ISSN: 2093-5374 / E-ISSN: 2093-5382

Facial Emotion Recognition and Intensity Estimation in Human Face Images Using RSRN and STFIS

R. Virupaksha Gouda[1*], and Dr. Yeresime Suresh[2]

[1*]Research Scholar, Assistant Professor, Department of Computer Science & Engineering, Ballari Institute of Technology and Management, Ballari, Karnataka, India; Visvesvaraya Technological University, Belagavi, Karnataka, India. gouda.viru@gmail.com, https://orcid.org/0009-0001-4638-3604

[2]Professor & Head of the Department, Department of Computer Science & Engineering - Artificial Intelligence, Ballari Institute of Technology and Management, Ballari, Karnataka, India; Visvesvaraya Technological University, Belagavi, Karnataka, India. dr.suresh@bitm.edu.in, https://orcid.org/0000-0002-8372-3612

Received: October 06, 2025; Revised: December 02, 2025; Accepted: January 05, 2026; Published: March 31, 2026

Abstract

Introduction: Facial Emotion Recognition (FER) is vital to the study of human emotion, as it enables the analysis of facial expressions. Nevertheless, current FER techniques do not effectively identify emotions in people with facial wrinkles, which are common in older people, leading to incorrect classifications. This research aims to recommend a superior FER model that considers facial wrinkles as a way of improving the identification of emotions and the degree of emotion. Methodology: The hybrid methodology is introduced. Processing of input photos begins with noise removal and contrast enhancement. Facial features are then derived using the Logarithmic Function-centric Viola-Jones (LF-VJ) algorithm, which detects faces and segments facial objects. The wrinkle score and cosine similarity are computed to measure the existence of wrinkles. Entropy Weighted Secretary Bird Optimization (EWSBO) is used to do optimal feature selection and then RSRN is used to classify emotions. STFIS has to do intensity estimation to assess the intensity of identified emotions. Results: The accuracy of the proposed framework on emotion classification was 98.97 and precision, recall and F-measure values were beyond 99. The estimation of the intensity was correct and rule generation, fuzzification and defuzzification time were 958.697ms, 923.013ms and 848.998ms, respectively. It was able to efficiently deal with wrinkles, lighting effects and lessen the complexity of features and was much faster and more accurate than current methods. Conclusion: It is proven in the current study that wrinkles in the FER frameworks enhance the classification accuracy and emotion strength prediction greatly, and thus the offered approach is appropriate to be applied to real time applications, such as mental health monitoring.

Keywords: Ridge Scaled Residual Network (RSRN), Sig-Tangent Fuzzy Inference System (STFIS), Polar Gradient-based Contrast Limited Adaptive Histogram Equalization (PG-CLAHE), Facial Emotion Recognition, Intensity Estimation, Feature Extraction, Emotion Classification.

Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), volume: 17, number: 1 (March - 2026), pp. 425-448. DOI: *10.58346/JOWUA.2026.I1.024*

*Corresponding author: Research Scholar, Assistant Professor, Department of Computer Science & Engineering, Ballari Institute of Technology and Management, Ballari, Karnataka, India; Visvesvaraya Technological University, Belagavi, Karnataka, India.

1 Introduction

Facial expressions act as a basic medium for non-verbal communication, thereby conveying identity, emotions, and psychological states (Chowdary et al., 2023). The facial expressions, which play an essential role in human interactions (Naga et al., 2023; Gautam & Seeja, 2023), are driven by muscle movements and convey emotions, namely happiness, sadness, anger, fear, and disgust. FER recognizes human emotions by examining facial landmarks and muscle activity, thereby offering valuable insights into the mental and emotional states of individuals (Saxena et al., 2022). This process involves capturing video frames or facial images, thus detecting key features and interpreting expressions centered on their variations and intensity (Shahabinejad et al., 2021; Sati et al., 2020). Emotion intensity estimation, which is an important aspect of FER, measures the strength of an expressed emotion (Do et al., 2021). For example, a broad smile represents more intense happiness, whereas a slight smile may exemplify mild happiness (Veltmeijer et al., 2021). Accurate intensity estimation improves FER applications, especially in fields like mental health monitoring Revina & Emmanuel, 2021. The accuracy and efficiency of FER have been significantly enhanced by recent advancements in Machine Learning (ML) and Deep Learning (DL) (Salama et al., 2021; Tan et al., 2021). For emotion detection, numerous DL models, including Graph Convolutional Network (GCN), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM), were broadly utilized (Sheng & Li, 2021). Furthermore, to improve classification performance, ML models like Random Forest and Support Vector Machine (SVM) were explored Li & Deng, 2020. Nevertheless, regardless of these advancements, prevailing methods faced challenges like high computational cost, drawbacks in real-time applications, and sensitivity to variations, thus diminishing their practicality in dynamic environments (Li & Lima, 2021; Bhagat et al., 2024). Moreover, the prevailing approaches failed to consider wrinkles in the facial region, thereby causing inaccurate FER. Hence, an enhanced FER and intensity estimation approach using RSRN and STFIS is proposed in this paper by considering wrinkles for improved accuracy.

Problem Statement

Conventional approaches to facial emotion recognition (FER) have a number of disadvantages. The older population has more crevices hence tends to be classified wrongly as they are not differentiated in the traditional methods. The approach used by Mohan et al., 2020 was inhibited by the lack of attention to the angle variations within the regions of eyebrows and other lower face areas. The use of only global features but not local features reduced the classification accuracy of Liu et al., (2021). Sikkandar & Thiyagarajan, 2021 did not consider the level of emotion intensity which is important to mental health assessment in real-time. Also, most approaches were not focused on lighting effects or were very time-consuming because of combination of features.

Objectives

The proposed framework will be useful in improving facial emotion recognition (FER) by solving most of the problems. It enhances better classification by computing wrinkle score and a patch similarity. The Bezier curve construction procedure is used to capture the angle change of facial objects. With the implementation of Band Block Transformation (BBT), the framework encompasses both the global and local features to make the recognition more specific. It also examines the intensity of emotion through STFIS technique. In the framework, the lighting effects are taken care of using Polar Gradient-based Contrast Limited Adaptive Histogram Equalization (PG-CLAHE), and less feature combination complexity is taken care of using EWSBO technique, which makes the processing more efficient.

The paper is structured as: The literature survey is demonstrated in Section 2, the proposed method is displayed in Section 3, the result and discussion are provided in Section 4, and finally, the proposed work is concluded in Section 5 with future suggestions.

2 Literature Survey

A facial expression recognition (FER) framework Mohan et al., (2020) was proposed using the Local Gravitational Force Descriptor (LGFD) within a Deep CNN (DCNN) architecture. The LGFD process was used to obtain local facial features, and the results were classified using a DCNN with score-level fusion, achieving an average accuracy of 98. Nevertheless, the model neglected differences in eyebrow shape and the angle of the lower face, which weakened its strength. A hybrid approach to feature extraction based on a Spatial Attention CNN and an Attention LSTM was proposed in Liu et al., (2021), which extracts expressive features and facial landmarks. The model was very accurate (95.15), but it focused more on global attributes and was very weak at capturing local facial attributes. In Sikkandar & Thiyagarajan, (2021), a DCNN-based FER system was created that optimally selected features and used SVM ensemble classification, achieving high precision. However, this model only predicted emotion categories rather than emotion intensity, and thus was not suitable for mental health monitoring. The four-layer CNN with Local Binary Pattern (LBP) features to FER in healthcare imaging was proposed by Debnath et al., (2022). Although a short training epoch can be used successfully, LBP's sensitivity to noise adversely affected feature extraction. Using Dynamic Local Ternary Patterns (DLTP) and a Kernel Extreme Learning Machine as its approach, Saurav et al., 2021 provide high performance and robust computation. But features of facial texture and shape were excluded, limiting expression. A 92.32% F1-score ResNet-50-based learner engagement detection system presented in Gupta et al., (2023) did not sufficiently analyse eye-movement-based expressions, but the framework was based on real-time prediction. In addition to the use of HOG and LBP features, (Lakshmi & Ponnusamy, 2021) integrated HOG and LBP features with Deep Stacked AutoEncoders and SVM in FER to classify emotions effectively but overfitting problems impacted the generalization. Muhammad & Hossain, 2021 found a way to implement a CNN-based edge-computing FER system, which allowed the recognition process to be performed with low latency, yet the model complexity did not allow for a widespread deployment of smartphones.

Recent deep learning solutions, such as Transformer-Augmented Networks (Ma et al., 2023), that have additionally improved relational facial feature modelling but need large training datasets, and Dynamic global–Static local Attention models (Xu et al., 2024), which showed higher accuracy but more complex computationally because of multiscale processing, were also developed.

Although the current methods have indeed come a long way in recognizing emotions, certain shortcomings exist, including that they are less specific about facial wrinkles and emotion intensity, which is difficult to compute without, and that the costs are high. These gaps make it clear that a more holistic approach to wrinkles and emotion intensity should be developed for real-time application, as discussed in the proposed framework in the current paper.

3 Proposed Methodology for Facial Emotion Recognition and Intensity Estimation

In this proposed framework, FER and intensity-level estimation are improved using RSRN and STFIS, which consider wrinkles in the face region. Pre-processing of input images, grayscale conversion and patching, calculation of wrinkle score and cosine similarity, face detection and facial object

segmentation, feature extraction and selection, facial point detection, band transformation, Beizure curve construction and angle estimation, emotion classification, and emotion intensity estimation are the main phases of the work. Figure 1 depicts the architectural diagram of the proposed framework.

Input Image

From the Affect Net dataset, the input images are gathered. This dataset comprises a variety of emotions. The gathered images($\Re$) are signified as equation 1,

$$\Re = \{\Re^1, \Re^2, \Re^3, \Re^4, \ldots\ldots, \Re^r\} \quad (1)$$

Where, the total number of($\Re$) is implied as(r).

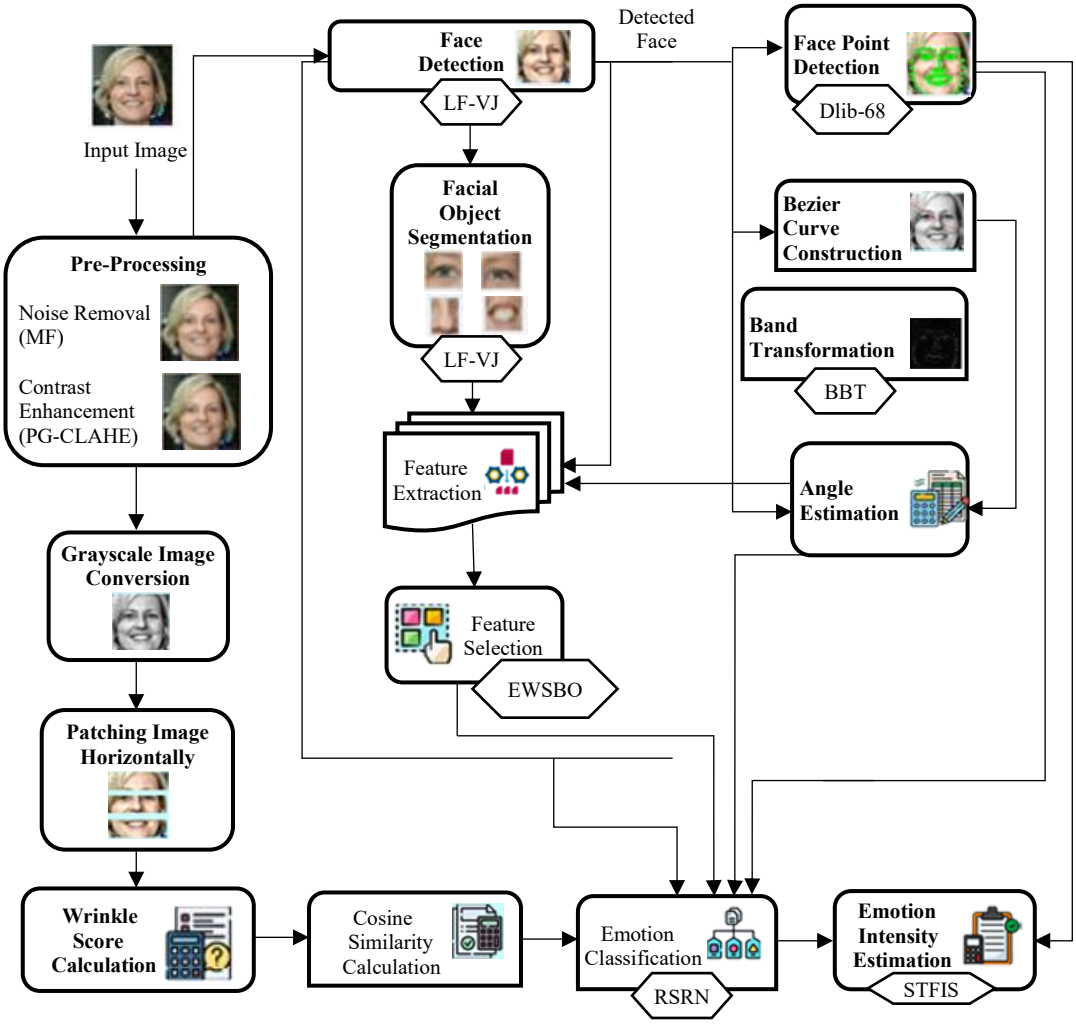


Figure 1: Architectural diagram of the proposed framework

Pre-processing

In this phase, to improve the accuracy of emotion classification, R data are pre-processed. The pre-processing steps include,

Noise Removal: Faulty pixels in ($\Re$) may cause the misrepresentation of the facial features which prevents the identification of emotions. The Median Filter (MF) uses the median of the neighbors instead

of the pixel value, and it essentially eliminates noise but still captures important features of the face that should be used to correctly identify emotions.

$$\eta = E^{med}\{\Re(m,n)|\ (m,n) \in \Upsilon\} \quad (2)$$

Equation 2 shows the noise removal and where the noise-removed image is notated as (η), the MF function that chooses the middle value from the set of neighboring pixel values(Υ) is exemplified as$\left(\Xi^{med}\right)$. The pixels in$(\Re)$ are implied as(m,n).

Contrast Enhancement: PG-CLAHE is used to improve contrast and overcome the problem of lighting in FER. It uses Contrast Limited Adaptive Histogram Equalization (CLAHE) to increase the visibility of the area without overexposing the area. The polar gradient of etching, ∇P (η (R, θ) is a function to tune the clipping threshold and tile size to avoid edge artifacts, and to provide a higher image quality. To dynamically adjust the tile size and clipping limit, $\nabla P\left(\eta(R,\theta)\right)$ is calculated, and it is equated in equation 3,

$$\nabla P\left(\eta(R,\theta)\right) = \check{e}_R \frac{\partial \eta}{\partial R} + \check{e}_\theta \frac{\partial \eta}{\partial \theta} \quad (3)$$

Where, the radial and angular gradients of(η) are notated as$\left(\frac{\partial \eta}{\partial R}\right)$ and$\left(\frac{\partial \eta}{\partial \theta}\right)$, respectively, the radial and angular coordinates in polar coordinates are exemplified as(R) and(θ), correspondingly, and $(\check{e}_R)$ and $(\check{e}_\theta)$ imply unit vector in the radial and angular directions, respectively. The tile size is adjusted from$\nabla P\left(\eta(R,\theta)\right)$, and (τ) signifies the divided tiles. Thereafter, for every single(τ), the histogram is computed. Here, $(\hbar)$ represents the computed histogram. Next, centered on$\nabla P\left(\eta(R,\theta)\right)$, a clipping limit$(L)$ is applied to $(\hbar)$ for preventing over-enhancement. The clipped histogram(Φ) is articulated as equation 4,

$$\Phi = min\left(\hbar, L\left(\nabla P\left(\eta(R,\theta)\right)\right)\right) \quad (4)$$

Therefore, dynamically adjusting the clipping limit and tile size using the Polar Gradient function solves the lighting issues and edge artifacts. Then, from$\left(\Phi\right)$, the Cumulative Distribution Function (CDF) is calculated to map pixel intensities for contrast equalization are shown in equation 5.

$$\Phi''' = \frac{\sum_{\tau=0}^{\ell} \Phi(\tau)}{\iota} \quad (5)$$

Here, the CDF output is notated as(Φ'''), the total number of pixels is implied as(ι), and the total number of tiles is exemplified as(ℓ). Afterward, the CDF is normalized to improve dynamic range and optimize contrast. The normalized CDF(μ) is articulated as equation 6,

$$\mu = \frac{\Phi''' - min(\Phi''')}{max(\Phi''') - min(\Phi''')} \quad (6)$$

Where, the minimum and maximum of(Φ''') are denoted as $min(\Phi''')$ and$max(\Phi''')$, respectively. Since the tiles are processed independently, boundary artifacts may happen. The interpolation (I) is applied to ensure smooth transitions, and it is depicted as equation 7,

$$I = (1-c)(1-d)\mu(m,n) + c(1-d)\mu(m+1,n) + cd\mu(m+1,n+1) + (1-c)d\mu(m,n+1) \quad (7)$$

Here, the fractional horizontal and vertical distances between two adjacent tiles are implied as (c) and(d), correspondingly, and (C) signifies the obtained contrast-enhanced image.

Algorithm 1 for PG-CLAHE

Input: Noise Removed Image(η)

Output: Contrast Enhanced Image(C)

Begin

Initialize$(\check{e}_R)$, $(\check{e}_\theta)$, (R),(θ)

For each (η)do

Compute Polar Gradient Function

$$\nabla P\big(\eta(R,\theta)\big) = \check{e}_R \frac{\partial \eta}{\partial R} + \check{e}_\theta \frac{\partial \eta}{\partial \theta}$$

Select tile(τ) size based on$\nabla P\big(\eta(R,\theta)\big)$

Calculate$(\hbar)$

Adjust clipping limit based on$\nabla P\big(\eta(R,\theta)\big)$

$$\Phi = min\left(\hbar, L\left(\nabla P\big(\eta(R,\theta)\big)\right)\right)$$

Compute CDF and normalize (Φ''')

$$\mu = \frac{\Phi''' - min(\Phi''')}{max(\Phi''') - min(\Phi''')}$$

Perform interpolation(I)

End for

Return $(\boldsymbol{C})$

End

Lastly, (χ) represents the pre-processed images.

PG-CLAHE (Algorithm 1) improves the contrast of an image (η) with noise removed by using Polar Gradient function ∇P (η (R, θ)). It calculates the gradient, picks a suitable tile size, and adjusts the clipping threshold to avoid overexposure. Normalization of the CDF is followed by interpolation, yielding the contrast-enhanced image (C).

Grayscale Conversion

Thereafter, $(\chi)$is transformed into a grayscale image to extract wrinkle-based information, which is essential for FER. The grayscaled image(G) is calculated as equation 8,

$$G = \frac{\zeta(\chi) + \zeta'(\chi) + \zeta''(\chi)}{3} \tag{8}$$

Here, the red, green, and blue channel values for each pixel in(χ) are notated as $\zeta(\chi), \zeta'(\chi),$ and $\zeta''(\chi)$, correspondingly.

Patching Image Horizontally

In this section, $(G)$is divided into three equal horizontal sections in which every single section is treated as an individual patch for feature extraction. This permits localized analysis of different regions of the

image. Here, ($\wp$) represents the patched images. Next, for detailed investigation, features like the Gabor filter, LBP, and edge density features are extracted from($\wp$). These extracted features are signified as(E).

Wrinkle Score Calculation

After that, to quantify the existence and intensity of wrinkles in an image, the wrinkle score is computed from(E) by combining multiple texture-based descriptors. Wrinkles, which reflect changes in skin texture, are small lines or creases on the skin caused by aging, facial expressions, or environmental factors. To identify emotions, wrinkles are essential. Hence, the wrinkle score(ϖ) is computed as equation 9,

$$\varpi = \alpha * \varsigma^{gab} + \alpha' * \varsigma^{lbp} + \alpha'' * \varsigma^{edge} \quad (9)$$

Where, the weight parameters assigned grounded on their significance in detecting wrinkles are exemplified as($\alpha, \alpha', \alpha''$), the Gabor filter is notated as$(\varsigma^{gab} \in E)$, the LBP is implied as$(\varsigma^{lbp} \in E)$, and the edge density is indicated as$(\varsigma^{edge} \in E)$. This wrinkle score is used for FER, in which the distribution of wrinkles can signify expressions, such as sadness, anger, surprise, and so on.

Cosine Similarity Calculation

Thereafter, to measure the similarity in texture characteristics across image patches, Cosine similarity is calculated between the wrinkle score values(ϖ) of each patch. The Cosine similarity(S) is expressed as equation 10,

$$S = \frac{\varpi' * \varpi''}{\|\varpi'\| \|\varpi''\|} \quad (10)$$

Here, the wrinkle score vectors for two patches from(ϖ) are implied as(ϖ') and(ϖ''), and the magnitude of the vectors is denoted as($\|\varpi'\|$) and ($\|\varpi''\|$).

Face Detection and Facial Object Segmentation

In this stage, LF-VJ is used to detect faces and facial objects to be separated out of. (χ) Viola-Jones (VJ) method is fast at recognizing faces, although its weight adjustment is exponential thus is susceptible to outliers. Learning is stabilized by the application of a logarithmic function update. Haar-like features (H) are then determined by computing pixel variance in neighbouring rectangular areas which are shown in equation 11.

$$H = \sum_{white} x(m,n) - \sum_{black} x(m,n) \quad (11)$$

Where, the sum of pixel intensities in the white and black regions is exemplified as$\sum_{white} \chi(m,n)$ and$\sum_{black} \chi(m,n)$, correspondingly. Subsequently, the integral image$(\hat{I})$ (also called the summed-area table) is utilized to speed up the computation of(H). It permits for quick calculation of the sum of pixel values in any rectangular region of the image.

$$\hat{I} = \sum_{m=0}^{m'} \sum_{n=0}^{n'} H(m,n) \quad (12)$$

In equation 12, the positions of$(\hat{I})$ are implied as(m', n'). Then, centered on$(\hat{I})$, the Haar-like features are calculated more effectively. After that, to choose the most informative features, the Adaptive Boosting (AdaBoost) classifier is used by iteratively training weak classifiers from$(\hat{I})$ and producing

them into a strong classifier. In this, for reducing the impact of outliers, a logarithmic function-centric weight update process is utilized in LF-VJ are shown in equation 13 and 14.

$$w = log\left(\frac{1}{1+err}\right) \quad (13)$$

$$A = sign\left[\sum_{0=1}^{\ddot{o}} w\Im(H)\right] \quad (14)$$

Where, the weak classifier is denoted as$(\Im)$, the strong classifier is exemplified as(A), the weight assigned to$(\Im)$ is implied as (w), $(log())$ notates logarithm, the classification error of$(\Im)$ is represented as(err), and the number and total number of weak classifiers are indicated as(o) and $(\ddot{o})$, correspondingly. Next, non-face regions$(\tilde{\lambda})$ in the image are rapidly rejected by the cascade classifier, thereby accelerating the face detection process by applying the strong classifier at multiple stages. The detected face (λ) is attained from(A), and it is equated as equation 15,

$$\lambda = \prod_{\tilde{r}=1}^{\iota'''} A_{\tilde{r}} \quad (15)$$

Here, the $(\tilde{r}^{th})$ strong classifier in the cascade is notated as$(A_{\tilde{r}})$, and the number of stages in the cascade classifier is implied as(ι''').

After detecting faces, the facial objects like eyes, nose, and mouth are segmented from(λ) utilizing LF-VJ, which is described above. Here, (Ψ) represents the segmented facial objects.

Facial Point Detection

For precise facial analysis, the facial points are detected from(λ) using the Dlib-68 function. A facial landmark detection model that recognizes 68 key facial points is known as the Dlib-68 function, which offers an accurate and detailed representation of facial features. Dlib-68 function captures key regions of the face, comprising the eyebrows, eyes, nose, mouth, and jawline, together with additional finer points along the contours of the face. Primarily, for detecting facial points, $(\lambda)$is signified in matrix form. Thereafter, the facial landmark points (F) are detected, and it is represented as equation 16,

$$F = \{F_1, F_2, F_3 \dots\dots, F_{68}\} \quad (16)$$

Where, the 68 landmark points are indicated as(F_{68}).

Band Transformation

Here, by utilizing BBT, band transformation is performed to cover all the features from(λ). The traditional Bandlet Transform is efficient in capturing overall facial features instead of concentrating on specific facial objects. But its performance is highly reliant on input parameters. Furthermore, errors can be increased by incorrect parameter selection. To address this issue, Bernoulli distribution-centric parameter selection is used. The Bernoulli distribution assists in probabilistically choosing optimal parameters for reducing errors. Primarily, by using the Bernoulli distribution(ϑ), the optimal parameters are selected, and it is depicted as equation 17,

$$\vartheta = \varphi^{y}(1-\varphi)^{1-y} \quad (17)$$

Here, the probability of selecting the best parameter from(λ) is exemplified as(φ), and the binary selection variable for each parameter in band transformation is implied as(y). In this, from(ϑ), the parameters, namely wavelet scale(λ), orientation threshold(λ'), bandlet basis function(λ''), and Bandlet coefficients$(\ddot{\lambda})$ are chosen. Then, for geometric analysis, (λ) is decomposed into wavelet coefficients at

multiple scales grounded on the selected parameters. Next, centered on(λ'), the dominant orientation of local features is recognized by utilizing the gradient of decomposed($\bar{\lambda}$) are shown in equations 18 & 19.

$$\nabla\bar{\lambda} = \left(\frac{\partial\bar{\lambda}}{\partial u}, \frac{\partial\bar{\lambda}}{\partial v}\right) * \lambda' \quad (18)$$

$$\lambda' = tan^{-1}\left(\frac{\frac{\partial\bar{\lambda}}{\partial u}}{\frac{\partial\bar{\lambda}}{\partial v}}\right) \quad (19)$$

Where, the gradient($\bar{\lambda}$) is notated as($\nabla\bar{\lambda}$), the inverse tangent function is indicated as(tan^{-1}), and the partial derivatives capturing horizontal(u) and vertical intensity(v) changes are implied as $\frac{\partial\bar{\lambda}}{\partial u}$ and $\frac{\partial\bar{\lambda}}{\partial v}$, correspondingly. Afterward, to align with the direction of the geometric flow, a bandlet basis(β) is constructed in equation 20.

$$\beta = \sum \nabla\bar{\lambda} * \lambda'' \quad (20)$$

Lastly, the bandlet coefficient($\ddot{\lambda}$) is computed. Here, (B'') signifies the obtained transformed band. Thereafter, for attaining the orientation and directional information, the directional frequency band(B) is chosen from(B'').

Beizure Curve Construction

In this section, the Beizure curves are constructed from($\bar{\lambda}$) to investigate the direction of facial lines formed by key points like the line between the eyes or from the nose to the mouth. The angle direction refers to how these facial points connect, thus signifying changes in facial expressions. Initially, from($\bar{\lambda}$), the control points(h_0, h_1, h_2) are selected. After that, a quadratic Beizure curve(Q) is articulated as equation 21,

$$Q = 2(t-1)(h_1 - h_0) + 2t(h_2 - h_1) \quad (21)$$

Here, the parameter that interpolates between(h_0, h_1, h_2) is denoted as(t) and(X) represents the constructed Beizure curves.

Angle Estimation

Then, the angles of the lines formed by facial features are evaluated from(X). To examine the direction and inclination of facial features, the estimated angles are utilized are shown in equation 22.

$$D = tan^{-1}\left(\frac{X_a}{X_{a'''}}\right) \quad (22)$$

Where, the estimated angle is notated as(D), and (a) and (a''')-components of the Beizure curve's derivative are exemplified as(X_a) and($X_{a'''}$), respectively.

Feature extraction

The features, namely Gray Level Co-occurrence Matrix (GLCM), LBP, shape, and geometry features are extracted from($\bar{\lambda}$),(Ψ), and (B). These extracted features(ξ) are represented as equation 23,

$$\xi = \langle \xi^1, \xi^2, \xi^3, \xi^4, \ldots\ldots \xi^g \rangle \quad (23)$$

Here, the total number of(ξ) is implied as(g).

Feature Selection

EWSBO is applied to the optimal features in ξ and the model complexity in terms of time is reduced. Secretary Bird Optimization (SBO) algorithm represents the model of birds that search prey and avoid predators, and a preliminary search in the environment. In order to prevent premature convergence, entropy-weighted function is used which optimizes the process.

Initialization: Grounded on(ξ), the position of every single secretary bird is initialized in the search space. The population matrix(K) of the solution is provided as equation 24 and 25,

$$K = \begin{bmatrix} k_{1,1} & \cdots & k_{1,q} & \cdots & k_{1,\hat{q}} \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ k_{b,1} & \cdots & k_{b,q} & \cdots & k_{b,\hat{q}} \\ \vdots & \cdots & \vdots & \cdots & \vdots \\ k_{M,1} & \cdots & k_{M,q} & \cdots & k_{M,\hat{q}} \end{bmatrix} \tag{24}$$

$$K_{b,q} = lb_q + \left(ub_q - lb_q\right) * \upsilon_{ran} \tag{25}$$

Where, the position of the$\left(b^{th}\right)$ secretary bird for the$\left(q^{th}\right)$ feature is notated as$\left(\mathrm{K}_{b,q}\right)$, the number of features is denoted as$(\hat{q})$, the number of secretary birds is implied as(M), the random number is exemplified as(υ_{ran}), and the lower and upper bounds of the $\left(q^{th}\right)$ feature are indicated as $\left(lb_q\right)$ and $\left(ub_q\right)$, correspondingly.

Fitness Function: The fitness function(Ω) is determined to assess the quality of every single candidate solution. Equation 26 represented as,

$$\Omega = max(\zeta^{acc}) \tag{26}$$

Here, the maximum accuracy is exemplified as$max(\zeta^{acc})$.

Exploration Phase: The hunting strategy of the secretary bird comprises three key stages.

✓ **Searching for Prey:** In this stage, the position of the secretary bird is updated, and it is calculated as equation 27 and 28,

$$K_{b,q}^{p1} = K_{b,q} + (\upsilon_{ran1} - \upsilon_{ran2}) * \upsilon_{ran}''' \tag{27}$$

$$K_{b,q} = \begin{cases} K_{b,q}^{p1}, \ if \ \Omega^{p1} < \Omega \\ K_{b,q}, \ else \end{cases} \tag{28}$$

Where, the updated position of the $\left(b^{th}\right)$ secretary bird for the$\left(q^{th}\right)$ feature in the first stage$(p1)$ is implied as $\left(K_{b,q}^{p1}\right)$, the random candidate solutions are notated as(υ_{ran1}) and (υ_{ran2}), the random array is exemplified as(υ_{ran}'''), and the updated (Ω) in$(p1)$ is signified as(Ω^{p1}).

✓ **Consuming Prey:** Here, by utilizing Brownian motion$\left(\breve{\beta}\right)$ and historical best position$(\tilde{m}_{best})$, the secretary bird updates its position to improve local search. Equation 29 and 30 represented as,

$$K_{b,q}^{p2} = \tilde{m}_{best} + e\left(\left(\frac{\vec{t}}{\bar{t}}\right) \wedge 4\right) * \left(\breve{\beta} - 0.5\right) * \left(\tilde{m}_{best} - K_{b,q}\right) \tag{29}$$

$$K_{b,q} = \begin{cases} K_{b,q}^{p2}, \ if \ \Omega^{p2} < \Omega \\ K_{b,q}, \ else \end{cases} \tag{30}$$

Here, the updated position of the (b^{th}) secretary bird for the(q^{th}) feature in the second stage$(p2)$ is indicated as$(K_{b,q}^{p2})$, the updated (Ω)in$(p2)$ is exemplified as(Ω^{p2}), the exponential function is implied as(e), and the current and total number of iterations are notated as $(\vec{t})$ and$(\check{t})$, correspondingly. This strategy improves the birds' local search efficiency.

Attacking Prey: The position is updated in the attacking prey stage by using an entropy-weighted function. This ensures a balanced exploration and mitigates the risk of premature convergence which are shown in equation 31 and 32.

$$\ddot{\kappa}_q = -\sum_q^{\sum q} \varphi_q log \tag{31}$$

$$W = \frac{\ddot{\kappa}_q}{\sum_{q=1}^{\bar{q}} \ddot{\kappa}_q} \tag{32}$$

Where, the entropy of the (q^{th}) feature is represented as$(\ddot{\kappa}_q)$, the probability distribution of the (q^{th})feature is indicated as(φ_q), and the entropy-based weight is implied as(W). The updated position is equated with equation 33 and 34,

$$K_{b,q}^{p3} = \tilde{m}_{best} * K_{b,q} * W * \vec{L} \tag{33}$$

$$K_{b,q} = \begin{cases} K_{b,q}^{p3}, & if\ \Omega^{p3} < \Omega \\ K_{b,q}, & else \end{cases} \tag{34}$$

Here, the updated position of (b^{th}) secretary bird for the(q^{th}) feature in the third stage$(p3)$is notated as$(K_{b,q}^{p3})$, the updated (Ω)in$(p3)$ is exemplified as(Ω^{p3}), and the Levy flight is denoted as$\vec{L}$.

Exploitation Phase: Secretary birds utilize two strategies, namely camouflage(J^1) and flight or rapid running(J^2), to evade predators. The position updation of secretary bird is depicted as equation 35 and 36,

$$K_{b,q}^{p4} = \begin{cases} J^1: \tilde{m}_{best} + (2 * \check{\beta} - 1) * \left(1 - \frac{\vec{t}}{\check{t}}\right)^2 * K_{b,q}, & if v_{ran} < v_{ran}^b \\ J^2: K_{b,q} + v_{ran2} * (\hat{v}_{ran} - \check{h} * K_{b,q}), & else \end{cases} \tag{35}$$

$$K_{b,q} = \begin{cases} K_{b,q}^{p4}, & if\ \Omega^{p4} < \Omega \\ K_{b,q}, & else \end{cases} \tag{36}$$

Where, the updated position of the (b^{th}) secretary bird for the(q^{th}) feature in the fourth stage$(p4)$ is implied as$(K_{b,q}^{p4})$, the updated (Ω) in$(p4)$ is signified as(Ω^{p4}), the random value for the (b^{th}) secretary bird is notated as(v_{ran}^b), the selection variable is exemplified as$(\check{h})$, the randomly selected candidate solution is indicated as$(\hat{v}_{ran})$, and $(\aleph)$ represents the chosen optimal features.

Emotion Classification

RSRN categorizes emotions from features $(\aleph)$, (D), (F), (λ), (Ψ), and (S). ResNet efficiently trains deep networks, capturing detailed facial features at multiple scales. To address challenges in feature complexity, a Scaled Dot-Product Attention Activation Function focuses on key features. Ridge Regression-based Feature Mapping reduces overfitting by preserving important features, ensuring robust learning. Figure 2 illustrates the RSRN framework.

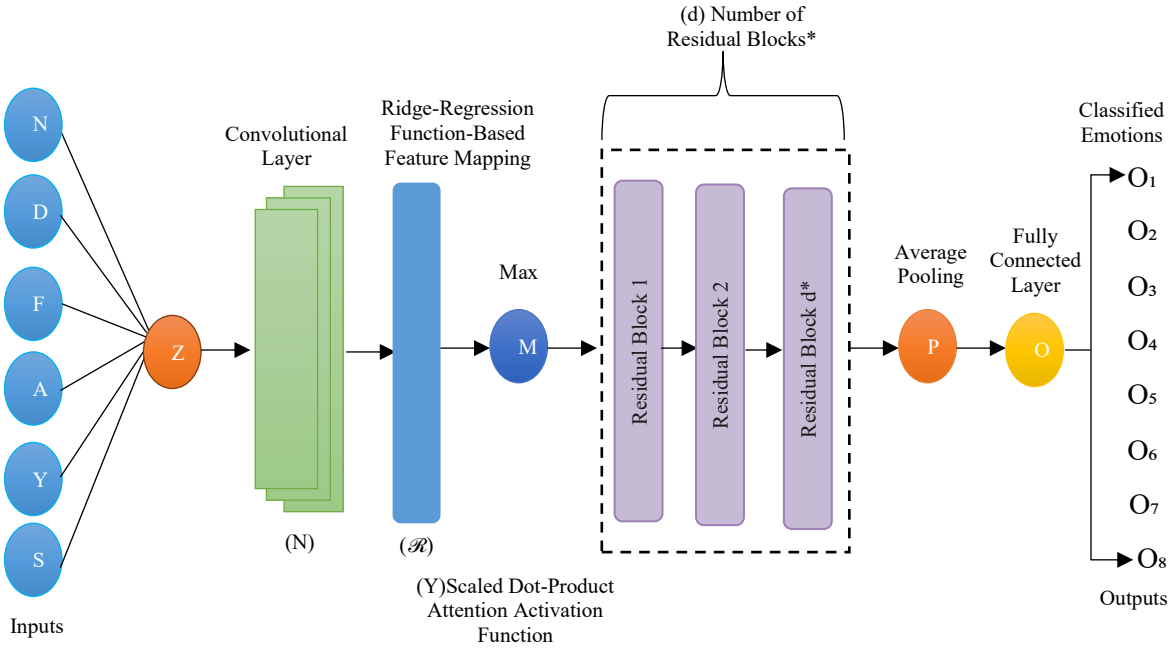


Figure 2: Diagrammatic representation of the proposed RSRN

Initially, the inputs like$(\aleph)$,(D),(F),(λ), (Ψ), and (S) are generally represented as(Z). This(Z) is given as input to the RSRN model.

Convolutional Layer: Primarily, the input(Z) is passed to the convolutional layer(N), which performs batch normalization$\hat{\wp}^{batch}$ and activation for stabilizing and enhancing learning which are represented in equation 37 and 38.

$$N = \left(\vec{\gamma}\left(\hat{\wp}^{batch}(x \bullet Z + z)\right)\right) \tag{37}$$

$$\wp^{batch} = \frac{Z - \vec{\mu}}{\sqrt{\ddot{\sigma}^2 + \tilde{\ell}}} \tag{38}$$

Here, the batch mean is implied as$(\vec{\mu})$, the batch standard deviation is notated as$(\ddot{\sigma})$, a small constant utilized for denominator stability is signified as$(\tilde{\ell})$, and the weight and bias of(N) are exemplified as(x) and(z), correspondingly. Moreover, $(\vec{\gamma})$ indicates the Scaled Dot-Product Attention Activation Function, which is provided as equation 39 and 40,

$$\vec{\gamma} = \ddot{X} * \left(\frac{\ddot{I}\ddot{K}}{\sqrt{\ddot{\tau}}}\right)\ddot{v} \tag{39}$$

$$\ddot{X} = \frac{e^{Z_i}}{\sum e^{Z_{i'}}} \tag{40}$$

Where, the SoftMax function is exemplified as $\ddot{X}$, the query, key, and values of (Z) are notated as $(\ddot{I})$, $(\ddot{K})$, and $(\ddot{\tau})$, respectively, the(i^{th}) and (i'^{th}) inputs are represented as (Z_i) and $(Z_{i'})$, correspondingly, and the scaling factor is denoted as $(\ddot{v})$.

Ridge Regression Function-based Feature Mapping: Ridge Regression function-based feature mapping$(\ddot{\Re})$ is applied for diminishing the impact of less important features without completely discarding them. Equation 41 demonstrated as,

$$\ddot{\Re} = min\left(\sum\left(N - \hat{N}\right)^2 + \hat{\alpha}\sum_{j=1}^{j''}\ddot{\chi}_j^2\right) \tag{41}$$

Here, the predicted input is implied as$(\hat{N})$, the ridge regularization parameter is indicated as$(\hat{\alpha})$, the feature mapping coefficient with index(j) is exemplified as$(\ddot{\chi}_j)$, and the total number of$(\ddot{\chi}_j)$ is notated as (j'').

Max Pooling: After mapping, the max pooling is applied for$(\ddot{\Re})$ to mitigate the spatial dimensions and retain the most significant features. Here, $(\vec{M})$ signifies the max pooled output.

Residual Blocks: In this, the RSRN has the (d')number of residual blocksω^{res}. The output of one block is given as input to the next block, thus preserving information that is shown in equation 42.

$$\omega^{res} = x' * \vec{\gamma}(x''\vec{M} + z') + z'' + \vec{M} \qquad (42)$$

Where, weights ofω^{res} are implied as(x') and(x''), and (z') and (z'') notate bias ofω^{res}.

Average Pooling: The mean activation per feature map is calculated by the average pooling$(\vec{P})$, thereby preserving key emotion features and diminishing complexity. Equation 43 represented as,

$$\vec{P} = \frac{\sum_{i=1}^{d'} \omega^{res}}{d'} \qquad (43)$$

Here, the number of$\omega^{res}$is exemplified as(i).

Fully Connected Layer: In this, for emotion classification, $(\vec{P})$ is fed into a fully connected layer(O) with softmax activation$\ddot{X}$ which are shown in equation 44 and 45.

$$O = \ddot{X}(\vec{P} * \vec{x}) + \vec{z} \qquad (44)$$

$$O = \{O^1, O^2, O^3, O^4, O^5, O^6, O^7, O^8\} \qquad (45)$$

Here, the bias and weight of(O) are notated as$(\vec{z})$ and $(\vec{x})$, respectively, angry is denoted as (O^1), contempt is implied as(O^2), disgust is indicated as(O^3), fear is signified as(O^4), happiness is exemplified as(O^5), neutral is represented as(O^6), sad is implied as (O^7), and surprise is exemplified as(O^8). Therefore, the output from (O) is regarded as the classified emotions.

Emotion Intensity Estimation

In this section, by considering facial landmark points(F), the intensity of emotions is evaluated for the obtained class(O) utilizing STFIS. Multi-criteria decision-making is supported by the traditional Fuzzy Inference System (FIS). But, the membership function of FIS lacks a smooth transition owing to negative points during membership construction. To address this problem, the Sig-Tangent membership function that ensures a gradual transition by overcoming negative point effects is utilized. Here, centered on landmark points, the Action Unit (AU) is determined. By summing the absolute differences of all landmark-based AU, the intensity of an emotion is evaluated. The intensity of AU(V) obtained from(F) calculated by equation 46,

$$V = |s^{\Theta^{land}} - s^{\Theta^{land}}| \qquad (46)$$

Where, the landmark points of (s^{th}) and (s'''^{th}) facial objects are implied as ${}^{s''}\Theta^{land}$ and ${}^{s''}\Theta^{land}$, correspondingly. Similarly, the intensity AU is calculated for all the classified emotions.

Fuzzification: After that, by utilizing the Sig-Tangent membership function$(\breve{\Xi}'')$, the AU intensity is converted into fuzzy values. The Sig-Tangent membership function overcomes the negative point effects. Moreover, the fuzzified data$(\vec{U}^{fuz})$ is equated as equation 47 and 48,

$$\vec{U}^{fuz} = \breve{\Xi}'' * V \quad (47)$$

$$\breve{\Xi}'' = \frac{1}{1+e^{-\hat{k}(V-\hat{g})}} * \frac{1}{1+\left(\frac{|V-\hat{g}|}{\hat{a}}\right)} \quad (48)$$

Here, the steepness of the sigmoid part is exemplified as$(\hat{k})$, the centre of the function is notated as$(\hat{g})$, and the width of the membership function is signified as$(\hat{a})$.

Fuzzy Rule Base: Grounded on the intensity level$(\breve{\psi})$ obtained from$(\vec{U}^{fuz})$, the fuzzy rules$(\ddot{\Phi}_{rules})$ are defined, and it is provided as equation 49,

$$\ddot{\Phi}_{rules} = \begin{cases} If\ \breve{\psi}\ \ is\ \ between\ \ (0-33)\ ,\ \ then\ \ \tilde{\Theta}^{low} \\ If\ \breve{\psi}\ \ is\ \ between\ \ (34-66)\ ,\ \ then\ \ \tilde{\Theta}^{mod} \\ If\ \breve{\psi}\ \ is\ \ between\ \ (67-100)\ ,\ \ then\ \ \tilde{\Theta}^{high} \end{cases} \quad (49)$$

Here, low emotion is implied as$\tilde{\Theta}^{low}$, moderate emotion is indicated as $\tilde{\Theta}^{mod}$, and high emotion is exemplified as $\tilde{\Theta}^{high}$.

Inference Engine: To infer emotion intensity, the STFIS processes fuzzy inputs$(\vec{U}^{fuz})$ utilizing$(\ddot{\Phi}_{rules})$ which are shown in equation 50,

$$inf^{\delta} = U^{fic} * \Phi_{rules} \quad (50)$$

Defuzzification: The final crisp intensity score is attained. The defuzzified data$\vec{U}^{def}$ is expressed as equation 51,

$$\vec{U}^{def} = \frac{\sum_{inf} \delta\ X\ \Xi(inf^{\delta})}{\sum \Xi(inf^{\delta})} \quad (51)$$

Hence, the intensity of emotions is evaluated from equation (51). Furthermore, centered on the obtained emotion classification, the final decision is made. Here, $(\breve{\xi})$ represents the estimated intensities.

Algorithm 2 for STFIS

Input: Classified Emotion(O) and Detected Landmark Points(F)

Output: Estimated Intensity $(\breve{\xi})$

Begin

Initialize$(\hat{k})$, $(\hat{g})$, $(\hat{a})$

For each (O) and(F)

Calculate (V)

$$V = \left|{}_{s}\Theta^{land} - {}_{s'''}\Theta^{land}\right|$$

Fuzzify (V) using $(\breve{\Xi}'')$

Compute $(\breve{\Xi}'')$

$$\breve{\Xi}'' = \frac{1}{1+e^{-\hat{k}(V-\hat{g})}} * \frac{1}{1+\left(\frac{|V-\hat{g}|}{\hat{a}}\right)}$$

Generate rules$(\ddot{\Phi}_{rules})$

If $\left(\breve{\psi} \xrightarrow{is\ between} 0-33\right)$

Return $\tilde{\Theta}^{low}$

Elif $\left(\breve{\psi} \xrightarrow{is\ between} 34-66\right)$

Return $\tilde{\Theta}^{mod}$

Elif$\left(\breve{\psi} \xrightarrow{is\ between} 67-100\right)$

Return $\tilde{\Theta}^{high}$

End if

Evaluate $(inf())$

Perform defuzzification and obtain$\vec{U}^{def}$

End for

Return Estimated Intensity $\left(\breve{\xi}\right)$

End

Algorithm 2 estimates emotion intensity based on classified emotion (O) and detected landmark points (F). It calculates the difference (V) between the predicted and actual landmark positions, then fuzzifies the value using a fuzzy function Ξ''. The system generates rules to classify the intensity into low, moderate, or high ranges. After evaluating the intensity, the algorithm performs defuzzification to obtain the final estimated intensity ξ'.

Once an emotion is identified (e.g., Anger), its intensity is evaluated in this phase for further determining whether it falls into low, moderate, or high intensity levels. The performance assessment of the proposed framework is explained in further sections.

4 Results and Discussions

Here, the performance of the proposed framework is assessed against prevailing methods. The framework was coded with Python 3.8, where such libraries are used as OpenCV to detect faces, Dlib to detect facial points, TensorFlow/Keras to train RSRN, and Scikit-learn to select features. Matplotlib/Seaborn was used to visualize and CUDA accelerated the training faster on the AffectNet data. To do the experiments, all parameters were properly initialized in order to provide reproducibility and performance. The RSRN had a learning rate of η = 0.001, 3 residual blocks, and EWSBO had 30 secretary birds that were aged to 50 iterations. PG-CLAHE, Dlib-68, and STFIS parameters were adjusted according to standard parameters of correct FER estimation and intensity estimation.

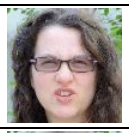
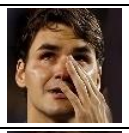
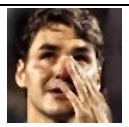
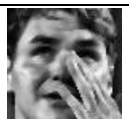
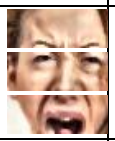
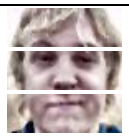
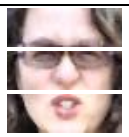
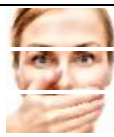
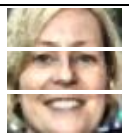
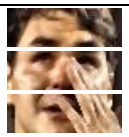
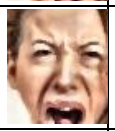
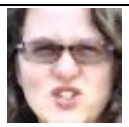
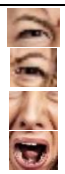
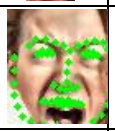
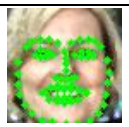
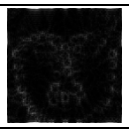
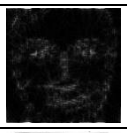
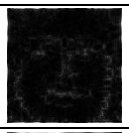
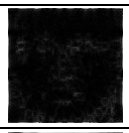
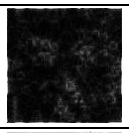
Dataset Description

Table 1: Number of images used during training and testing

Number of Images/ Emotions	Training	Testing
Angry	2394	598
Contempt	2,136	533
Disgust	1,843	460
Fear	2,363	590
Happy	3,752	937
Neutral	3,814	952
Sad	2,300	574
Surprise	3,004	751
Total	21,606	5,395

The AffectNet dataset is utilized in the proposed framework. Moreover, the dataset link is given in the reference section. A large-scale facial expression dataset labeled with 8 emotions is termed AffectNet, which is generally used for training facial expression recognition models. In table 1, the number of images utilized for training (80%: 21,606) and testing (20%: 5,395) every single emotion is depicted.

Table 2: Sample image outcomes of the proposed work

Phases/ Emotions	Angry	Contempt	Disgust	Fear	Happy	Neutral	Sad	Surprise
Input Face Image								
Noise Removed Image								
Contrast Enhanced Image								
Gray scaled Image								
Patched Images								
Face Detected Image								
Segmented Facial Objects								
Face Point Image								
Band Transformed Image								
Beizure curve Image								

The phases involved in FER across different emotions, comprising Angry, Contempt, Disgust, Fear, Happy, Neutral, Sad, and Surprise, are presented in table 2. Input Face Image, Noise Removed Image, Contrast Enhanced Image, Grayscaled Image, Patched Image, Face Detected Image, Segmented Facial

Objects, Face Point Image, Band Transformed Image, and Beizure Curve Image are encompassed in the phases.

Performance Analysis

Performance Analysis of Emotion Intensity Estimation

In figure 3, the performance of the proposed STFIS and existing techniques, namely FIS, Gaussian FIS, Sigmoid FIS, and Trapezoidal FIS regarding fuzzification, defuzzification, and rule generation times is displayed. The proposed STFIS fuzzified, defuzzified and generated rules in times of 958.697ms, 923.013ms, and 848.998ms, respectively, and thus proved to be efficient.

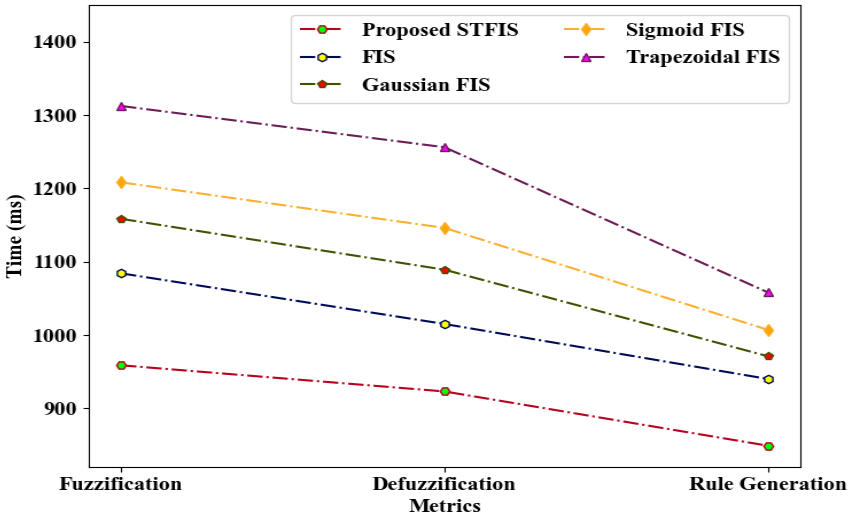


Figure 3: Analysis of the proposed STFIS and existing techniques

Performance Analysis of Emotion Classification

The proposed RSRN (Figure 4) was more precise and accurate (98.97%), recall (99.03%), and F-Measure (99.03%) and highest MCC (97.92) compared to existing techniques (ResNet, CNN, LSTM, DNN).

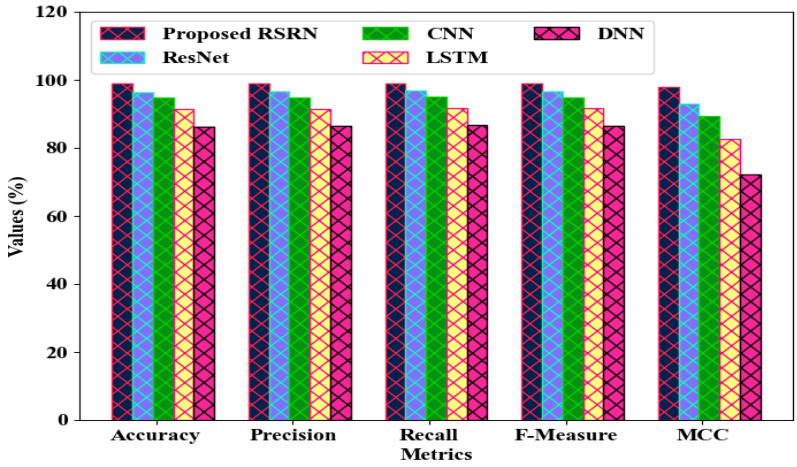


Figure 4: Performance analysis of the proposed RSRN and existing techniques

Conversely, ResNet, CNN, LSTM and DNN had lower accuracies of 96.47, 94.75, 91.3, and 86.13 respectively. The high performance of RSRN was attributed to Scaled Dot-Product Attention Activation Function and Ridge Regression-based Feature Mapping, something that increased the focus on features and minimal overfitting.

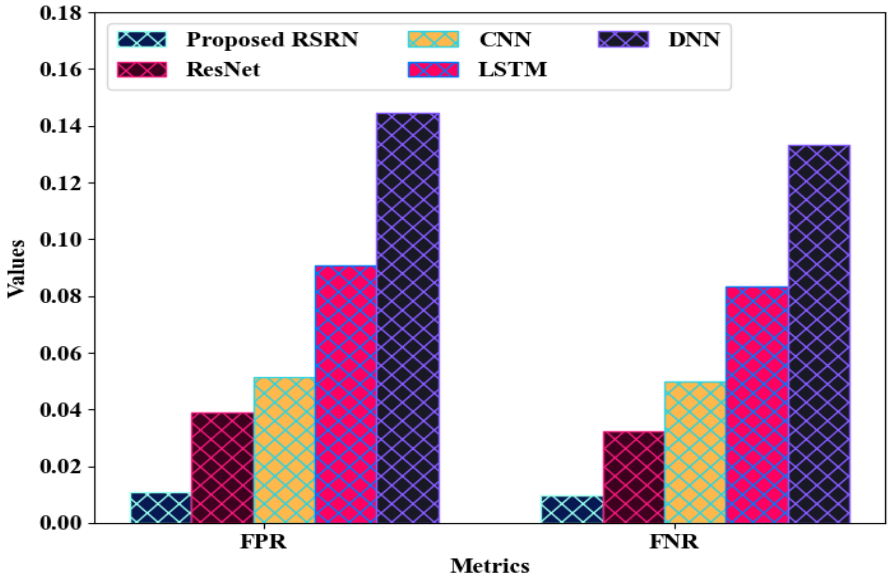


Figure 5: FPR and FNR analysis

Table 3: Analysis of training time

Techniques	Training Time (ms)
Proposed RSRN	14453467
ResNet	15533466
CNN	17693466
LSTM	19853466
DNN	22373466

In figure 5 and table 3, the performance of the proposed RSRN and existing techniques concerning False Positive Rate (FPR), False Negative Rate (FNR), and training time is displayed. The proposed RSRN attained low FPR (0.0111), FNR (0.0097), and training time (14453467ms) by utilizing the Scaled Dot-Product Attention Activation Function and Ridge Regression-based Feature Mapping. Contrarily, the prevailing techniques, namely ResNet, CNN, LSTM, and DNN had higher FPR as well as FNR and took longer training times. Hence, when contrasted with these existing techniques, the proposed RSRN efficiently reduced training time and minimized errors.

Performance Analysis of Face Detection and Facial Object Segmentation

In this section, the performance of the proposed LF-VJ and prevailing techniques is examined and analogized regarding mean Average Precision (mAP).

Table 4 depicted that the highest mAP of 98.245% was obtained by proposed LF-VJ, thereby outperforming Viola Jones (95.736%), Dlib (93.681%), DNN (91.068%), and SVM-HOG (88.438%). This superior performance was owing to the enhanced feature extraction and classification capabilities

of LF-VJ. Accurate face detection and facial object segmentation were ensured by LF-VJ's logarithmic function-centric weight update.

Table 4: mAP analysis

Techniques	mAP (%)
Proposed LF-VJ	98.245
Viola Jones	95.736
Dlib	93.681
DNN	91.068
SVM-HOG	88.438

Performance Analysis of Contrast Enhancement

Here, concerning Mean Squared Error, Peak Signal-to-Noise Ratio (PSNR), and Structural Similarity Index Measure (SSIM), the performance of the proposed PG-CLAHE and prevailing techniques like CLAHE, Dynamic Pseudo Histogram Equalization (DPHE), Histogram Equalization (HE), and Adaptive Histogram Equalization (AHE) is investigated.

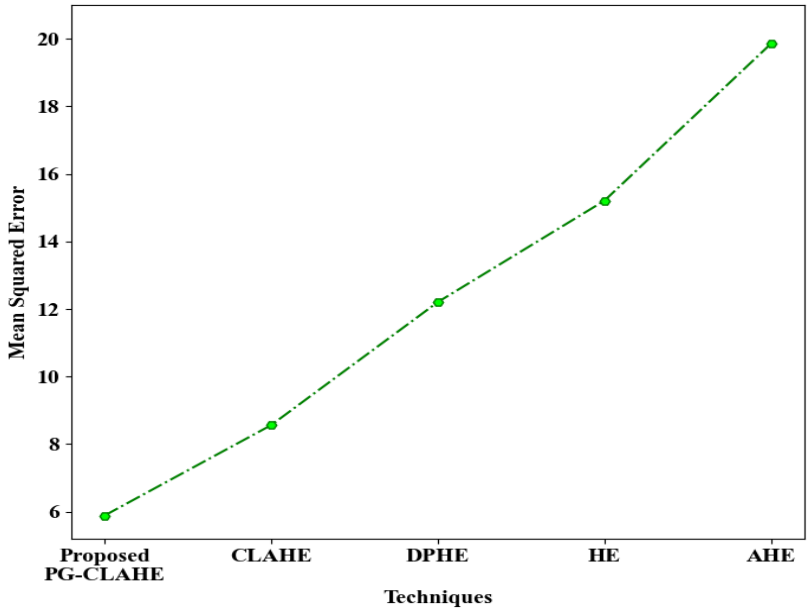


Figure 6: Analysis of mean squared error

The Mean Squared Error analysis of the proposed PG-CLAHE and existing techniques is depicted in figure 6. The lowest Mean Squared Error (5.8879) was obtained by the proposed PG-CLAHE, thus outperforming CLAHE (8.5643), DPHE (12.206), HE (15.2074), and AHE (19.8563).

Table 5: PSNR and SSIM analysis

Techniques	PSNR (dB)	SSIM
Proposed PG-CLAHE	28.4145	0.9095
CLAHE	20.4145	0.8295
DPHE	17.304	0.7383
HE	14.2427	0.6707
AHE	9.998	0.4427

The lower Mean Squared Error of the proposed PG-CLAHE was due to the utilization of a polar gradient function to adjust the clip limit and tile size, thereby resulting in more accurate image enhancement. Thus, image contrast was effectively enhanced by the proposed methodology. The highest PSNR (28.4145dB) and SSIM (0.9095) were obtained in the proposed PG-CLAHE, which means high-quality images and structural maintenance shown in table 5. By comparison, CLAHE, DPHE, HE, and AHE had smaller PSNR and SSIM values, and, effectively, the image structure and contrast enhancement were achieved with the help of PG-CLAHE.

Performance Analysis of Feature Selection

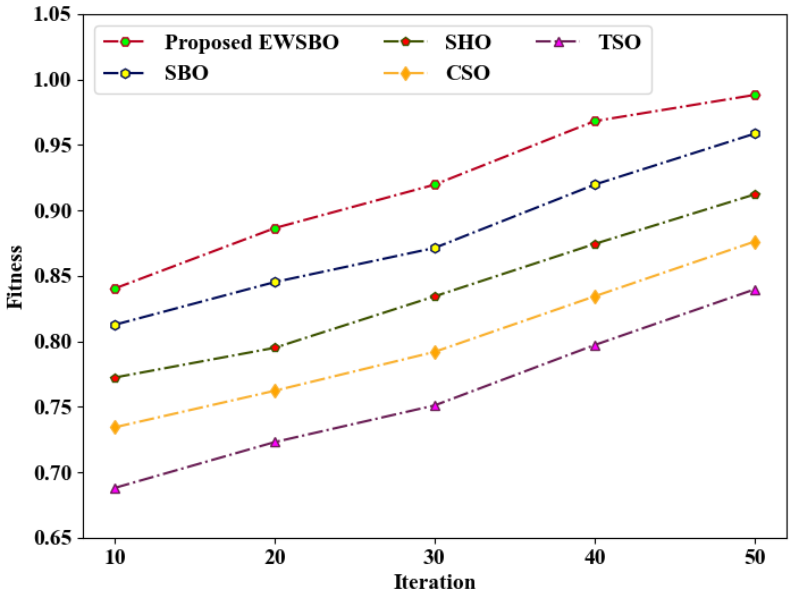


Figure 7: Fitness vs. iteration

Figure 7 compares the performance of the proposed EWSBO and the current techniques including SBO, SHO, CSO, and TSO. When compared to the existing techniques, the proposed EWSBO consistently obtained higher fitness values at all iterations. Specifically, at iteration 50, EWSBO achieved a fitness value of 0.9883, thereby outperforming SBO (0.9587), SHO (0.9123), CSO (0.8763), and TSO (0.8398). This enhancement was due to the use of an entropy-weighted function that improved the optimization process, thus causing superior convergence behavior. Therefore, the best optimal features were chosen more effectively.

Metrics and Evaluation

Accuracy: Equation 52 measures the overall correctness of the model by calculating the ratio of correctly classified samples (both positive and negative) to the total number of samples.

$$\text{Accuracy} = \frac{\text{True Positives+True Negatives}}{\text{Total Samples}} \qquad (52)$$

Precision: Equation 53 assesses how many of the predicted positive samples were actually positive, providing insight into the model's ability to avoid false positives.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives+False Positives}} \qquad (53)$$

Structural Similarity Index (SSIM): Equation 54 compares the structural similarity between two images, accounting for luminance, contrast, and structure to evaluate image quality. Higher SSIM values indicate more similar images.

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y+C_1)(2\sigma_{xy}+C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)} \quad (54)$$

Comparative Analysis

To assess performance differences, the comparative analysis of the proposed framework is investigated with existing works.

Table 6: Comparative analysis of proposed with existing works

Authors' Name	Techniques Used	Accuracy (%)
Proposed	RSRN	98.97
Bisogni et al., 2022	Visual Transformers with Feature Fusion (VTFF)	61.85
Wang et al., 2023	GCN	60.13
Kuruvayil & Palaniswamy, 2022	Few-shot learning with meta-learning	68
Gao et al., 2023	Spectral and Spatial Attention- Intra-dataset Continual Learning (SSA-ICL)	65.78
Liao et al., 2022	GCN	62.03

Table 6 shows the comparative analysis of the proposed RSRN with existing works. The highest accuracy of 98.97% was obtained by the proposed RSRN, thereby surpassing prevailing techniques. Here, the existing VTFF attained an accuracy of 61.85%. However, it suffered from high computational costs. In prevailing Wang et al., 2023, the GCN achieved a 60.13% accuracy. But it lacked accurate AU detection, thus limiting its potential to handle real-world uncertainties. Then, the Few-shot learning with meta-learning attained an accuracy of 68%; however, it failed to categorize emotions grounded on intensity. Likewise, an accuracy of 65.78% was achieved by SSA-ICL, thus facing challenges in interpreting attention weights. Moreover, GCN utilized in prevailing Liao et al., 2022 obtained an accuracy of 62.03%. Nevertheless, it depicted poor scalability for large graphs. The proposed RSRN addressed these drawbacks by considering facial wrinkles, thereby accurately identifying emotions and efficiently evaluating intensities.

5 Conclusion

In this paper, an enhanced FER and intensity estimation framework utilizing RSRN and STFIS was presented, thus integrating facial wrinkles to enhance accuracy. Initially, the input images were pre-processed. Moreover, by using PG-CLAHE, the contrast was enhanced with a 28.4145dB PSNR. The images were transformed into grayscale and further patched horizontally. Thereafter, wrinkle scores and cosine similarity were computed. Afterward, by utilizing LF-VJ, faces were detected from the pre-processed images and facial objects were segmented, thus obtaining a mAP of 98.245%. From the detected face, facial points were detected, followed by band transformation and Bezier curve construction. After that, the angle was evaluated. Then, features were extracted. Also, by using EWSBO, optimal features were chosen with a 0.9883 fitness value at iteration 50. From the calculated cosine similarity, selected features, estimated angle, detected facial points, detected faces, and segmented facial

objects, the classification was done utilizing RSRN, thereby attaining an accuracy of 98.97% and a precision of 99.03%. Lastly, by using STFIS, emotion intensity was evaluated with a rule generation time of 848.998ms. Therefore, the proposed framework efficiently recognized facial emotions and evaluated their intensity.

Future Enhancement

Even though the facial emotions were efficiently identified and assessed, the proposed framework failed to recognize sub-emotions, thus causing inaccurate emotion recognition. To enhance the accuracy of the system, sub-emotions will be identified in the future.

Dataset Link: https://www.kaggle.com/datasets/noamsegal/affectnet-training-data

References

[1] Bhagat, D., Vakil, A., Gupta, R. K., & Kumar, A. (2024). Facial emotion recognition (FER) using convolutional neural network (CNN). *Procedia Computer Science*, *235*, 2079-2089. https://doi.org/10.1016/j.procs.2024.04.197

[2] Bisogni, C., Castiglione, A., Hossain, S., Narducci, F., & Umer, S. (2022). Impact of deep learning approaches on facial expression recognition in healthcare industries. *IEEE Transactions on Industrial Informatics*, *18*(8), 5619-5627. https://doi.org/10.1109/TII.2022.3141400

[3] Chowdary, M. K., Nguyen, T. N., & Hemanth, D. J. (2023). Deep learning-based facial emotion recognition for human–computer interaction applications. *Neural Computing and Applications*, *35*(32), 23311-23328. https://doi.org/10.1007/s00521-021-06012-8

[4] Debnath, T., Reza, M. M., Rahman, A., Beheshti, A., Band, S. S., & Alinejad-Rokny, H. (2022). Four-layer ConvNet to facial emotion recognition with minimal epochs and the significance of data diversity. *Scientific Reports*, *12*(1), 6991. https://doi.org/10.1038/s41598-022-11173-0

[5] Do, L. N., Yang, H. J., Nguyen, H. D., Kim, S. H., Lee, G. S., & Na, I. S. (2021). Deep neural network-based fusion model for emotion recognition using visual data. *The Journal of Supercomputing*, *77*(10), 10773-10790. https://doi.org/10.1007/s11227-021-03690-y

[6] Gao, H., Wu, M., Chen, Z., Li, Y., Wang, X., An, S., ... & Liu, C. (2023). SSA-ICL: Multi-domain adaptive attention with intra-dataset continual learning for Facial expression recognition. *Neural Networks*, *158*, 228-238. https://doi.org/10.1016/j.neunet.2022.11.025

[7] Gautam, C., & Seeja, K. R. (2023). Facial emotion recognition using Handcrafted features and CNN. *Procedia Computer Science, 218,* 1295-1303. https://doi.org/10.1016/j.procs.2023.01.108

[8] Gupta, S., Kumar, P., & Tekchandani, R. K. (2023). Facial emotion recognition based real-time learner engagement detection system in online learning context using deep learning models. *Multimedia Tools and Applications*, *82*(8), 11365-11394. https://doi.org/10.1007/s11042-022-13558-9

[9] Kuruvayil, S., & Palaniswamy, S. (2022). Emotion recognition from facial images with simultaneous occlusion, pose and illumination variations using meta-learning. *Journal of King Saud University-Computer and Information Sciences*, *34*(9), 7271-7282. https://doi.org/10.1016/j.jksuci.2021.06.012

[10] Lakshmi, D., & Ponnusamy, R. (2021). Facial emotion recognition using modified HOG and LBP features with deep stacked autoencoders. *Microprocessors and Microsystems*, *82*, 103834. https://doi.org/10.1016/j.micpro.2021.103834

[11] Li, B., & Lima, D. (2021). Facial expression recognition via ResNet-50. *International Journal of Cognitive Computing in Engineering*, *2*, 57-64. https://doi.org/10.1016/j.ijcce.2021.02.002

[12] Li, S., & Deng, W. (2020). Deep facial expression recognition: A survey. *IEEE transactions on affective computing*, *13*(3), 1195-1215. https://doi.org/10.1109/TAFFC.2020.2981446

[13] Liao, L., Zhu, Y., Zheng, B., Jiang, X., & Lin, J. (2022). FERGCN: facial expression recognition based on graph convolution network. *Machine Vision and Applications*, *33*(3), 40. https://doi.org/10.1007/s00138-022-01288-9

[14] Liu, C., Hirota, K., Ma, J., Jia, Z., & Dai, Y. (2021). Facial expression recognition using hybrid features of pixel and geometry. *Ieee Access*, *9*, 18876-18889. https://doi.org/10.1109/ACCESS.2021.3054332

[15] Ma, F., Sun, B., & Li, S. (2023). Transformer-augmented network with online label correction for facial expression recognition. *IEEE Transactions on Affective Computing*, *15*(2), 593-605. https://doi.org/10.1109/TAFFC.2023.3285231

[16] Mohan, K., Seal, A., Krejcar, O., & Yazidi, A. (2020). Facial expression recognition using local gravitational force descriptor-based deep convolution neural networks. *IEEE Transactions on Instrumentation and Measurement*, *70*, 1-12. https://doi.org/10.1109/TIM.2020.3031835

[17] Muhammad, G., & Hossain, M. S. (2021). Emotion recognition for cognitive edge computing using deep learning. *IEEE Internet of Things Journal*, *8*(23), 16894-16901. https://doi.org/10.1109/JIOT.2021.3058587

[18] Naga, P., Marri, S. D., & Borreo, R. (2023). Facial emotion recognition methods, datasets and technologies: A literature survey. *Materials Today: Proceedings*, *80*, 2824-2828. https://doi.org/10.1016/j.matpr.2021.07.046

[19] Revina, I. M., & Emmanuel, W. S. (2021). A survey on human face expression recognition techniques. *Journal of King Saud University-Computer and Information Sciences*, *33*(6), 619-628. https://doi.org/10.1016/j.jksuci.2018.09.002

[20] Salama, E. S., El-Khoribi, R. A., Shoman, M. E., & Shalaby, M. A. W. (2021). A 3D-convolutional neural network framework with ensemble learning techniques for multi-modal emotion recognition. *Egyptian Informatics Journal*, *22*(2), 167-176. https://doi.org/10.1016/j.eij.2020.07.005

[21] Sati, V., Sánchez, S. M., Shoeibi, N., Arora, A., & Corchado, J. M. (2020, June). Face detection and recognition, face emotion recognition through NVIDIA Jetson Nano. In *International Symposium on Ambient Intelligence* (pp. 177-185). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-58356-9_18

[22] Saurav, S., Saini, R., & Singh, S. (2021). Facial expression recognition using dynamic local ternary patterns with kernel extreme learning machine classifier. *IEEE Access*, *9*, 120844-120868. https://doi.org/10.1109/ACCESS.2021.3108029

[23] Saxena, S., Tripathi, S., & Sudarshan, T. S. B. (2022). An intelligent facial expression recognition system with emotion intensity classification. *Cognitive Systems Research*, *74*, 39-52. https://doi.org/10.1016/j.cogsys.2022.04.001

[24] Shahabinejad, M., Wang, Y., Yu, Y., Tang, J., & Li, J. (2021, December). Toward personalized emotion recognition: A face recognition-based attention method for facial emotion recognition. In *2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021)* (pp. 1-5). IEEE.https://doi.org/10.1109/FG52635.2021.9666982

[25] Sheng, W., & Li, X. (2021). Multi-task learning for gait-based identity recognition and emotion recognition using attention enhanced temporal graph convolutional network. *Pattern Recognition*, *114*, 107868. https://doi.org/10.1016/j.patcog.2021.107868

[26] Sikkandar, H., & Thiyagarajan, R. (2021). Deep learning based facial expression recognition using improved Cat Swarm Optimization. *Journal of Ambient Intelligence and Humanized Computing*, *12*(2), 3037-3053. https://doi.org/10.1007/s12652-020-02463-4

[27] Tan, Y., Sun, Z., Duan, F., Solé-Casals, J., & Caiafa, C. F. (2021). A multimodal emotion recognition method based on facial expressions and electroencephalography. *Biomedical Signal Processing and Control*, *70*, 103029. https://doi.org/10.1016/j.bspc.2021.103029

[28] Veltmeijer, E. A., Gerritsen, C., & Hindriks, K. V. (2021). Automatic emotion recognition for groups: a review. *IEEE Transactions on Affective Computing*, *14*(1), 89-107. https://doi.org/10.1109/TAFFC.2021.3065726

[29] Wang, S., Zhao, A., Lai, C., Zhang, Q., Li, D., Gao, Y., ... & Wang, X. (2023). Gcanet: Geometry cues-aware facial expression recognition based on graph convolutional networks. *Journal of King Saud University-Computer and Information Sciences*, *35*(7), 101605. https://doi.org/10.1016/j.jksuci.2023.101605

[30] Xu, J., Li, Y., Yang, G., He, L., & Luo, K. (2024). Multiscale facial expression recognition based on dynamic global and static local attention. *IEEE Transactions on Affective Computing*. https://doi.org/10.1109/TAFFC.2024.3458464

Authors Biography

R. Virupaksha Gouda is currently working as Assistant Professor in the Department of Computer Science and Engineering at Ballari Institute of Technology & Management, Ballari, Karnataka. He obtained his Master's degree & BE from Ballari Institute of Technology & Management, Ballari, Karnataka, in the year 2013 & 2007 respectively. He has over 15 years of academic experience and is currently pursuing his Ph.D. at Visvesvaraya Technological University (VTU), Karnataka. He has published 2 research papers, holds 3 Indian patents. His research interests include Artificial Intelligence, Machine Learning.

Dr. Yeresime Suresh is currently working as Professor & HOD in Dept. of Artificial Intelligence at Ballari Institute of Technology & Management (BITM), Ballari. He obtained his Ph.D., from National Institute of Technology Rourkela in 2015. He obtained his Master's degree from Sri Jayachamarajendra College of Engineering (SJCE) Mysore & BE from Rao Bahadhur Y Mahabaleshwar Appa Engineering College (RYMEC) Ballari in the year 2010 & 2008 respectively. He has a teaching experience of 11 years, research experience of 4 years & one year industry experience. He has 50 publications in reputed International Journals & International Conferences till date. He is also the member of IEEE, ISTE and IAENG. He has also been reviewer for various International Conferences & referred IEEE and Inder science Journals. Three Scholars have been awarded PhD Degree & Six research scholars are currently pursuing PhD under his guidance from VTU Belagavi. His research interests are in the area of Artificial Intelligence & Machine Learning and Software Engineering.