College Email id: aswathyravi998@gmail.com

²Assistant Professor & Head, Department of Digital and Cyber Forensic Science, Nehru Arts and Science College Email id: sherinfaizalrahiman@gmail.com

Abstract: Face recognition is a pivotal biometric tool extensively utilized in commercial and law enforcement applications for identifying individuals. This study introduces a novel framework for face recognition that leverages two-dimensional (2D) facial images sourced from diverse origins to construct three-dimensional (3D) face meshes using 468 MediaPipe landmarks. The preprocessing pipeline incorporates adaptive median filtering to enhance image quality, followed by deep learning-based segmentation via U-Net architectures to effectively separate the foreground (face) from the background, thereby reducing noise and improving detection precision. Feature vectors are generated by calculating Euclidean and Geodesic distances and their ratios between identified landmarks, capturing intricate facial geometries. These feature vectors are subsequently trained using an Improved Convolutional Neural Network (ICNN) classifier, designed to maintain high accuracy in environments characterized by significant pose variations, facial expressions, and occlusions. According to the experimental results, a range of image attributes may be learned by the improved convolutional neural network approach proposed in this paper, increasing the network's ability to identify images and extract information efficiently.

Keywords: Face recognition, two-dimensional (2D) facial images, three-dimensional (3D), preprocessing, Segmentation, Euclidean and Geodesic distances and Improved Convolutional Neural Network (ICNN) classifier.

1. Introduction

A significant quantity of visual data has come along with the rapid expansion of the mobile Internet, the pervasive usage of smartphones, and the popularity of social media and self-media [1], [2]. However, when images become a significant network information carrier, issues also occur. By searching for keywords, we may obtain and analyze the necessary content from traditional information items, which are recorded by words [3]. The information that is sent by visuals, however, is not something we can extract or process. We can easily record and transmit information with the picture, but we find it challenging to utilize the information it conveys [4]. In this instance, it is very important to figure out how to utilize a computer to properly categorize and identify the information contained in these images.

Following the extraction of a particular quantity of synthetic characteristic points, developing the image is described by a mathematical statistical model that is the crucial phase in traditional pattern recognition approaches [5], [6]. The images may then be identified using the image matching approach. The fundamental idea behind this approach is that similar samples cluster together and create a "clustering" in the pattern space. The classifier is then used to classify and recognize the clustering. Such shallow machine learning techniques have low recognition, for instance, pedestrian identification employs histogram of oriented gradient (HOG) features, scale-invariant feature transform (SIFT) features for object identification, and local binary patterns

(LBP) features for face recognition. As artificial intelligence has advanced, deep learning advances resulted in significant success in several fields, speech recognition, computer vision, multimedia, and natural language processing and video analysis [7]. An excellent advancement for artificial intelligence is the growing application of deep learning in image categorization by academics and companies.

In recent years, artificial intelligence has rapidly improved. Examples of contemporary innovations include self-service supermarkets and self-driving automobiles. The domains of artificial intelligence and computer vision are closely intertwined [8]. In contrast to computer vision, which replicates human vision but does so electronically to receive and interpret images, individuals utilize their vision to understand and adjust to their surroundings. In addition to seeing, computer vision must also be able to respond. As with human vision, it must be equipped with the capacity to recognize, detect, and analyze images. To comprehend and adapt to their environment, individuals use their vision [9]. The objective of computer vision is to carry out the same tasks as effectively as possible. Identifying, assessing, and selecting are the three primary processes that the driver's brain goes through. But one of the basic components of intelligence is vision. coordinating, memory, recall, reasoning, estimate, recognizing, and more are visual components. A system is not considered a vision if it merely has one of these capabilities [10]. In reality, computer vision resembles human systems. It is more difficult for computers to assess objects in three dimensions since our reality is three-dimensional, although the representations that our visual sensors usually show are merely two-dimensional.

Considering that, there are several convolutional neural network-based image recognition systems with excellent recognition results. However, to construct the network's depth and level, many recognition methods are now dependent on a particular database [11]. The optimal settings and optimization techniques are discovered by ongoing investigation. There is no systematic theory to influence the convolutional nerve's recognition impact, and the human aspect is comparatively significant. The selection of the convolutional neural network's starting state parameters and optimization technique will significantly affect network training, particularly when it comes to categorizing and identifying natural images. A disastrous selection may prevent the network from functioning or may result in local minimization, under-fitting, over-fitting, and several other issues. To increase the classification capabilities of convolutional neural networks, this research suggests an improved approach and recognition capabilities for two-dimensional images, accelerate the algorithm's convergence, provide accurate classification results while reducing training time and iterations.

The remaining studies are presented as follows: Section 2 explores the latest face-identification methods. Section 3 shows the suggested method. Section 4 reports findings and comments. The conclusion and next steps are in Section 5.

2. Literature Review

In this section, some of the most modern methods for facial recognition employing sophisticated deep learning algorithms are reviewed. The comparison of the alternative approaches is shown in Table 1.

Table 1. Comparison of the existing methods

Author and year	Methods	Advantages	Disadvantages
He et al [2023]	Three-channel convolutional neural network (HFT-CNN)-based multi-layer feature recognition method	Allow robust reconstruction and interpretation of the data	Hyper parameter tuning requires more time
Tian et al [2020]	Convolutional Neural Network (CNN)	Best Accuracy results	Overfitting problem occurs
Aggarwal et al [2021]	Artificial Neural Network (ANN) and Singular Value Decomposition in the Hidden Markov Model (HMM)	greatest rate of 97.49% recognition	Feature representation is need to be analyzed
Soni et al [2021]	Optimized Deep Neural Network (DNN)	lightweight face recognition models	Noise and illumination variation problem occur
Liu et al [2021]	MobileFaceNet	obtained high-precision	Less landmark points handled
Sun et al [2018]	Region based Convolutional Neural Network (RCNN) framework	best model in terms of ROC curves	High error rate for high dimensional dataset
Rastgoo et al [2019]	Alexnet model	High recognition rate	Difficult to handle large dataset
Teoh et al [2021]	Deep learning	Suitable technique for facial identification because of its great precision	High processing time
Hangaragi et al [2023]	Face Mesh and deep neural network	To minimize the average power consumption	Time complexity
Singh et al [2020]	artificial neural network (ANN) and convolutional neural network (CNN)	CNN outperforms ANNs in face recognition accuracy and epoch count	Occlusion problem occur in nature

Sanil et al [2023]	Hyper-parameter optimized machine learning algorithms	This method provided excellent results	Limitation of face orientation
--------------------	---	--	--------------------------------

From the literature study, the above techniques probably contributed to the robustness and classification the face recognition system's accuracy. This survey delves into the extensive literature on FR, examining the essential stages of the FR process and highlighting numerous challenges that profoundly influence system performance. The survey offers a comprehensive overview of FR systems, tracing their development throughout history. Moreover, a taxonomy of facial recognition methods and summarizes popular facial datasets used for training and testing these systems. The survey meticulously examines image-based and video-based FR methods, providing detailed comparisons of various approaches.

Cross-sensor, cross-resolution, and cross-age 3D face recognition is still very difficult to do. Typically, various sensors are used to gather gallery 3D faces and probe 3D faces at different periods, which leads to differing noise levels and resolutions. The challenge of creating techniques for reliable, effective, and accurate 3D face identification under these circumstances remains unresolved. This paper presents a new face identification system that uses 468 MediaPipe markers to build two-dimensional (2D) facial images from several sources are converted into three-dimensional (3D) face models.

3. Proposed Methodology

This study introduces a novel framework for face recognition that leverages two-dimensional (2D) facial images sourced from diverse origins to construct three-dimensional (3D) face meshes using Media Pipe landmarks. The preprocessing pipeline incorporates adaptive median filtering to enhance image quality, followed by deep learning-based segmentation via U-Net architectures to effectively separate the foreground (face) from the background, thereby reducing noise and improving detection precision. Feature vectors are generated by calculating Euclidean and between recognized landmarks, geodesic distances and their ratios, capturing intricate facial geometries. These feature vectors are subsequently trained using an Improved Convolutional Neural Network (ICNN) classifier, designed to maintain high accuracy in environments characterized by significant pose variances, obstructions, and face movements.

To produce the most effective machine learning methods for face recognition selection, a framework for detecting the same image, separate images, and look-alikes is offered. The database included 2D facial images. To recognize and normalize faces, preprocessing was applied to each selected image. An actual 3D face model was created by finding landmarks on 2D facial images using the suggested approach. A 3D face image was then layered with these landmarks. After that, the "Euclidean Distance" and "Geodesic Distance" between two anthropometric point pairs are determined to extract features. A fusion of features is created using the curvature properties, and curves are the basis for their ratios. These feature vectors are subsequently trained using an Improved Convolutional Neural Network (ICNN) classifier, designed to maintain high accuracy.

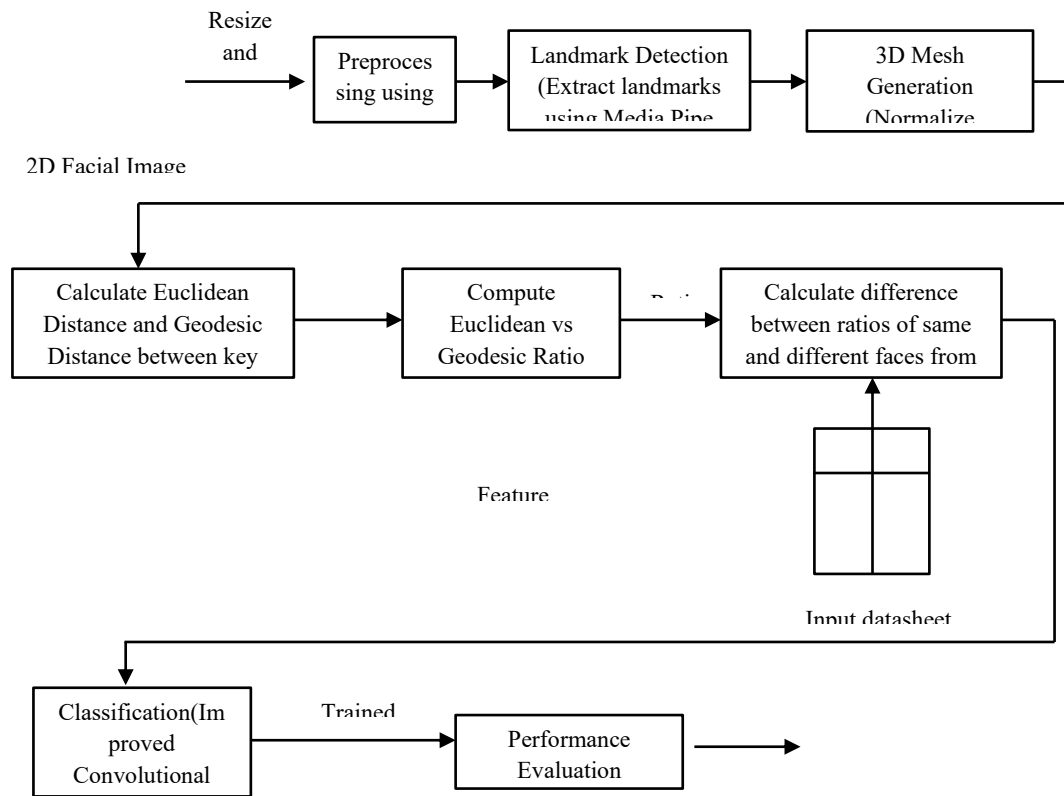


Figure 1. The suggested methodology's overall procedure

3.1. Data Collection

The purpose according to the database Labelled Faces in the Wild (LFW) is to research the unconstrained face identification issue using face images. The Acknowledgments section contains explicit references to University of Massachusetts, Amherst scholars established and maintain this database. The 5,749 people's 13,233 images that were collected from the internet were recognized and centered using the Viola Jones face detector. Two or more distinct images of 1,680 of the people in the collection are included. Four LFW sets and three "aligned" sets may have been in the primary database. Most face verification methods worked better with deep-funnelled images.

3.2. Pre-processing

The purpose of acquiring 2D face data was to offer "clean" faces for normalization and detection. The suggested method creates a facial image dataset in MATLAB software by first selecting a face image and then doing pre-processing steps like cropping and scaling. AMFs, or adaptive median filters, are used in this study to eliminate extraneous noise from the provided FERC images. Their main function is to improve image results in situations with high noise levels by acting as a reliable and effective model. AMFs employ spatial algorithms for identifying image pixels that are noisy [23]. They compare pixels with their neighbours for these determinations where both comparison criteria and neighbourhood sizes can be varied [24]. Noisy pixels can be defined as pixels that differ from majority of their neighbours and are not physically aligned with their comparable pixels. Median pixel values of nearby pixels that pass noise labelling tests are then used to replace noisy pixels. Therefore, they are applied to lower noises in FERC images and enhance image qualities.

Adaptive filter window sizes grow in proportion to location's candidate noise levels. Region window sizes are computed using equations based on counts of noisy pixels in the regions (1). Thresholds are established for altering window sizes after rigorous simulations with guarantees of the best results according to the quantity of pixel noise in areas and MSEs (Mean Square Errors).

$$W(M) = \sum_p C(p, D_p) + \sum_{q \in N_p} T[|M_p - M_q| = 1] \quad (1)$$

$$+ \sum_{q \in N_p} T[|M_p - M_q| > 1]$$

The window matching function is W , and M is its optimal solution. The face image's pixel P and its nearest pixel q are shown. N_p lists pixels adjacent to p predicted p - q matching windows M_p and M_q . C represents a matching window's cost. For true assertions, the logic function $T[\cdot]$ returns 1, and for false assertions, statements after evaluation. When window's median pixels are noisy or no noise-free pixels exist in windows, their sizes are increased based on equation (1). Window sizes are employed to filter adaptive image pixels implying they are increased when predetermined parameters are not satisfied and median values are used to filter pixels when the criteria are met. Consider the following: W denotes the current window size; I_{ij} represents the impaired image pixel; I_{min} signifies the minimum pixel value; I_{max} indicates the window's maximum pixel value; and W_{max} refers to the maximum attained window size. I_{med} may represent the median of the specified window.

This method employs a two-layer filtering strategy, detailed below.

Level A:

- a) If $I_{min} < I_{med} < I_{max}$, thus, median numbers aren't noise hence Level B is followed by the model to examine if current pixels are noisy.
- b) Alternatively, using equation (1) for increasing window sizes, Level A gets repeated until median values are no longer noisy and until the maximum window widths and median values are achieved, the model transitions to Level B are assigned as pixel values in filtered images.

Level B:

- a) If $I_{min} < I_{ij} < I_{max}$ after which the current. The filtering image pixel retains constant since there is not noise.
- b) Alternatively, if an image pixel is corrupted and larger than I_{max} or I_{min} . Level A median is performed to filtered image pixels.

3.3. Face Detection

This technique recognizes a set of data to create a human face in an image. Because the observed results are dependent on a number of factors, including the context, lighting, movement, orientation, and facial emotions, facial recognition in images is challenging. Face identification and face landmark drawing are simple to perform using Media Pipe [25]. The face identification model from Media pipes, which uses an image or video as the input and recognizes faces in real-time, is used in the suggested technique.

3.3.1. Landmark Detection

As a result, face landmarks may be accurately and consistently identified. The necessary measurements were obtained by mapping a few exact images onto individual photographs of faces. Features were automatically found and extracted using the suggested method. In the current study, the Media pipe library was used to extract 468 landmarks rather than 68 to increase accuracy. To identify many faces and the number of face landmarks in a three-dimensional region, the media pipe structure was used. Even when using a mobile device, our face geometry solution makes real-time face landmark estimations.

3.3.2. 3D Face Mesh Generation

To create a face mesh, landmarks from 2D facial images were put over the 3D image. Face mesh detection API produces each face has 468 3D triangles, edges, and points.

3.4. Segmentation using Unet architecture

In this work the segmentation is done by using U-Net architectures to effectively separate the foreground (face) from the background, thereby reducing noise and improving detection precision. By giving each pixel in an image a name, the computer vision task termed image segmentation divides an image into many regions [26]. Compared to object detection, which encloses the identified item in a bounding box, or image classification, which labels the object, it offers considerably more information about an image.

The model design is rather straightforward: a decoder with skip connections for upsampling and an encoder for downsampling. The term U-Net comes from the fact that it resembles the letter U, as shown in Figure 2.

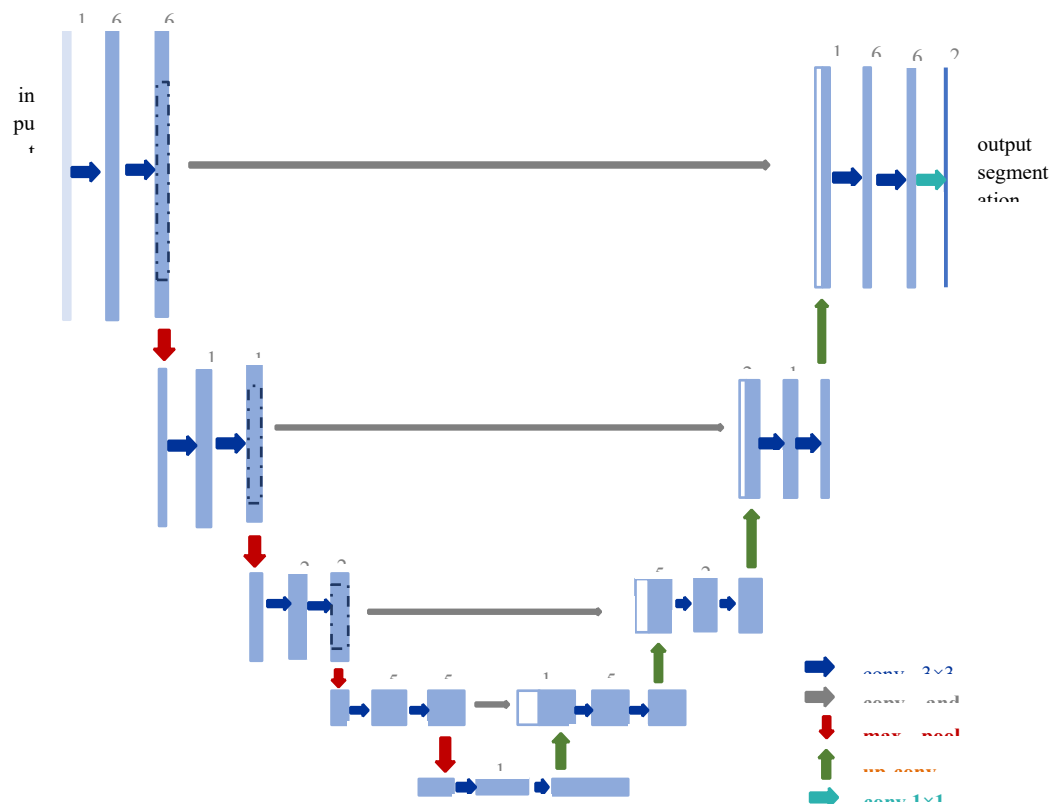


fig.2. U-Net architecture

A classifier, encoding, skip connections, bridging, and decoding are some of the components that make into figure 2 illustrates the network structure of our study. A single enhanced residual block makes to the bridging area, four enhanced residual blocks are included in each of the encoding and decoding sections, respectively. Batch normalization (BN), a 3×3 convolutional layer, a dynamic rectified linear unit (Dy-ReLU), and an identity mapping component makes to an enhanced deep residual block. The parts that follow will go into further depth about these components. Global max pooling is the main downsampling technique used in the encoding area, which is mostly composed deep residual blocks that have been enhanced and down sampling procedures. Four operations are carried out, with a global max pooling step after each deep residual block. The image size gradually decreases as these processes go, which causes the number of channels to increase at the same time.

As a necessary connection between the encoding and decoding portions of the network architecture, bridging plays an essential function.

The enhanced deep residual blocks and upsampling processes, which make up the decoding region, comprise the majority of the four operations. Following upsampling, the image size doubles and the number of channels decreases in half. As a result, an image's size that approximates the input feature image is eventually obtained. Using the Sigmoid function together with 1×1 convolutions, the classifier uses traditional techniques. The main objective of using 1×1 convolutions is to lessen the computational cost. To show pixel categories, the Sigmoid function is used to map the feature map.

The fusion of feature mappings is accomplished by concatenating deep and shallow-level features extracted from the encoding using skip connections. On the other hand, a large quantity of redundant feature information is added since feature information collected from the encoding region is not very effective. In this study, both linear and geodesic distances are used to compute the feature vectors.

3.5. Feature Extraction

Through this procedure, raw data from the most unique position is converted into digital information, numbers, or particulars. The extracted attributes were utilized to identify individuals and only included discriminating information. Both geodesic and Euclidean distances, as well as their ratios, were included as characteristics in this investigation. Using the Anthropometric Model to determine the distances between face geometric points, both Euclidean and geodesic, feature extraction for the entire face is suggested. The Euclidean distances and their ratios between face fiducial sites were calculated, and the curve and curvature properties provide the basis for the geodesic distance. The suggested method uses the sample Linear and Geodesic Distances for facial analysis in four areas of the face: the chin, nose, lips, and forehead/eyes. Anthropometric Reference Points, or 468 landmarks, are used to assess faces in a wide sense.

All measurements were taken, including mouth-to-nose ratio, nose width, nose height, eyebrow/eye width, lip height, chin height, and breadth. The suggested method's linear and geodesic distances are defined in Table 2. Using 468 face landmarks that had previously been found, they measured the geodesic and Euclidean distances, and the corresponding ratios were computed.

Table 1. The proposed research uses linear and geodesic distances

Landmarks	Facial Distance	Distance Type
Ft-Ft	Forehead width	Geodesic
Ex-Ex	Outer canthal width	Geodesic
Ex-Ch	Upper cheek height	Geodesic
Tr-G	Forehead height	Geodesic
Sbal-Sbal	Alar base width	Linear
N-Prn	Nasal bridge length	Geodesic
N-Sn	Nose height	Geodesic
Sn-Prn	Nasal tip protrusion	Geodesic
Sn-Sto	Upper lip height	Geodesic
Sn-Sto	Upper lip height	Linear
Ls-Li	Lip height	Geodesic
Ls-Li	Lip height	Linear
Chi-Chi	Mouth breadth	Geodesic
Sn-Gn	Lower facial height	Geodesic
Enl-Enr	Intercanthal width	Geodesic
Zy-Zy	Bizygomatic breadth	Geodesic
Zy-Zy	Bizygomatic breadth	Linear
Go-Go	Bigonial breadth	Geodesic

1) Euclidean distance is used to extract the characteristics

Euclidean distance between P and Q equals line segment length between PQ, as shown in Equation (2).

$$D(P, Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2} \quad (2)$$

where

- where a single point's coordinates are (p1, q1).
- The coordinates of additional points are (p2, q2).
- Between (p1, q1) and (p2, q2), d is the distance.

Equation (3) provides the distance in three-dimensional Euclidean space.

$$d(p, q) = \sqrt{\sum_{i=1}^3 (p_i - q_i)^2} \quad (3)$$

2) Feature extraction using geodesic distance

In surface terms, the shortest route between two places is a geodesic curve. X, y, and z coordinates define a three-dimensional point. Equation (3) states that the difference between the relevant coordinates, squared as the sum of squares is the shortest distance in three dimensions between places A (x1, y1, z1) and B (x2, y2, z2).

$$d(P, Q) = \sum_{i=1}^n \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2 + (z_{i+1} - z_i)^2} \quad (4)$$

where the number of landmarks is indicated by n.

3.6. Classification using improved convolutional neural network (ICNN)

Convolutional neural networks (CNNs) are among the most potent deep networks; they may include many hidden layers that execute low to high input data properties extraction, use convolution and subsampling [27]. As shown in figure 3, this kind of network is essentially composed of three layers: filled connection layers,

convolution layers, and subsampling or pooling levels. Feature extraction and classification are a CNN's two core parts. Feature extraction involves an activation function, max-pooling, and many convolution layers. Typically, the classifier is made from multiple layers that are completely connected.

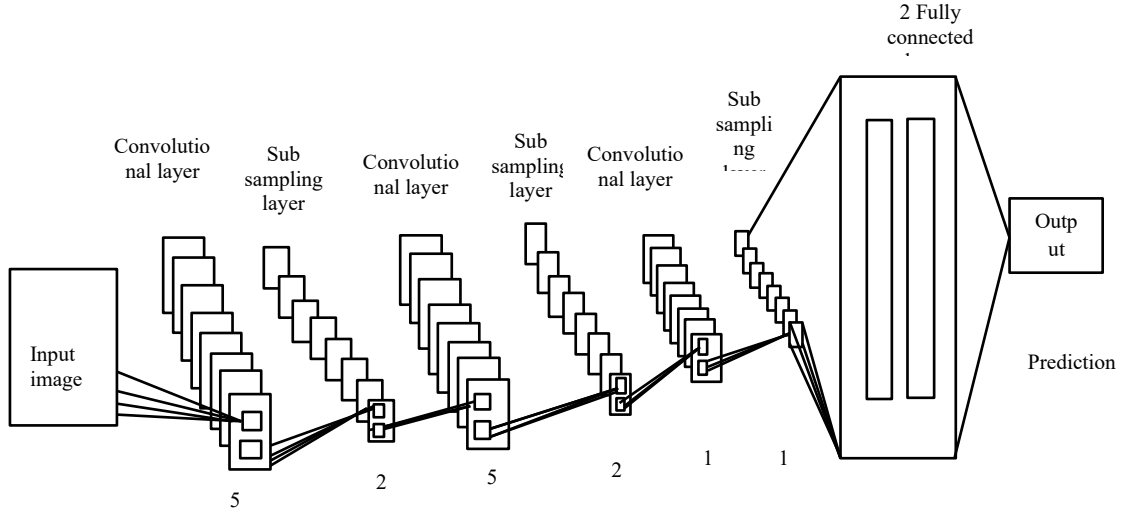


Figure 3: Convolutional Neural Network

i) Convolution layer

The detected images of faces are used as input in this suggested study. CNN feature extraction is done using a convolution layer. Activation functions and convolution operations are examples of linear and nonlinear processes that are often combined in this process. A kernel (filter) convolves input information in this convolution layer. The result of the convolutional layer is n output features which can be detected anywhere on input images. In general, the output features that are created by convolving the kernel and as the convolution matrix's kernel can be identified as a filter, the input is identified as feature maps of size $i \times i$.

Multiple convolutional layers may be included in the CNN; the features are the successive convolutional layers' inputs and outputs. There are n filters in each convolution layer. After filter convolution with input, feature map depth ($n \times$) equals number of convolution filters. Each input position has a distinct filter map.

Feature maps make the l -th output of the convolution layer, shown as $C_i^{(l)}$. It is determined as

$$C_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{a_i^{(l-1)}} K_{i,j}^{(l-1)} * C_j^{(l)} \quad (5)$$

where $K_{i,j}^{(l-1)}$ a kernel or convolution filter of size $a \times a$ combines the i -th and j -th feature maps in the same layer ($l - 1$), and $B_i^{(l)}$ is the bias matrix. In the output $C_i^{(l)}$ layer, feature mappings are present. (5) uses $C_i^{(0)} = X_i$ as input space for the first convolutional layer, $C_i^{(l-1)}$. The kernel creates a feature map. The activation function may be used after the convolution layer to change the convolutional layer's outputs nonlinearly.

$$Y_i^{(l)} = Y(C_i^{(l)}) \quad (6)$$

Where,

$Y_i^{(l)}$ - the activation function's output

$C_i^{(l)}$ - data that it obtains

Tanh, sigmoid, and rectified linear units (ReLUs) are the activation functions that are most often utilized. introduced ReLUs in this paper, which are represented as

$$Y_i^{(l)} = \max(0, Y_i^{(l)}) \quad (7)$$

The capacity to lessen interaction and nonlinear effects makes this function an attractive option for deep learning models. If the input is positive, the same value is returned by ReLU; if it is negative, it changes the output to 0. The advantage of this activating function over others is that it allows for faster training since the error component is extremely tiny in the saturating region, making weight updates almost invisible. The vanishing gradient problem is the name given to this issue.

ii) Sub sampling or pooling Layer

This layer's major objective is to geographically reduce the convolution layer's features map's dimensions. Successful subsampling occurs between feature mappings and masks. The mask and feature maps are subsampled after selecting a $b*b$ mask size.

$$X_j^l = f(\beta_j^l \text{down}(X_j^{l-1}) + b_j^l) \quad (8)$$

where the sub-sampling function is represented by $\text{down}(\cdot)$. In both spatial dimensions, to provide an output that is n times less, this function usually adds up each unique n -by- n feature in the input t . Each result map has an additive bias b and a multiplicative bias β .

iii) Fully Connected layer

In classification tasks, a subset of fully connected layers downsamples the convolution and pooling layers' features to the network's result, such as class probability. It uses Softmax activation in the output layer.

$$Y_i^{(l)} = f(z_i^{(l)}) \quad (9)$$

$$\text{where } z_i^{(l)} = \sum_{j=1}^{m_i^{(l-1)}} w_{i,j}^{(l)} y_j^{(l-1)} \quad (10)$$

Where, $w_{i,j}^{(l)}$ is the weights that the whole connected layer uses, and is the transfer function that depicts the instability should adjust to provide the representation of each class. It should be noted that, similar to association and pooling layers, an entirely connected layer generates nonlinearity within its neurons.

3.5.1. Parameter selection using Improved Inertia Weight based Whale Optimization Algorithm (IIWWOA)

The success rate is often impacted by directly increasing the number of training samples. Another element that improves the models' performance is adjusting the deep learning hyperparameters. Important parameters for training options include shuffle type,

momentum, l_2 regularization parameter, starting learning rate, maximum epochs, and small batch size. IIWVOA selected the almost ideal values for these factors in this study. The CNN architecture is then trained use the modified hyperparameter values and the training set. To evaluate the recommended technique's efficacy, the testing set is finally categorized using a trained model.

Mirjalili & Lewis proposed the WOA, a meta-heuristic optimization method. This method, Inspiration from nature, resembles some of the world's largest mammals [28]. The humpback whale bubble-net strategy inspired WOA, a swarm-based approach uses exclusively for hunting in the ocean and was developed based on their social behavior. Typically, the largest group of baleen whales is the humpback whale, which spends its days with its fellows. By producing bubbles around their prey in a spiral pattern, near the surface, they pursue small fish and smaller groups of krill. They then swim up to the top along this route. Here, we will examine and analytically simulate the three common behaviours of humpback whales.

Encircling of Prey: The hyperparameters X_i ($i= 1, 2, 3, \dots, n$) are taken into consideration as the population of whales. These parameters are defined by shuffle type, momentum, l_2 regularizations, starting learning rate, maximum epochs, and small batch size. One definition of a fitness function is the recognition error. When they can find their prey, humpback whales can surround them. WOA assumes the best candidate solution is near the optimum or has the desired optimal parameter values. After determining the best search agent, others will alter their positions to pursue it. The representation of this behavior

$$\vec{D} = |\vec{C} \cdot \vec{X}^*(t) - \vec{X}(t)|, \quad (11)$$

$$\vec{X}(t+1) = \vec{X}^*(t) - \vec{A} \cdot \vec{D} \quad (12)$$

Equation (12) suggests and updates the time-varying inertia weight to enhance the WOA's performance.

$$\text{Inertia weight } \omega = (\omega_1 - \omega_2) \left(\frac{Iter_{max} - Iter}{Iter} \right) + \omega_2 \quad (13)$$

$$\vec{X}(t+1) = \omega \cdot \vec{X}^*(t) - \vec{A} \cdot \vec{D} \quad (14)$$

The letter t indicates the most repeats that are permitted, which represents the current iteration is denoted by $Iter_{max}$. The beginning and final values of the inertia weight are represented by ω_1 and ω_2 , respectively. $\vec{A}$ and $\vec{C}$ stand for the vectors of coefficients. The location vector is represented by $\vec{X}$, and the X^* represents the optimal solution's location vector that has been found so far. It is necessary to update X via iteration if a better option is available.

The following is the computation of the vectors $\vec{A}$ and $\vec{C}$:

$$\vec{A} = 2 \cdot \vec{a} \cdot \vec{r} - \vec{a} \quad (15)$$

$$\vec{C} = 2 \cdot \vec{r} \quad (16)$$

where r represents the random vector in the interval [0, 1] and $\vec{a}$ is reduced with the number of repeats in a straight line from two to zero.

The bubble-net technique is also used by humpback whales as was mentioned in the previous part, to attack an individual. This approach is expressed mathematically as follows.

Exploitation phase: Also by another name, bubble net assault, it operates via the two processes listed below:

Shrinking encircling technique: The unpredictability of $\vec{A}$ in $[-1, 1]$, between the agent's previous positioning and the best position at the moment, is where we may find the search agent.

Spiral updating position: The amount of space that divides the whale and its prey, which are situated at (X, Y) and (X^*, Y^*) , respectively, is first determined in this case. Following that, a spiral equation connecting the prey's and whale's positions is produced as follows:

$$\vec{X}(t+1) = \vec{D}' \cdot e^{bl} \cdot \cos \cos(2\pi l) + \omega \cdot \vec{X}^*(t) \quad (17)$$

Exploration phase: The objective of global optimization is aided by WOA. Based on their random positions in relation to one another, the whales search for their prey (Fig. 3). $\vec{A}$ must be more than 1 or less than -1 as it is randomly set from $[-1, 1]$ to enable movement from the reference whale by the search agent. They may also execute a search agent's most recent position by randomly selecting a WOA global search agent. A mathematical formula models this exploratory mechanism:

$$\vec{X}(t+1) = \vec{X}_{rand} - \vec{A} \cdot \vec{D} \quad (18)$$

For a random whale from the population, $\vec{X}_{rand}$ denotes a random position.

Algorithm 1: IIWWOA

1. Initialize the hyper parameters X_i ($i = 1, 2, 3, \dots, n$).
2. Determine the parameters' fitness.
3. X^* = optimal hyperparameter values should be set.
4. while ($t < \text{maximum number of iterations}$) do
5. for (each hyper parameter) do
6. Update a, A, C , and p .
7. if ($p < 0.5$) then
8. if ($|A| < 1$) then
9. Equation (12) updates the current search position.
10. Else if ($|A| \geq 1$) then
12. Select the X_{rand} random parameter.
13. Equation (16) is used to update the position.

14. End if
16. else if ($p \geq 0.5$) then
18. Use Equation (18) to modify the position.
19. End if
20. End for
22. Find and modify search agents that exceed the search space
23. Determine the search agent's fitness.
24. Update X^* if a better option is accessible.
25. Output near-optimal hyperparameters
26. $t = t+1$
27. End while

4. Results and Discussion

The proposed facial expression recognition is implemented in Matlab. On <https://www.kaggle.com/datasets/jessicali9530/lfw-dataset>, the suggested approach is tested. The LFW face image database was created to study unrestricted recognition of faces. The Acknowledgments section specifically acknowledges the researchers from University of Massachusetts, Amherst produced and maintains this database. The Viola Jones face detector catalogued 13,233 images of 5,749 individuals online. Collection includes two or more images of 1,680 individuals. Four LFW and three "aligned" sets were in the initial database. Concerning recall, f-measure, accuracy, and precision, the proposed Improved Convolutional Neural Network based face recognition outperforms previous adaptive boosting (AB) and Random Forest (RF).

1. Accuracy

The weighed percentage of facial expressions is correctly identified by the measurement accuracy. It is represented as,

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (19)$$

Where,

TP-True Positive

TN-True negative

FP-False Positive

FN-False Negative

2. Precision

The proportion of successfully predicted successful results to all of expected observations is referred to as precision.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (20)$$

3. Recall

In the original class, recall is the proportion of all observations to anticipated positive results.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (21)$$

4. F-Measure

The weighted mean of accuracy and recall is used to get the F-measure.

$$F = 2 \frac{P \times R}{P + R} \quad (22)$$



Figure 4. Collection of input image from the dataset

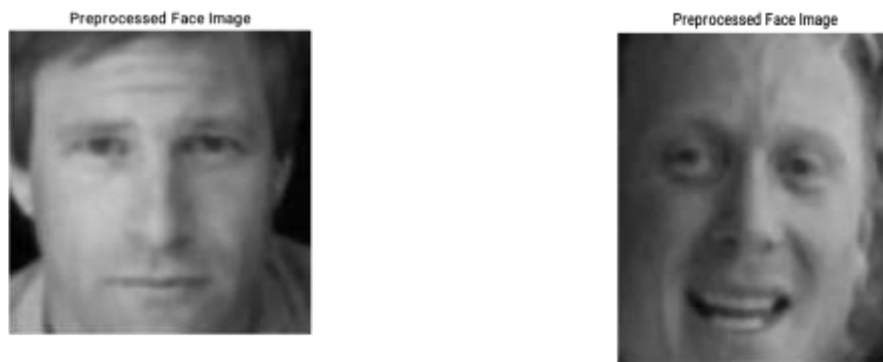


Figure.5. Pre-processed image

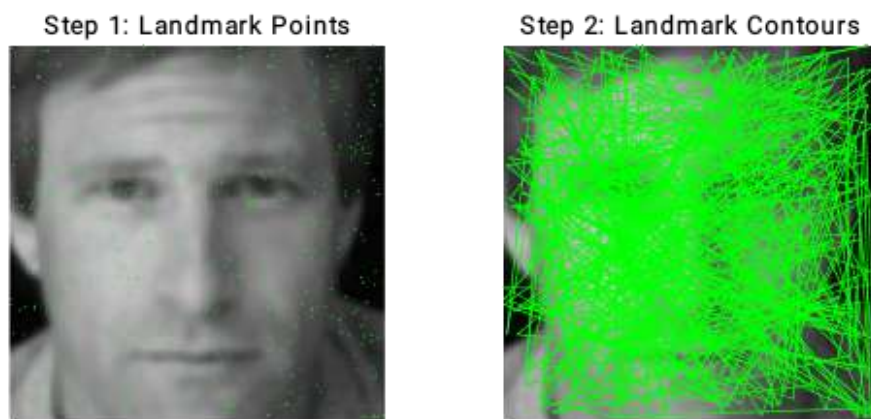


Figure 6. landmark points and land mark contours

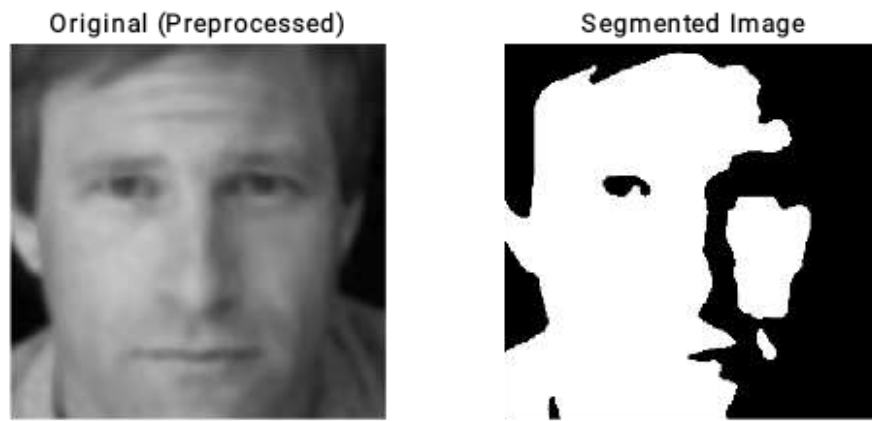


Figure 7. original and segmented image

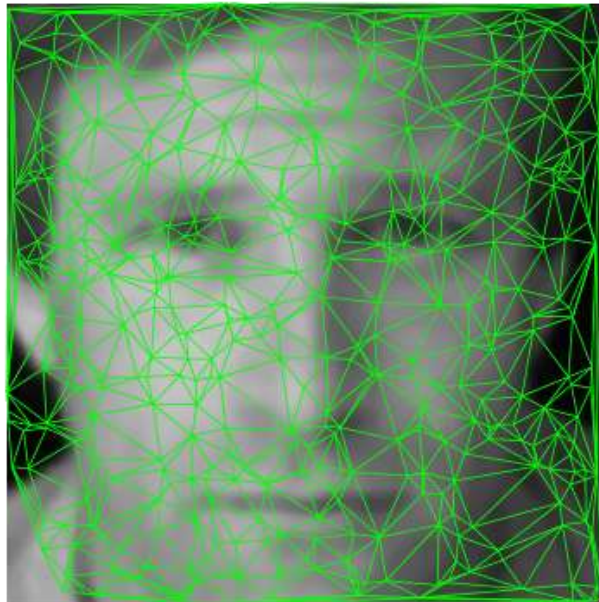


Figure 8. face landmarks



Figure 9. output images

Table 1. performance results between the proposed and existing methods

Performance metrics (%)	AB	RF	ICNN
Precision	81	90.09	95.77
Recall	84	92.2	93.41
F-measure	83	91.12	94.58
Accuracy	85.12	93	94.636

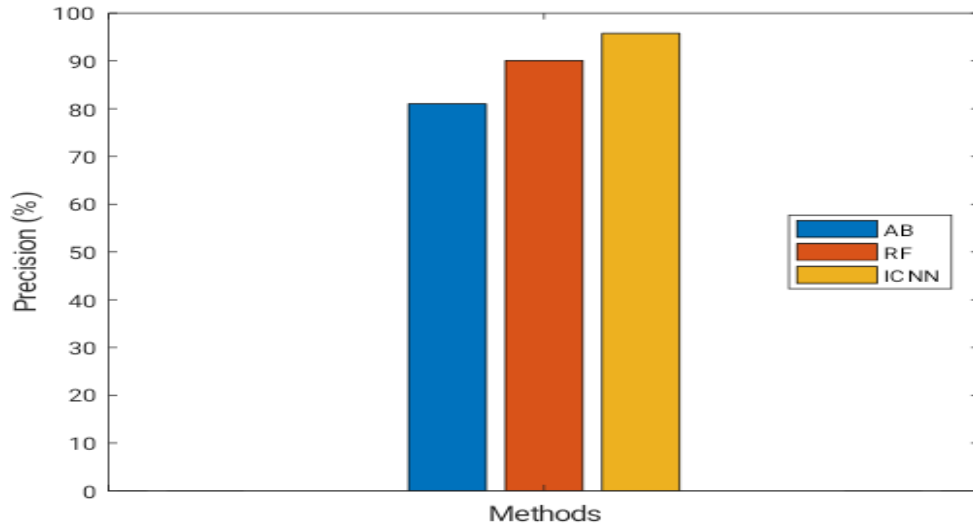


Fig.10. Precision comparison of the proposed ICNN

In the Fig.10. shows how the suggested and existing methods for face image recognition compare in terms of accuracy. In comparison to the previously considered in general, the results of machine-learning algorithms indicated that the ICNN model performed more effectively on the datasets. This is consistent with the error rate that was previously produced and can be ascribed to the non-redundant rule sets that the proposed classifier model generated. The findings indicate that the suggested ICNN technique has high precision results compare to the existing classification techniques.

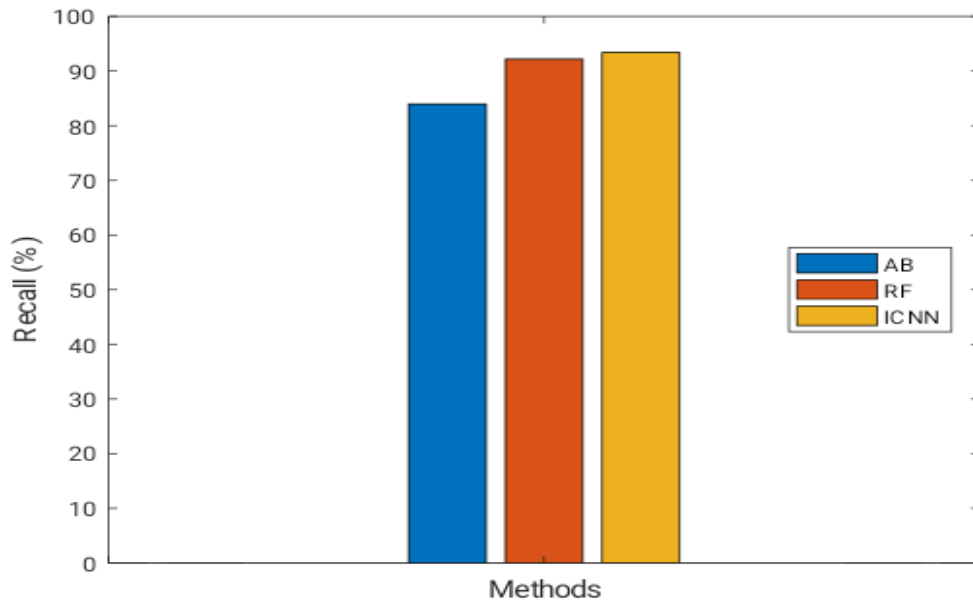


Fig.11. Recall comparison of the proposed ICNN

In the Fig.11. shows how the suggested and current methods for identifying the facial image compare in recall. Using the ICNN technique, various images are used to train this model for each applicant. The result was a massive dataset that increased the accuracy overall. The findings of the analysis indicate that one of the factors influencing the identification process is the light conditions. Low light intensity causes the recognition system to perform erroneous recognition. Adding additional training images that were taken in low light levels while creating the face classifier may solve. And also the parameter optimization using improved whale optimization increases the recall ratio.

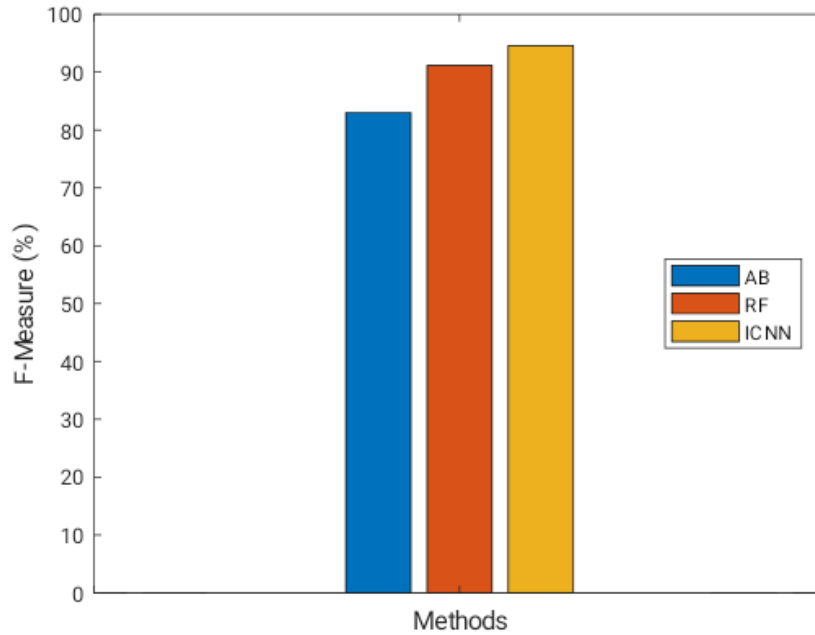


Fig.12. F-measure comparison of the proposed ICNN

The F-measure comparison results between the suggested and current methods for facial image recognition are shown in Fig. 12. According to the test, the variable in the LFW database has the highest correlation with the target class, according to both feature-extraction and classification methods. Comparing the suggested model to the other machine-learning models, the comparison above demonstrates that it has the highest precision rate measurement in the database. The best f-measure results among the datasets are found in the LFW database. It is evident from the graph that the suggested ICNN model surpasses existing techniques in terms of f-measure.

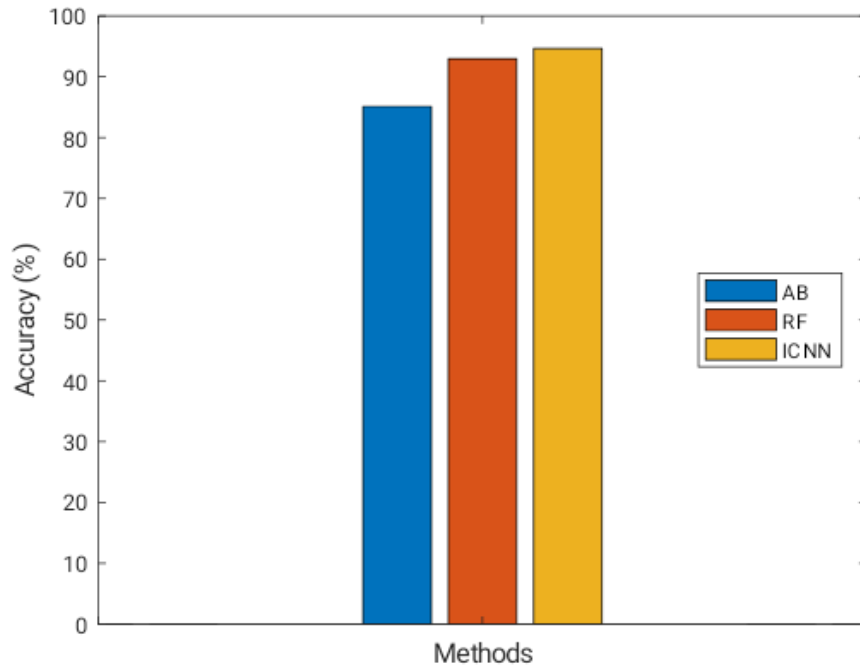


Fig.13. Accuracy comparison of the proposed ICNN model

The recommended and present methods for accuracy comparison for facial image recognition is shown in Fig. 13. When a deep learning model can accurately predict the objective and generalize predictions to new examples, it is considered successful. Generally, accuracy which includes two subtypes: sensitivity and specificity can be used to assess a model's validity. From the results it concludes that the proposed ICNN technique has high accuracy results compare to the existing classification techniques. The suggested ICNN strategy provides the highest accuracy when compared to other existing techniques. Additionally, the total success rate of the facial recognition system increases by this preprocessing technique. The proposed ICNN yields improved accuracy results of 92.85% whereas RF technique yields is 91.24% and AB technique provides 89.15%,

5. Conclusion

This study uses feature vectors and face detection and identifying system development using deep learning. This study introduces a novel framework for face recognition that leverages two-dimensional (2D) facial images sourced from diverse origins to construct three-dimensional (3D) face meshes using Media Pipe landmarks. The preprocessing pipeline incorporates adaptive median filtering to enhance image quality, followed by deep learning-based segmentation via U-Net architectures to effectively separate the foreground (face) from the background, thereby reducing noise and improving detection precision. Feature vectors are generated by calculating Euclidean and the ratios of geodesic distances between recognized landmarks, capturing intricate facial geometries. These feature vectors are subsequently trained using an Improved Convolutional Neural Network (ICNN) classifier, designed to maintain high accuracy in environments characterized by significant pose variations, facial expressions, and occlusions. It describes the general process of creating data training for this facial recognition system utilizing the ICNN technique to face recognition. It has been shown that a classifier can

recognize image data with an accuracy of 91.7% when it is taught a large amount of facial images. The accuracy of the system may be impacted by a limited number of circumstances. Insufficient light intensity results in relatively low accuracy as compared to higher light intensity. Apart from that, the classifier is key to recognition. Classifier performance improves with training time. Therefore, a future study might focus on photo-to-sketch matching utilizing a deep learning method in conjunction with a transfer learning strategy, which could prove to be a new work. If the study in this field is completed, it may be possible enhance the system's identical twin recognition.

6. REFERENCES

1. Huang, D., Shan, C., Ardabilian, M., Wang, Y., & Chen, L. (2011). Local binary patterns and its application to facial image analysis: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(6), 765-781.
2. Alblushi, A. (2021). Face recognition based on artificial neural network: A review. *Artificial Intelligence & Robotics Development Journal*, 116-131.
3. Vasilescu, M. A. O., & Terzopoulos, D. (2002, August). Multilinear image analysis for facial recognition. In *2002 International Conference on Pattern Recognition (Vol. 2, pp. 511-514)*. IEEE.
4. Huang, D., Shan, C., Ardebilian, M., & Chen, L. (2011). Facial image analysis based on local binary patterns: A survey. *IEEE Transactions on Image Processing*.
5. Kakadiaris, I. A., Passalis, G., Theoharis, T., Toderici, G., Konstantinidis, I., & Murtuza, N. (2005, June). Multimodal face recognition: Combination of geometry with physiological information. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) (Vol. 2, pp. 1022-1029)*. IEEE.
6. Li, Y. (2022). Research on neural network algorithm in artificial intelligence recognition. *Sustainable Energy Technologies and Assessments*, 53, 102691.
7. Le, T. H. (2011). Applying artificial neural networks for face recognition. *Advances in Artificial Neural Systems*, 2011(1), 673016.
8. Nagi, J., Ahmed, S. K., & Nagi, F. (2008, March). A MATLAB based face recognition system using image processing and neural networks. In *4th International Colloquium on Signal Processing and its Applications (Vol. 2, pp. 83-8)*.
9. Al-Dabbas, H. M., Azeez, R. A., & Ali, A. E. (2024). Two proposed models for face recognition: Achieving high accuracy and speed with artificial intelligence. *Engineering, Technology & Applied Science Research*, 14(2), 13706-13713.
10. Martindenz-Diaz, Y., Luevano, L. S., Mendez-Vazquez, H., Nicolas-Diaz, M., Chang, L., & Gonzalez-Mendoza, M. (2019). Shufflefacenet: A lightweight face architecture for efficient and highly-accurate face recognition. In *Proceedings of the IEEE/CVF international conference on computer vision workshops (pp. 0-0)*.
11. Wen, Y., Zhang, K., Li, Z., & Qiao, Y. (2016). A discriminative feature learning approach for deep face recognition. In *Computer vision-ECCV 2016: 14th European conference, amsterdam, the netherlands, October 11-14, 2016, proceedings, part VII 14 (pp. 499-515)*. Springer International Publishing.
12. He, Y., Zhang, Y., Chen, S., & Hu, Y. (2023). Facial expression recognition using hierarchical features with three-channel convolutional neural network. *IEEE Access*, 11, 84785-84794.
13. Tian, Y. (2020). Artificial intelligence image recognition method based on convolutional neural network algorithm. *Ieee Access*, 8, 125731-125744.
14. Aggarwal, A., Alshehri, M., Kumar, M., Sharma, P., Alfarraj, O., & Deep, V. (2021). Principal component analysis, hidden Markov model, and artificial neural network

- inspired techniques to recognize faces. *Concurrency and Computation: Practice and Experience*, 33(9), e6157.
15. Soni, N., Sharma, E. K., & Kapoor, A. (2021). Hybrid meta-heuristic algorithm based deep neural network for face recognition. *Journal of Computational Science*, 51, 101352.
 16. Liu, W., Zhou, L., & Chen, J. (2021). Face recognition based on lightweight convolutional neural networks. *Information*, 12(5), 191.
 17. Sun, X., Wu, P., & Hoi, S. C. (2018). Face detection using deep learning: An improved faster RCNN approach. *Neurocomputing*, 299, 42-50.
 18. Rastgoo, R., & Kiani, K. (2019). Face recognition using fine-tuning of Deep Convolutional Neural Network and transfer learning. *Journal of Modeling in Engineering*, 17(58), 103-111.
 19. Teoh, K. H., Ismail, R. C., Naziri, S. Z. M., Hussin, R., Isa, M. N. M., & Basir, M. S. S. M. (2021, February). Face recognition and identification using deep learning approach. In *Journal of Physics: Conference Series* (Vol. 1755, No. 1, p. 012006). IOP Publishing.
 20. Hangaragi, S., & Singh, T. (2023). Face detection and Recognition using Face Mesh and deep neural network. *Procedia Computer Science*, 218, 741-749.
 21. Singh, N. S., Hariharan, S., & Gupta, M. (2020). Facial recognition using deep learning. In *Advances in data sciences, security and applications: Proceedings of ICDSSA 2019* (pp. 375-382). Springer Singapore.
 22. Sanil, G., Prakash, K., Prabhu, S., Nayak, V. C., & Sengupta, S. (2023). 2d-3d facial image analysis for identification of facial features using machine learning algorithms with hyper-parameter optimization for forensics applications. *IEEE Access*, 11, 82521-82538.
 23. Verma, Kesari, Bikesh Kumar Singh, and A. S. Thoke. "An enhancement in adaptive median filter for edge preservation." *Procedia Computer Science* 48 (2015): 29-36.
 24. Ibrahim, Haidi, Nicholas Sia Pik Kong, and Theam Foo Ng. "Simple adaptive median filter for the removal of impulse noise from highly corrupted images." *IEEE Transactions on Consumer Electronics* 54.4 (2008): 1920-1927.
 25. M. Zubair. (Jan. 6, 2022). Face Detection With Mediapipe| Towards Data Science. [Online]. Available: <https://towardsdatascience.com/write-a-fewlines-of-code-and-detect-faces-draw-landmarks-from-complex-imagesmediapipe-932f07566d11>
 26. Ibtehaz, N., & Rahman, M. S. (2020). MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation. *Neural networks*, 121, 74-87.