Published by :
http://www.ijert.org

# Facial Expression Recognition using HOG, DWT and Sine Cosine Algorithm-Optimized Neural Network Classifier

Madhavi Ravikumar Upasani
Department of E&TC Engineering,
Gangamai College of Engineering, Nagaon
Dhule, Maharashtra, India

Jagruti Sarang Patil
Department of E&TC Engineering,
Gangamai College of Engineering, Nagaon
Dhule, Maharashtra, India

***Abstract*— Facial recognition is a challenging task for artificial intelligence, despite the human retina's apparent ease in recognizing faces. Variations and complexities such as noise, rotation, and lighting conditions often affect the images captured by cameras, complicating the recognition process. While many techniques utilize algorithms to match facial models with test images, achieving consistent high accuracy across different applications and image sources remains difficult. This is because a single algorithm cannot universally deliver optimal performance under all conditions. Even the best algorithms face challenges in various aspects of face recognition.**

**This paper explores a hybrid approach that integrates Histogram of Oriented Gradients (HOG) and Discrete Wavelet Transform (DWT) features for facial expression recognition. The proposed method constructs a feature subspace for training, and employs a SCO optimized Neural network classifier to compute similarity scores for performance evaluation. By leveraging the complementary strengths of both HOG and DWT features, this hybrid approach aims to enhance recognition accuracy and improve overall performance in facial expression recognition tasks.**

***Keywords*— DWT, JAFFE, LBP, SCO, NN,Viola Jones.**

## I. INTRODUCTION

Biometric recognition systems involve measuring, storing, and comparing specific individual characteristics. Key biometric traits include fingerprints, facial images, hand and finger geometry, iris patterns, retina patterns, signatures, and voice. For a biometric feature to be effective, it should meet several criteria: universality (applicable to all users), uniqueness (distinct for each user), invariance (consistent under various conditions), and resistance (against fraud attempts). The choice of biometric method largely depends on the application and user acceptance. For example, while signatures are widely accepted socially, retina scans using low-intensity infrared beams often provoke skepticism among users [1].

Biometric systems are developed to complement traditional authentication methods that rely on knowledge-based elements (such as PINs or passwords) or possession-based elements (like magnetic cards). These traditional methods are vulnerable to loss, theft, or forgetfulness, whereas biometric characteristics are unique and permanent to each individual. This inherent permanence, along with the speed of extraction, makes biometric methods valuable for personal identification and automatic systems. However, current biometric systems face challenges including precision issues (false acceptance and rejection rates), limitations for certain individuals who lack specific biometric traits, vulnerability in some contexts, and user acceptance concerns. Despite these limitations, biometric recognition serves as an effective supplementary authentication method. Even basic and cost-effective biometric solutions can significantly enhance overall security when used alongside traditional methods and tailored to the specific needs of the application [2].

Despite appearing straightforward to human eyes, face recognition is a complex task for computational systems. Effective face recognition schemes need to incorporate sufficient parameters to identify faces accurately and remain robust against noise. One simple approach involves matching the pixels of a test image with those in a database image at corresponding positions. If the proportion of matched pixels exceeds a defined threshold, the face is considered authentic. However, practical challenges arise when input images are not in a standard position for matching. For instance, images captured from security cameras positioned at a higher altitude may show faces from different angles, leading to tilts that complicate recognition with straightforward methods.

Another significant issue is the quality of the input images. Variations in face size due to different distances from the camera and changes in skin color due to lighting conditions can affect recognition accuracy [1]. Given these variations, algorithms must perform with high accuracy under varying conditions. Addressing these issues involves considering both the angle of the face captured and the computational model used. Most systems currently employ 2D face recognition models, which can limit accuracy based on the face's angle and the rotational extent of the image. The effectiveness of mathematical algorithms in detecting faces in noisy 2D images has been a focus of study [2]. While many methods claim robustness and high efficiency, their assumptions often remain unvalidated in practical scenarios.

Previous research [3] demonstrated the use of Principal Component Analysis (PCA) and Euclidean distance for face recognition, resulting in lower accuracy (93.57%). In contrast, the present study utilizes a Support Vector Machine (SVM) Classifier-based approach to calculate similarity scores for performance evaluation, showing improved results in recognition accuracy.

## II. PROPOSED METHOD

Face recognition using extraction of HOG (Histogram Gradient Orientation) and DWT (Discrete Wavelet Transform) features is proposed in this heading. In domain of inductive inference problem, source separation could be a challenging task. As to derive the solution one needs the sufficient information, the available information is exploited in maximum limits. The adaptive systems tend to inherit most of the available feature information to replicate the original set of input with elaborated clarity. The efficiency of an algorithm is subjected to its performance in case when evaluation parameters reflect sound values even in cases of noise, orientation and luminance conditions.

This section proposes a face recognition method that combines Histogram of Oriented Gradients (HOG) and Discrete Wavelet Transform (DWT) features. Source separation, a common challenge in inductive inference problems, requires maximizing the use of available information. Adaptive systems are designed to utilize as much feature information as possible to achieve accurate reconstruction of the original input with enhanced clarity.

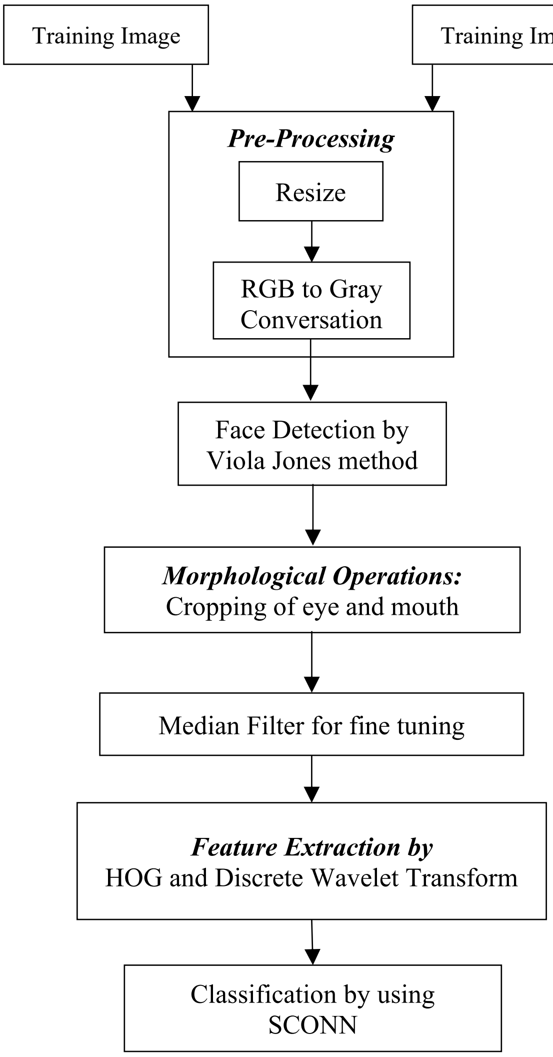

Training Image
Training Image
Pre-Processing
Resize
RGB to Gray Conversation
Face Detection by Viola Jones method
Morphological Operations: Cropping of eye and mouth
Median Filter for fine tuning
Feature Extraction by HOG and Discrete Wavelet Transform
Classification by using SCONN

Fig 1: Flow diagram for proposed approach

The effectiveness of the proposed algorithm is evaluated based on its performance under various conditions, such as noise, orientation, and luminance variations. The algorithm's ability to maintain high performance despite these challenging factors demonstrates its robustness and reliability in practical scenarios.

Rest of methodology is explained as follows:

### A. Pre-Processing of Face

Algorithm-1

- The input photograph is resized using MATLAB's built-in resize function, adjusting the image dimensions to 250×250 pixels. The resized image is then saved in the JPEG data folder.
- If the input image is in color, it is converted to grayscale using the rgb2gray function after resizing.

The Viola Jones Method for Face Detection
The Viola-Jones algorithm, developed by Paul Viola and Michael Jones in 2001 [4], is a real-time face detection method. This algorithm utilizes Haar features in a classifier to detect faces efficiently.

### B. Morphological Operations

Mathematical morphology processes images with the help of previously chosen special shapes, which are generally smaller than the image and are called the structural element, which acts as an operator on an image to produce a result.

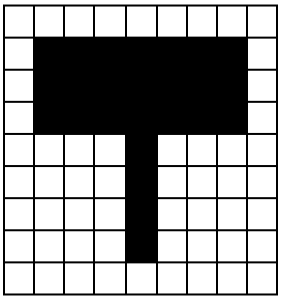
Dilation*:* Dilation, also called expansion, filling, or growth, produces a thickening effect on the edges of the object. This algorithm is used to increase the contour of the objects and to join the discontinuous lines of these, produced by some filtration, mathematically the binary dilation is defined as:

$$A \oplus B = \left\{c \in E^{N|\,C=a+}b\; for\; all\; a \in A\; and\; b \in\; B\right\} \qquad (3)$$

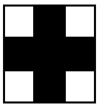
*Erosion:* Erosion is the dual function of expansion, but it is not the reverse, i.e. if erosion is done and then a dilation the resulting image will not be equal to the actual image, mathematically erosion is defined as:

$$A \ominus B = \{x \in E^N | x + b \in A\; for\; all\; b \in B \qquad (4)$$

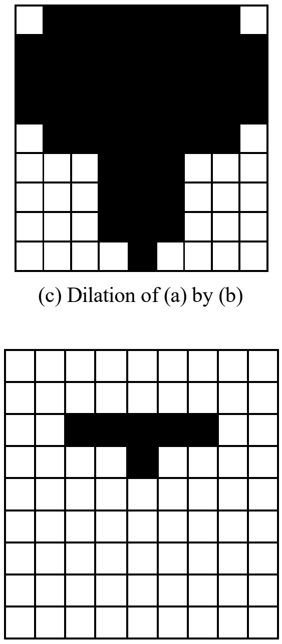
Figure 5 gives an example of dilation and erosion operations with the given structuring element.



(a) Original image



(b) Structuring element called simple cross



(c) Dilation of (a) by (b)

(d) Erosion of (a) by (b)

Figure 5: Illustration of morphological operations [5]

## C. Median Filter

The principle of the Median filter is often defined in the case of a discrete image, whose practical implementation is direct. From the discrete concepts, it is possible to give a continuous version, which will be adapted to the theoretical study of some of its properties

Consider a discrete image $F$ characterized by a gray level $f(x,y)$. Let $V(x_0,y_0)$ be the neighborhood associated with the coordinate point $(x_0,y_0)$; it is assumed that this neighborhood has $N$ coordinate pixels $(x_0-u,y_0-v)$ with odd N.

Let $\{f_1,f_2,\ldots,f_i,\ldots,f_{N-1},f_N\}$ the gray levels associated with the $N$ pixels of $V(x_0,y_0)$.

The median filtering first proceeds by sorting the gray level values of the neighborhood followed by a selection of the middle element of the sorting.

Sorting is done in ascending order generally. It leads to the ordered set of gray values of the neighborhood of $f(x_0,y_0)$. Since the ordered elements are denoted by $f_i$, the ascending sort is characterized by:

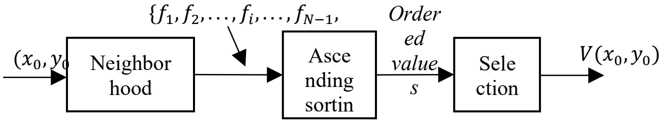


Fig 6: Median Filter

$$f_q < f_2 < \cdots < f_{\frac{N+1}{2}} < \cdots < f_{N-1} < f_N \qquad (5)$$

The middle element of the neighborhood is $f_{\frac{N+1}{2}}$. Its property is to be preceded by $\frac{N-1}{2}$ lower values and followed by as many higher values.

The filtering consists of replacing $f(x_0,y_0)$ by the median value of the neighborhood $f_{\frac{N+1}{2}}$ [6].

## D. Feature Extraction

The purpose of feature extraction in the field of recognition is to express the feature in numerical or symbolic form called encoding. Depending on the case, the values of these features can be real, integer or binary. The vector composed of feature n represents a point in the new space of n dimensions. The steps involved in feature extraction are shown in the flow diagram in Figure 1. We use two feature extraction methods:

- DWT (Discrete Wavelet Transform)
- HOG (Histogram Oriented Gradients)

The feature extraction technique captures the local gradient information in an image to represent its visual content, making it particularly effective in describing object shape and texture. HOG features have been widely used in computer vision tasks, including image segmentation.

HOG (Histogram Oriented Gradients)

The process of extracting HOG features for image segmentation involves the following steps:

Image Preprocessing: The input image is typically preprocessed to enhance its quality and remove noise. This may include converting the image to grayscale and applying normalization techniques to improve contrast and remove variations in lighting.

Gradient Computation: Gradients are calculated to capture the intensity changes or edges in the image. This is done by applying gradient operators such as the Sobel operator in both the horizontal $(\mathrm{G_x})$ and vertical $(\mathrm{G_y})$directions. The magnitude (M) and direction (θ) of the gradient at each pixel are computed using the following equations:

$$M = \sqrt{\left(G_x^2 + G_y^2\right)} \qquad (7)$$

$$\theta = atan2(G_y, G_x) \qquad (8)$$

The $atan2$ function takes the ratio of the vertical gradient $(G_y)$ to the horizontal gradient $(G_x)$ as arguments and returns the corresponding angle. The resulting gradient orientation ranges from $-\pi$ to $\pi$ or 0 to $2\pi$, depending on the conventions used.

Cell Division: The image is divided into small cells, typically of size 8x8 or 16x16 pixels. Each cell represents a local region of the image.

Orientation Binning: Within each cell, the gradient orientations (θ) are quantized into a discrete number of bins covering the full range of possible angles (e.g., 0-180 degrees or 0-360 degrees). The magnitude of the gradient is assigned to the corresponding bin based on its orientation. This process creates a histogram of orientations for each cell.

Block Normalization: To capture local spatial information and provide robustness to lighting variations, neighboring cells are grouped into larger blocks. The block size and overlap are typically defined as parameters. Within each block, the histograms of neighboring cells are concatenated, and normalization techniques such as L1-norm or L2-norm are applied to the concatenated histograms.

Descriptor Extraction: The normalized block-level features are concatenated to form the final HOG descriptor, representing the global structure and texture information of the image. This descriptor can be used as input for various machine learning algorithms to perform image segmentation or other computer vision tasks.

By utilizing HOG features, image segmentation algorithms can leverage the information captured in local gradients to identify object boundaries and regions of interest within the image.

Discrete Wavelet Transform (DWT):

DWT employees Fourier transform to convert time domain image into frequency domain. The mathematical expression of DWT is given by:

$$DWT_{x(n)} = \begin{cases} dd_{j,k} = \sum img(n)hh^{*}{}_{s}(n-2^{s}r) \\ ap_{j,k} = \sum img(n)ll^{*}{}_{s}(n-2^{s}r) \end{cases} \qquad (9)$$

Where, $dd_{j,k}$ represents detailed coefficients.

$ap_{j,k}$ are the approximate coefficients of DWT transform.

$hh(n)$ are high pass filter functions.

$ll(n)$ low pass filter functions.

$s$ is wavelet scale parameters.

$r$ is translation factor.

## E. Similarity Measure using Sine Cosine Algorithm-Optimized Neural Network Classifier

Feature Vector Integration**:**

After extracting features using HOG and DWT, the individual feature vectors are concatenated to form a combined feature vector. Let $F_{HOG}$ and $F_{LBP}$ represent the feature vectors obtained from HOG and DWT, respectively. The combined feature vector $F_{combined}$ is constructed by concatenating these vectors:

$$F_{combined} = [F_{HOG}, F_{dwt}] \qquad (10)$$

Where $[\cdot,\cdot]$ denotes concatenation. If $F_{HOG}$ has $\eta_{HOG}$ dimensions and $F_{LBP}$ has $\eta_{Dwt}$ dimensions, then $F_{combined}$ will have $\eta_{combined} = \eta_{HOG} + \eta_{dwt}$ dimensions.

Feature Fusion for Classification:

The combined feature vector $F_{combined}$ is then used as input to the classification algorithm. This unified feature set provides a comprehensive representation of the image, capturing both gradient and texture information. The improved feature set helps the classifier to better distinguish between different classes or disease stages, enhancing the accuracy and robustness of the classification model.

Sine Cosine Algorithm-Optimized Neural Network Classifier

The Sine Cosine Algorithm (SCA) is a nature-inspired optimization technique used to enhance the performance of neural network classifiers. It optimizes the parameters of the neural network to achieve better classification results. This section provides a detailed description of the SCA-optimized neural network classifier, including the optimization process and its integration into neural network training.

Neural Network

A neural network classifier consists of multiple layers: input, hidden, and output layers. Each layer comprises neurons (nodes) that process the input data and produce predictions. The neural network can be mathematically represented as:

$$y = \sigma(W_L \cdot \sigma(W_{L-1} \cdots \sigma(W_1 \cdot x + b_1) \cdots + b_{L-1}) + b_L) \qquad (11)$$

Where:

$x$ is the input feature vector.

$W_i$ and $b_i$ are the weight matrix and bias vector for layer $i$.

$\sigma(\cdot)$ is the activation function, such as ReLU or sigmoid.

$y$ is the output vector representing class probabilities.

Sine Cosine Algorithm (SCA)

The SCA is inspired by the sine and cosine functions, which mimic the natural behavior of animal movement and are used to optimize continuous functions. The SCA optimizes the neural network parameters by iteratively updating them based on the sine and cosine functions.

Initialization: Let $W_t$ denote the weights of the neural network at iteration$t$. Initialize the weights randomly within a specified range:

$$W_0 = Random(Range) \qquad (12)$$

*SCA Update Rules:* The SCA updates the weights based on sine and cosine functions to explore the solution space. The update rules are as follows:

*Sine and Cosine Update:* The weight update for each iteration $t$ is given by:

$$W_{t+1} = W_t + A_t \cdot \sin(B_t \cdot C_t) \cdot (W_{best} - W_t) \qquad (13)$$

Where:

$A_t$ is the amplitude of the sine function, which decreases over time to ensure convergence:

$$A_t = A_0 \cdot \left(1 - \frac{t}{T}\right) \qquad (14)$$

$B_t$ is the frequency of the sine function, which adjusts the exploration rate:

$$B_t = B_0 \cdot \left(1 - \frac{t}{T}\right) \qquad (15)$$

$C_t$ is a random vector with values between 0 and 1, which introduces randomness in the update process.

$W_{best}$ represents the best solution found so far.

Position Update: Additionally, the position update rule is given by:

$$W_{t+1} = W_t + A_t \cdot \cos(B_t \cdot C_t) \cdot (W_{best} - W_t) \qquad (16)$$

Termination Criteria: The SCA iteration continues until a stopping criterion is met, such as a maximum number of iterations T or convergence to a satisfactory solution.

Integration with Neural Network Training
The optimized weights obtained from the SCA are used to train the neural network. The training process involves minimizing a loss function, such as cross-entropy loss, which measures the difference between predicted probabilities and actual class labels:

$$L(y, \hat{y}) = -\sum_{i=1}^{C} y_i \cdot \log(\hat{y}_i) \qquad (17)$$

Where $y$ is the true label vector, $\hat{y}$ is the predicted probability vector, and $C$ is the number of classes.

Back propagation: The gradient of the loss function with respect to the network parameters is computed using back propagation:

$$\nabla_W L = \frac{\partial L}{\partial W} \qquad (18)$$

The gradients are used to adjust the weights of the neural network, which are optimized further by the SCA.

Optimization***:*** The neural network is trained using the optimized weights, and performance is evaluated using metrics such as accuracy, precision, recall, and F1-score.
The integration of SCA with neural network training enhances the classifier's performance by providing an optimized set of weights that improves the network's ability to classify diabetic retinopathy stages accurately.

## III. SIMULATION AND RESULTS

The simulation is carried out by using image processing toolbox of MATLAB software.
The Japanese Female Facial Expression (JAFFE) Database is a widely used dataset in facial expression recognition research. It consists of 213 images depicting 7 different facial expressions (6 basic expressions—anger, disgust, fear, happiness, sadness, and surprise—plus a neutral expression). These images feature 10 Japanese female models and were taken at the Psychology Department of Kyushu University. Each image has been rated on 6 emotion adjectives by 60 Japanese subjects, providing valuable data for studies in emotion recognition and analysis. This dataset is often utilized for training and evaluating facial expression recognition systems.
Figure 10: Test images for JAFFE (Japanese female facial expression) [10]

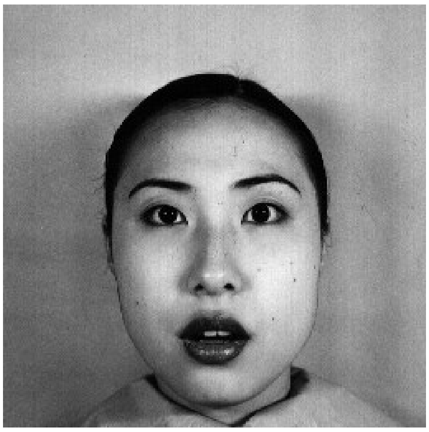


Fig 7: Input image

Fig 8: Resized to 224×224

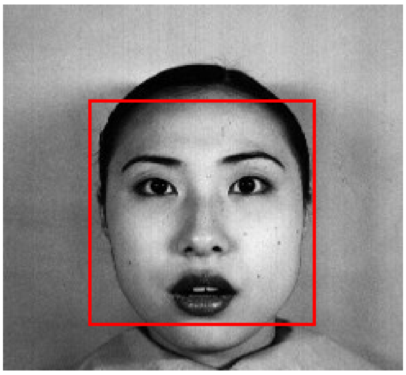


Figure 13: Face detection using Viola Jones method

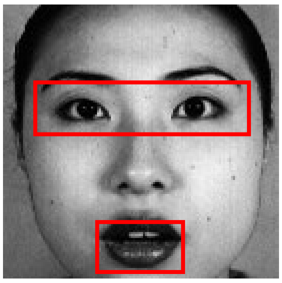
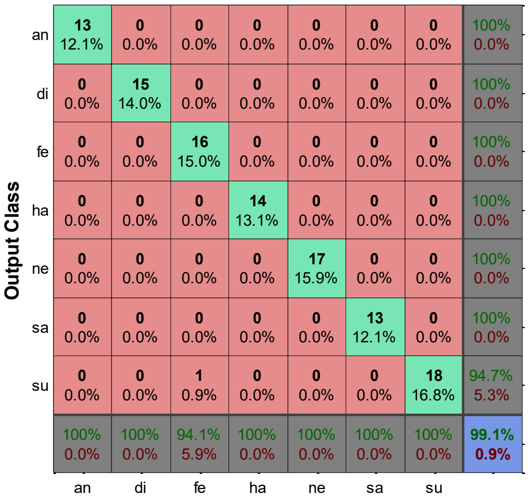


Figure 15: Eye and mouth detection

In proposed support vector machine classifier based approach there is not any threshold value for face recognition. SVM itself does the similarity measure and recognizes test image. Finally, confusion matrix plot show the performance of LBP and DWT based method.



| | an | di | fe | ha | ne | sa | su | |
|---|---|---|---|---|---|---|---|---|
| an | 13<br>12.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| di | 0<br>0.0% | 15<br>14.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| fe | 0<br>0.0% | 0<br>0.0% | 16<br>15.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| ha | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 14<br>13.1% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| ne | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 17<br>15.9% | 0<br>0.0% | 0<br>0.0% | 100%<br>0.0% |
| sa | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 13<br>12.1% | 0<br>0.0% | 100%<br>0.0% |
| su | 0<br>0.0% | 0<br>0.0% | 1<br>0.9% | 0<br>0.0% | 0<br>0.0% | 0<br>0.0% | 18<br>16.8% | 94.7%<br>5.3% |
| | 100%<br>0.0% | 100%<br>0.0% | 94.1%<br>5.9% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 100%<br>0.0% | 99.1%<br>0.9% |

Figure 18: Confusion Matrix plot for proposed approach

The row and column are the classes of facial expression database. There are 7 sets of classes and each class having different set of expressions. The confusion matrix plot indicates the accuracy i.e. 99.1% for proposed algorithm.

Table 1: Notations for expression

| S. No. | Abbreviation | Meaning |
|---|---|---|
| 1 | An | Angry |
| 2 | Di | Disgust |
| 3 | Fe | Fear |
| 4 | Ha | Happy |
| 5 | Ne | Neutral |
| 6 | Sa | Sad |
| 7 | Su | Surprise |

## IV. CONCLUSION

Facial expression recognition (FER) plays a crucial role in security and other technical applications. While initially developed for security purposes, such as identifying individuals who may attempt to disguise their appearance, FER technology has expanded into various everyday applications, including smile detection in cameras and interactive systems. The importance of FER in security contexts underlines the need for continued research in this area. FER presents several challenges, particularly in artificial intelligence, due to the complexity of human emotions and the subtlety of facial expressions. This complexity necessitates robust and accurate methods for recognizing and classifying facial expressions.

In this paper, we present a facial expression recognition system that leverages a combination of Histogram of Oriented Gradients (HOG) and Discrete Wavelet Transform (DWT) features. These features are classified using a Random Forest classifier. Our approach is evaluated using a confusion matrix, demonstrating that the proposed SCO Optimized Neural network based method achieves higher accuracy compared to previous research efforts. This improvement highlights the potential of our hybrid feature extraction technique in enhancing the performance of FER systems.

## REFERENCES

[1] Li, S., & Deng, W. (2020). Deep facial expression recognition: A survey. IEEE transactions on affective computing, 13(3), 1195-1215.

[2] Ge, H., Zhu, Z., Dai, Y., Wang, B., & Wu, X. (2022). Facial expression recognition based on deep learning. Computer Methods and Programs in Biomedicine, 215, 106621.

[3] Niu, B., Gao, Z., & Guo, B. (2021). Facial expression recognition with LBP and ORB features. *Computational Intelligence and Neuroscience*, *2021*(1), 8828245.

[4] Viola, Paul, and Michael Jones. "Rapid object detection using a boosted cascade of simple features." In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, vol. 1, pp. I-I. IEEE, 2001.

[5] Liao, J., Lin, Y., Ma, T., He, S., Liu, X., & He, G. (2023). Facial expression recognition methods in the wild based on fusion feature of attention mechanism and LBP. Sensors, 23(9), 4204.

[6] Yaddaden, Y., Adda, M., & Bouzouane, A. (2021, February). Facial expression recognition using locally linear embedding with lbp and hog descriptors. In 2020 2nd International Workshop on Human-Centric Smart Environments for Health and Well-being (IHSH) (pp. 221-226). IEEE..

[7] Abdulsattar, N. S., & Hussain, M. N. (2022). Facial expression recognition using HOG and LBP features with convolutional neural network. *Bulletin of Electrical Engineering and Informatics*, *11*(3), 1350-1357.

[8] Zhang, W., & Xiang, S. (2020). Face anti-spoofing detection based on DWT-LBP-DCT features. Signal Processing: Image Communication, 89, 115990.

[9] Guenther, Nick, and Matthias Schonlau. "Support vector machines." Stata J 16, no. 4 (2016): 917-937.

[10] Hassan, S. M., Alghamdi, A., Hafeez, A., Hamdi, M., Hussain, I., & Alrizq, M. (2021). An effective combination of textures and wavelet features for facial expression recognition. Engineering, Technology & Applied Science Research, 11(3), 7172-7176.

[11] Lakshmi, D., & Ponnusamy, R. (2021). Facial emotion recognition using modified HOG and LBP features with deep stacked autoencoders. Microprocessors and Microsystems, 82, 103834.

[12] Alaluosi, W. M. (2021). Recognition of human facial expressions using DCT-DWT and artificial neural network. Iraqi Journal of Science, 2090-2098.S

[13] anchez-Mendoza, David, David Masip, and Agata Lapedriza. "Emotion recognition from mid-level features." Pattern Recognition Letters 67, pp. 66-74, 2015