SSRG International Journal of Electrical and Electronics Engineering
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I7P129

Volume 11 Issue 7, 317-331, July 2024

Original Article

Video Analysis Based Improved Multi-Facial Emotion Recognition and Classification Framework Using GCRCNN

Jyoti S. Bedre[1*], P. Lakshmi Prasanna[2]

[1,2]*Computer Science and Engineering, KL University, Andhra Pradesh, India.*

[1]*Corresponding Author : jyoti.phd2020@gmail.com*

Received: 14 May 2024 Revised: 17 June 2024 Accepted: 13 July 2024 Published: 26 July 2024

Abstract *- Facial emotions are the varying expressions of a person's face that communicate one's feelings and moods. Facial emotion in videos can be detected using techniques that analyze keyframes for facial muscle movements and patterns. However, these detections can be challenging due to potential simultaneous expressions and camera angle complexities. To overcome these pitfalls, this paper provides a practical framework for detecting facial emotions in videos. Firstly, the input key frames are pre-processed by MF and IN algorithms to acquire an enhanced image. Secondly, human detection and tracking occur using YOLOV7 and BYTE tracking algorithms. Then, the T-SNEVJ algorithm is used for face detection. Thirdly, facial landmark extraction using the HC technique, mesh generation, and feature extraction are done. Here, ED-SVR is utilized for mesh generation. In the meantime, feature point tracking followed by motion analysis is done using CC_OF. Finally, the GCRCNN algorithm classifies multi-facial emotions. The proposed system achieves a better accuracy and recall of 99.34% and 99.20%. Thus, the proposed methodology outperforms the existing FER techniques.*

Keywords *- Graph Convolution and Regular 1-D convolutional based Convolutional Neural Network (GCRCNN), You Only Look Once Version7 (YOLO V7), Intensity Normalization (IN), Median Filter (MF), Facial expressions, Human detection, Facial Emotion Recognition (FER).*

1. Introduction

Facial emotions refer to the various expressions and movements on a person's face that convey the person's emotional state, including happiness, sadness, anger, and more [1]. Depending on psychological, sociological, and cultural factors [2], the functions of emotions can be categorized into intrapersonal, interpersonal, and cultural dimensions [3].

These expressions play an important role in nonverbal communication. In recent years, computer vision has witnessed significant advancements in the development of techniques for facial emotion recognition in images and videos [4]. Understanding and interpreting facial expressions in Real-Time (RT) video sequences also plays a vital role in emotion-aware human-computer interfaces, intelligent surveillance systems, and psychological research [5]. Face detection and recognition technology offers enhanced security measures, streamlines transactions, and enables personalized healthcare services [6].

While the analysis of single facial expressions has been extensively done and studied in the existing works, the recognition and classification of multiple facial emotions in videos [7] pose unique and complex challenges. Most of the traditional works outperformed in video analysis-based FER. The process of recognizing facial emotions typically involves several crucial steps. First, the system captures an image or video of a human face using a camera or a similar device. Then, it pre-processes the captured data, which involves tasks such as normalization, alignment, and noise reduction to enhance the quality of the input. Subsequently, the technology employs various algorithms and models to detect facial landmarks, such as the eyes, eyebrows, nose, and mouth, which are pivotal in understanding and interpreting different emotional expressions [8].

Here, Deep Learning (DL) and Machine Learning (ML) techniques are used for the efficient detection and classification of human facial emotions by automatically identifying the individual's moods or behaviors [9, 10]. Artificial Intelligence (AI) based FER [11] also uses these ML and DL techniques for the efficient detection of human facial emotions [12].

One of the existing methods uses Convolutional Neural Networks (CNN), which classify facial emotions like

frustration, happiness, furiousness, and more [13]. Also, some traditional methods use Support Vector Machine (SVM), Recurrent Neural Network (RNN) [14], and Long Short-Term Memory (LSTM) for efficient facial-based emotion recognition in videos [15]. However, FER may face challenges in accurately predicting human expressions and capturing nuances of complex emotions. Additionally, difficulties in recognizing multiple emotions simultaneously can further impede accurate recognition.

Existing FER methods predominantly focus on identifying a single emotion per video frame, neglecting the presence of multiple individuals and the diversity of their simultaneous emotional expressions. This gap leads to inaccuracies in recognizing emotions in more dynamic and realistic settings. Moreover, traditional approaches often overlook the continuous movement of faces in videos, which is crucial for accurate landmark extraction and motion analysis.

Reinforcement learning-based methods, while innovative, have shown limited success in extracting domain knowledge necessary for distinguishing different emotions accurately. Additionally, existing methods tend to ignore the importance of analyzing motion-based feature sequences, further impacting their effectiveness in recognizing complex emotions.

To address these challenges, this research introduces a novel video analysis-based FER framework leveraging multiple advanced algorithms. The proposed framework employs MF and IN algorithms for pre-processing, YOLOV7 for human detection, BYTE tracking for human tracking, and T-SNEVJ for face detection. Furthermore, Sparse Voronoi Refinement (SVR) with Euclidean Distance (ED) and GCRCNN for emotion classification are utilized to enhance the accuracy and reliability of FER in videos.

1) To classify various emotions in a video with multiple individuals, the YOLOV7 technique is introduced. This enables the detection and counting of humans in the video.
2) Multi-object-based BYTE tracking tracks the total number of persons in the video frame.
3) The T-SNEVJ algorithm and Haar Cascade techniques are introduced for human face detection and feature landmark extraction.
4) SVR is created for mesh generation, producing sensitive and distinct angular features.
5) The GCRCNN method is introduced to recognize and classify emotions effectively.

Our proposed framework for multi-facial emotion recognition and classification in videos showcases significant novelty by integrating advanced algorithms such as MF, IN, YOLOV7, BYTE tracking, T-SNEVJ, SVR with ED, and GCRCNN.

This unique combination allows for the simultaneous recognition of multiple emotions within a single video frame, effectively addressing the complexities of real-world scenarios that traditional methods often fail to handle. By incorporating these advanced techniques, the proposed system overcomes the pitfalls of existing methods, providing a robust solution for recognizing and classifying multiple facial emotions in dynamic video environments.

The framework's effectiveness is demonstrated through comprehensive evaluations, which highlight its superiority over traditional FER techniques. This integration ensures seamless data processing, high efficiency, consistent performance metrics, and reduced error rates, all while being cost-effective.

These enhancements clearly illustrate the novelty and superior performance of our proposed FER framework, distinguishing it from the prior art and making it a practical solution for various real-world applications. The rest of this paper is organized as follows: Section 2 describes the proposed FER methodology, Section 3 discusses the related works and their limits, Section 4 presents the results and discusses them, and finally, Section 5 concludes the proposed work with future development.

2. Proposed Methodology

The proposed work is mainly used to detect and classify human facial emotions accurately. The research methodology involves significant processes, such as pre-processing, human detection and tracking, human face detection, facial landmark extraction, mesh generation, feature extraction, feature point tracking, motion analysis, and GCRCNN algorithms for efficiently classifying facial emotions in videos. The diagram of the proposed work is shown in Figure 1.

2.1. Key Frame Extraction

Initially, input video is converted into image frames. This involves the extraction of individual frames from the video sequence. Then, the keyframes are extracted. These frames are essential for capturing the most salient information and processing the video content. The input image keyframes K are expressed as,

$$K = K_1, K_2, \dots \dots K_e \quad (1)$$

Here, e represents the total number of K.

2.2. Pre-Processing

Next, the extracted key frames are pre-processed using MF and IN algorithms. This enhances the images for subsequent analysis and more accurate image recognition. The pre-processing phase contains grayscale conversion, noise removal, and enhancement of images. These are described below:

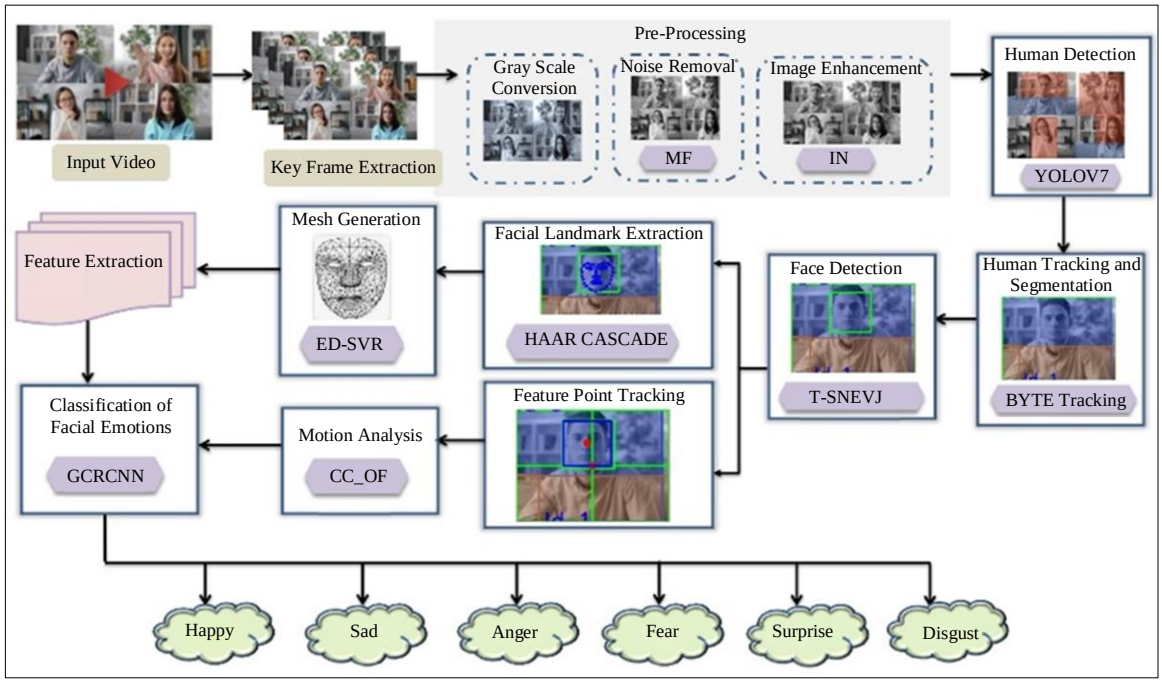


Fig. 1 Proposed diagram for facial emotion recognition

2.3. Gray Scale Conversion

Firstly, *K*they are converted into grayscale images. Converting an image to grayscale improves the image's interpretability, which leads to better visual distinction. The grayscale-converted images R_g are given as,

$$R_g = R_1, R_2, \dots \dots R_s \quad where\ g = 1\ to\ s \qquad (2)$$

Here, s represents the total number of R_g.

2.4. Noise Removal

After the grayscale conversion, the noise removal R_g is done using the MF. The MF is an image noise removal technique that replaces each pixel's value with the median of its neighboring pixel values, resulting in a smoother image with reduced noise interference and retaining sharp details from the more refined representation of the image. The denoised images are given by,

$$R_n(m', n') = M_e\big(f''(m'', n'')\big) \qquad (3)$$

Where, $R_n(m', n')$ represents the denoised output with image pixels m', n', M_e the non-linear MF operation, and $f''(m'', n'')$ the input image with pixels m'', n''.

2.5. Image Enhancement

After removing the noise, image enhancement $R_n(m', n')$ using the IN algorithm is done. The IN algorithm adjusts the pixel values of an image to improve the overall visual quality. The input-denoised image that must be enhanced is denoted as $I_{d'}$ and is given by,

$$I_{d'} = I_1, I_2, \dots \dots I_l \quad where\ d' = 1\ to\ l \qquad (4)$$

Here, l represents the total number of $I_{d'}$. Now, the output of enhanced images $I_{enhance}$ using the IN algorithm is given by,

$$I_{enhance} = \frac{I_{d'} - I_{d'min}}{Id'_{min\, d'_{max}}} \qquad (5)$$

Here, $Id'_{max\, d'_{min}}$ it represents the minimum and maximum intensity values in the image. Then, from the enhanced image, humans are detected using the following algorithms.

2.6. Human Detection

Now, $I_{enhance}$ humans are detected using the YOLOV7 algorithm. YOLOV7 is an RT object detection algorithm that efficiently detects and localizes various objects, including humans, in images and video frames through a single neural network. Utilizing an efficient layer aggregation network, the technique balances speed and accuracy.

This layer aggregation network focuses on managing the lengths of both the shortest and longest gradient paths so that it can provide effective convergence and learning within the model. The YOLOv7 has many layers, including convolutional, pooling, activation, and batch normalization. The algorithm of YOLOV7 is described below:

- Firstly, the input $I_{enhance}$ is denoted as I_f. This can be described as,

$$I_f = I_1, I_2, \dots \dots I_b \quad where \quad f = 1 \ to \ b \tag{6}$$

Here, it *b*represents the total number of enhanced images.

- Then, the input I_f is given to the convolution layer (ς). This is used in human detection by extracting spatial information from images. The output of the convolutional layer $\varsigma(i,j)$ with image pixels (i,j) for detecting humans is expressed as,

$$\varsigma(i,j) = (I_f * \chi)(i,j) = \sum x' \sum y' \ I_f(i + x', j + y') \ . \chi(x', y') \tag{7}$$

Where, χ represents the convolutional kernel used for detecting the human features and (x', y') are the kernel indices.

- After convolution, a max-pooling is performed. This layer is used to downsample the feature maps to retain essential spatial information. The formula gives this,

$$\varsigma(i,j) = max_{x',y'}\big(I_f[i * s_t + x', j * s_t + y']\big) \tag{8}$$

Here, it maxrepresents the maximum pooling function, and s_t represents the step size.

- Then, the Rectified Linear Unit $(Re\,L\,U)$ activation function provides non-linearity and detects human-related features.

$$\Lambda = Re\,L\,U\ (\varsigma(i,j)) \tag{9}$$

Here, Λ represents the output of $Re\,L\,U$activation. The activation function $Re\,L\,U$is given as,

$$Re\,L\,U(h) = max(\,0, h) \tag{10}$$

Here, h represents the input value, $max$the maximum of h, and $Re\,L\,U$ the output of the activation function.

- Now, batch normalization is done to normalize the output of the previous activation layer. This aids in stabilizing and accelerating the training process. The input acquired after activation is denoted by, ϑ and the normalized output N_o is given as,

$$N_o = \gamma \frac{\vartheta - \mu}{\sqrt{\sigma^2 + \varepsilon}} + \beta \tag{11}$$

Where, μ represents the mean of ϑ, σ^2 represents the variance to normalize $\vartheta, \gamma \ and \ \beta$ represents the scale and shift parameter of the normalized features, and ε represents a small constant to ensure numerical stability.

- Now, the loss function L_f for the given YOLOV7 algorithm is calculated. This loss function aims to optimize the model by balancing localization, confidence, and classification losses to ensure precise human detection in images through effective error minimization during training. The equations of loss functions are given as,

$$L_{local} = \lambda_{chord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left[(a_i - \hat{a}_i)^2 + \left(b_i - \hat{b}_i\right)^2\right] + \lambda_{chord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i}\right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i}_i\right)^2\right] \tag{12}$$

Where, L_{local} represents the localization loss, λ_{chord} represents the coefficient that adjusts the impact of L_{local}, S^2 represents the total number of grid cells, B is the number of bounding boxes predicted by each grid, 1_{ij}^{obj} represents an indicator function that evaluates whether a human exists in the particular grid cell represented by the indices i and j $a_i,\ b_i$, and $\sqrt{w_i}, \sqrt{h_i}$ represent the predicted centre coordinates and width and height of the bounding box for the i^{th} grid cell, and $\hat{a}_i, \hat{b}_i$ and $\sqrt{\hat{w}_i}, \sqrt{\hat{h}_i}$ represent the ground truth center coordinates and width and height of the human subject in the same grid cell. The confidence loss $L_{confidence}$ is computed by,

$$L_{confidence} = \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left[\left(C_i - \hat{C}_i\right)^2\right] + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{noobj} \left[\left(C_i - \hat{C}_i\right)^2\right] \tag{13}$$

Where, λ_{noobj} represents a coefficient that fine-tunes the impact of the $L_{confidence}$ in grid cells where no humans are present, 1_{ij}^{noobj} represents an indicator function that evaluates whether a human subject is absent in the particular grid cell and $C_i\hat{C}_i$ represents the predicted and ground truth confidence scores for the bounding box in the i^{th} grid cell. From $L_{confidence}$, the classification loss $L_{classify}$ is calculated using Equation (14),

$$L_{classify} = \sum_{1=0}^{S^2} 1_i^{obj} \sum_{z \in classes}^{S^2} \left(p_i(z) - \hat{p}_i(z)\right)^2 \tag{14}$$

Where, $p_i(z) \ and \ \hat{p}_i(z)$ represents the class's predicted and ground truth probability for human subjects in the i^{th} grid cell.

- After identifying losses, the object detection phase is used to precisely identify and locate human subjects within images for accurate human detection. The detected

humans in an image can be denoted as H_d and is expressed as,

$$H_d = H_1, H_2, \dots \dots H_{c'} \quad where\ d = 1\ to\ c' \tag{15}$$

Here, c' represents the total count ofH_d.

2.7. Human Tracking and Segmentation

After H_d, the human tracking and segmentation are done using the BYTE tracking algorithm. BYTE Track is a multi-object tracker algorithm that uses tracking-by-detection to maintain tracks for multiple objects within a video sequence. The tracking algorithm uses lower-confidence detections to supplement and confirm established tracks based on higher-confidence detections.

The detection input includes specific human-related features, such as bounding boxes, confidence scores, and critical point information for human pose estimation. These features are crucial for accurately identifying and tracking human subjects within images. The detection input $D'$is expressed as,

$$D' = \{(a_i, b_i, w_i, h_i, C_i, k_i, p_{i1}, p_{i2}, \dots \dots p_{iz})\} \tag{16}$$

Where, a_i, b_i, w_i, h_i represent the bounding box coordinates with width and height for i^{th} human detection, C_i the confidence score, k_i the key point information for human pose estimation and $p_{i1}, p_{i2}, \dots \dots p_{iz}$ the z number of class probabilities for different human classes.

2.7.1. Initialization

Initially, the specific human features are considered, and the output S_j from the human detected input D'_l with feature points $l\ and\ j$is given as,

$$S_j = \{D'_l\} \tag{17}$$

2.7.2. Maintenance and Association

Now, the distance metrics for $S_j$and $D'_l$are calculated. This includes human-specific features like pose consistency and body measurements to accurately associate detections by customizing the distance metric to accommodate these characteristics. The distance M_{ij} between S_j and D'_l is expressed as,

$$M_{ij} = d''\left(S_j, D'_l\right) \tag{18}$$

Here, d''it represents the distance metrics.

2.7.3. Confidence Based Fusion

Then, the fusion process for human tracking is done. The weights w_g assigned to different detection confidence scores can be adjusted to prioritize the stability and consistency of tracked humans within the image sequence. This is given by,

$$S_{fus} = \sum_l w_{g_l}.D'_l \tag{19}$$

Here, S_{fus} represents the output of the fusion process.

2.7.4. Updating the Tracking Information

Next, the tracked information is updated using Kalman Filters (KF) to focus on human-specific movement patterns. These techniques help to enhance the accuracy and robustness of tracking human subjects within image sequences. The S_{fus} is updated using the KF for precise tracking of humans, and the updated equation is given as,

$$r^t = A'_t r_{t-1} + B'_t q_t + v_t \tag{20}$$

Where, $r_t$represents the state of the human subject at the time t, $A'_t\ and\ B'_t$ represents the state transition and control input matrixes at t, q_t represents the control input at t, and v_t represents the process noise. Thus, from the updated results, the s' numbers of tracked humans T'_H are expressed as,

$$T'_H = T'_1, T'_2, \dots \dots T'_{s'} \quad where\ H = 1\ to\ s' \tag{21}$$

Now, T'_H are segmented. The o' numbers of segmented humans δ_h in images are given as,

$$\delta_h = \delta_1, \delta_2, \dots \dots \delta_{o'} \quad where\ h = 1\ to\ o' \tag{22}$$

Here o' is the total number of segmented humans δ_h in images.

2.8. Human Face Detection

Next, facial features are detected from the segmented human images using the T-distributed Stochastic Neighbor Embedding-based Viola-Jones algorithm (T-SNEVJ). Viola Jones (VJ) is a within cascade object detector known for quickly identifying people's faces. It utilizes the Adaboost technique for effective feature reduction.

Even though it compresses numerous features into a concise format, VJ is highly susceptible to outliers. So, to tackle this issue, the T-distributed Stochastic Neighbor Embedding (T-SNE) method is used to perform the efficient feature reduction process. The algorithm steps of T-SNEVJ are discussed below:

2.8.1. Step 1: Selection of Haar-Like Features

Initially, from the δ_h, haar-like features are selected. These are simple rectangular features used to detect human facial patterns in images. They involve calculating the difference between the sum of pixel intensities in adjacent rectangular regions.

These features help to capture patterns like edges, lines, and corners in the image. The selection of haar-like features $\delta_h$is expressed as,

$$H(j',k') = \sum_{q'} w_{q'} * \wp_{q'}(j',k') \quad (23)$$

Where, $H(j',k')$ represents the Haar-like feature value at the position (j',k'), q' refers to the index representing the individual Haar-like features, such as edge, line, or corner features, $w_{q'}$ represents the weight in the rectangular region and $\wp_{q'}$ represents the pixel intensity.

2.8.2. Step 2: Generation of an Integral Image

Next, the integral image is generated. This is a method used to speed up the computation of $H(j',k')$. This approach enables the rapid calculation of the sum of pixels within any rectangular region by facilitating efficient feature analysis and is given as,

$$q''(j',k') = \sum_{q'=0}^{j'} \sum_{r'=0}^{k'} q''(j',k') \quad (24)$$

Where, $q''(j',k')$ represents the value of the integral image at a specific position (j',k'), and $q'\ and\ r'$ represent the feature points of pixel intensities within the rectangular region.

2.8.3. Step 3: T-SNEVJ Calculation

Now, the similarities between the integral image features are computed. Adaboost techniques are susceptible to noise and outliers, potentially affecting the detection process's overall accuracy. Hence, T-SNE-based VJ is used in human face detection to effectively visualize and cluster high-dimensional data for improved pattern recognition and analysis.

The formula calculates the conditional similarity $F_{y'|x'}$ between integral image features $u_{x'}, u_{y'}$ using a Gaussian kernel,

$$F_{y'|x'} = \frac{e\left(-\|u_{y'} - u_{x'}\|^2 / 2\sigma^2\right)}{\sum_{g' \neq x'} e\left(-\|u_{x'} - u_{g'}\|^2 / 2\sigma^2\right)} \quad (25)$$

Where, g'represents the index used for iterating over the different$u_{g'}$ features, and σ represents the bandwidth parameter of the Gaussian kernel.

2.8.4. Step 4: Development of Haar Cascade Classifier

Then, the Haar Cascade (HC) classifiers are developed to discard non-face regions in the detection process efficiently. The HC classifier uses a robust classifier combining multiple weak classifiers to detect facial landmarks. This is given by,

$$H_{(\dot{c})} = \sum_{f=1}^{n} \alpha_f * \delta\left(F'_f(\dot{c})\right) \quad (26)$$

Here, $H_{(\dot{c})}$ represents the final cascade classifier output, n represents the total number of week classifiers, α_f represents the weight associated with the f^{th} weak classifier, δ represents the indicator function that outputs 1, if the condition is satisfied and 0 if not satisfied, and $\left(F'_f(\dot{c})\right)$ represents the weak classifier having (F') a threshold function.

Thus, the output of the u'number of detected faces D_f of humans is expressed as,

$$D_f = D_1, D_2, \ldots \ldots D_{u'}, \quad where\ f = 1\ to\ u' \quad (27)$$

Pseudo Code of T-SNEVJ:

Input : Input the segmented human images, δ_h
Output : HC output, $H_{(\dot{c})}$

Begin
Initializeδ_h,H,j',k'
While $\tau < \tau^{max}$
Select Haar features, H
$$H(j',k') = \sum_{q'} w_{q'} * \wp_{q'}(j',k')$$
Generate integral image, Q
$$Qq''(j',k') = \sum_{q'=0}^{j'} \sum_{r'=0}^{k'} q''(j',k')$$
Calculate conditional similarity $F_{y'|x'}$
$$F_{y'|x'} = \frac{e\left(-\|u_{y'} - u_{x'}\|^2 / 2\sigma^2\right)}{\sum_{g' \neq x'} e\left(-\|u_{x'} - u_{g'}\|^2 / 2\sigma^2\right)}$$
Develop Haar Cascade Classifier
$$H_{(\dot{c})} = \sum_{f=1}^{n} \alpha_f * \delta\left(F'_f(\dot{c})\right)$$
End while
Return $\rightarrow H_{(\dot{c})}$
End

2.8.5. Feature Point Tracking and Motion Analysis

From $H_{(\dot{c})}$, feature point tracking followed by motion analysis is done using the Concordance Correlation Coefficient based Optical Flow (CC_OF) technique. Feature point tracking, also known as specific facial landmarks, monitors and analyzes facial movement for expression recognition.

The p'' number of tracked feature points $F_{f''}$ from the face are expressed as,

$$F_{f''} = F_1, F_2, \ldots \ldots F_{p''}, \quad where\ f'' = 1\ to\ p'' \quad (28)$$

Now, the motions are analyzed using the CC_OF algorithm from the facial tracked points. Optical Flow (OF) techniques are widely used in extracting motion information. However, OF has difficulty accurately capturing subtle facial expressions and abrupt changes in lighting conditions.

Hence, Concordance Correlated (CC) based OF is used to overcome this issue. The analysis of the CC in emotion analysis is used to identify and quantify the strength and nature of relationships between specific emotional expressions. Now, from the tracked features, the motions are analyzed by the CC_OF algorithm, which is described below:

1) Initially, the tracked feature points are taken for motion analysis. Here, the OF vectors are calculated to estimate the motion of facial features. The OF equation for the tracked facial features is given as,

$$J_k V_k + J_t V_t + J_{t'} = 0 \tag{29}$$

Where, J_k,J_t represent the partial derivatives of image intensity in the $k\ and\ t$directions, V_k, V_t represent the velocities in the $k\ and\ t$directions, and $J_{t'}$ represents the change in intensity over timet'.

2) Now, integrate the CC in OF for efficient motion analysis. OF techniques are susceptible to noise due to their reliance on intensity gradients, which can lead to inaccuracies in the motion analysis of $F_{f''}$ images. Thus, CC coefficient based OF is used for practical motion analysis to overcome this issue. The formula expresses this,

$$\Im_p = \frac{2*\sigma'_{kt}}{(\sigma'^2{}_k+\sigma'^2{}_t+(\mu_k-\mu_t)^2)} \tag{30}$$

Where, $\Im_p$ represents the motion analyzed output, σ' the covariance between the observed and predicted motions, and $\mu$the respective mean.

2.9. Facial Landmark Extraction

In the meantime, from $H_{(\dot{c})}$, facial landmarks are extracted using HC classifiers. These are used to detect specific facial features by analyzing patterns of pixel intensities. The purpose of HC classifiers includes accurately detecting facial features and enabling precise landmark localization for tasks like facial recognition.

The working steps of the HC classifier are already discussed above. The $\dot{q}$ numbers of extracted facial landmarks $L_{\dot{g}}$are denoted as,

$$L_{\dot{g}} = L_1, L_2, \dots \dots L_{\dot{q}} \quad where\ \dot{g} = 1\ to\ \dot{q} \tag{31}$$

2.10. Mesh Generation and Feature Extraction

Then, from $L_{\dot{g}}$, the mesh is generated using the Euclidean Distance-based Sparse Voronoi Refinement (ED-SVR) algorithm. Sparse Voronoi Refinement (SVR) is a mesh generation technique that selectively refines regions of interest. However, SVR includes the potential loss of fine details.

To tackle this issue, Euclidean Distance (ED) is used in SVR to determine the proximity of landmarks, thus aiding in the accurate placement of vertices for mesh construction. The algorithm of ED-SVR is discussed below:

i) Firstly, a Q' set of initial point features (e_p, o_p) is given as,

$$Q' = (e_p, o_p) \tag{32}$$

ii) Then, for each point $Q' = (e_p, o_p)$, calculate the Voronoi region V_p. Then, the regions are identified based on the break phase. Using the farthest point during Steiner point construction impacted SVR's meshing performance in the break phase. Hence, ED-SVR measures the gap between the active and farthest points. The distance D''_p is expressed as,

$$D''_p = \sqrt{(e_{far} - e_{act})^2 + (o_{far} - o_{act})^2} \tag{33}$$

Here, o_{far}, o_{act}, $e_{far} and\ e_{act}$ represent the active and farthest points in the 2D space.

iii) After this, the clean phase is performed. This phase uses a Gaussian Filter (GF) to ensure more accurate mesh generation and is given by,

$$G(e,o) = \frac{1}{2\pi\sigma^2}\sum_{p=-n}^{n}\sum_{q=-n}^{n} G(e+p, o+q)\, exp\left(\frac{-(p^2+q^2)}{2\sigma^2}\right) \tag{34}$$

Where, $G(e,o)$ represents the output of the clean phase using GF with pixels (e,o), σ represents the standard deviation of GF, n represents the Gaussian kernel with relative positions p, q regarding (e,o), and exprepresents the exponential value in GF.

Thus, the generated mesh can be denoted by M_g. Subsequently, various features, including geometric aspects and the angles formed by specific edges, are extracted from M_g and denoted by E_g. The g'' numbers of extracted E_g are expressed as,

$$E_{\dot{g}} = E_1, E_2, \dots \dots E_{g''} \quad where\ g = 1\ to\ g'' \tag{35}$$

2.11. Classification of Facial Emotions

Finally, the extracted features and motion-analyzed output are given to the GCRCNN algorithm to classify facial emotions effectively. CNN is commonly used to classify facial emotions due to its automatic ability to learn accurately.

However, CNN needs a large amount of data for practical training. Hence, to avoid this issue, Graph Convolution and

Regular 1-dimensional (GCR) based CNN are utilized to classify various facial emotions. The purpose of GCRCNN in classifying facial emotions is to use simplified visual representations to capture and classify key emotions from facial expressions efficiently. The classifier diagram of the GCRCNN algorithm is shown below:

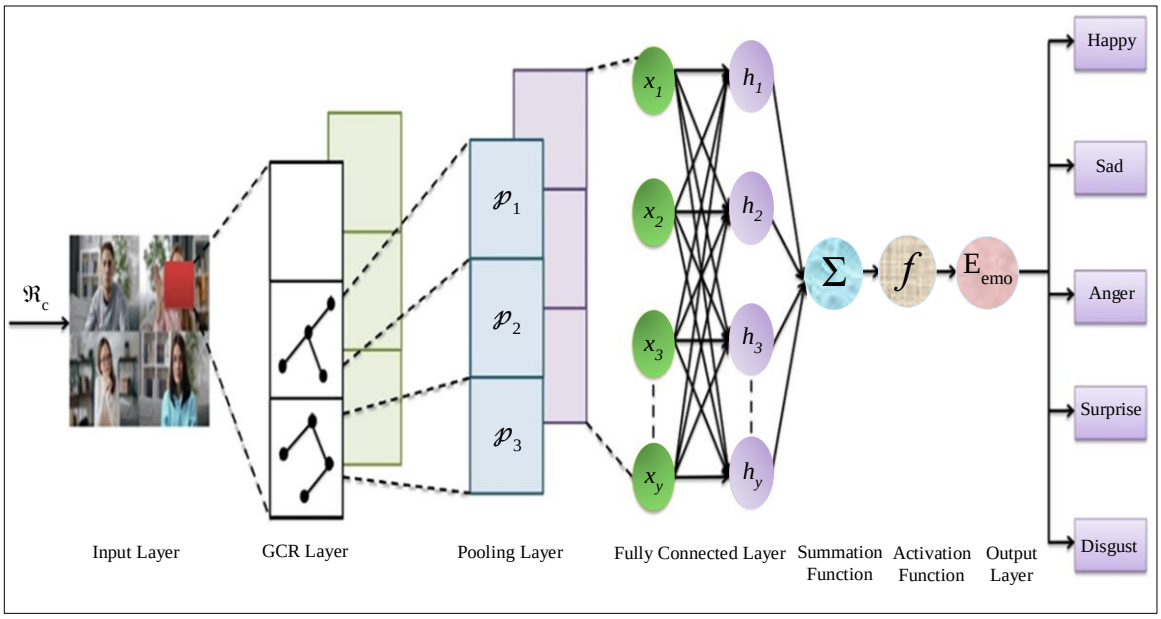


Fig. 2 GCRCNN classifier

The algorithm of GCRCNN is discussed below:

1) Initialization: Initially, the input of $E_g$and $\Im_p$is denoted by $\Re_c$and is expressed as,

$$\Re_c = \Re_1, \Re_2, \dots \dots \Re_{a'} \quad where\ c = 1\ to\ a' \tag{36}$$

Here, a' represents the total number of $\Re_c$.

2) Graphical-Based 1D Layer: Then, the input $\Re_c$ is given to the GCR layer. The normal CNN requires a substantial number of filters for deeper convolutions, which leads to increased training times. Hence, GCR-based CNN is used.

This can effectively process graph-structured data. Thus, it enables more efficient analysis and classification of facial emotions. The output of the GCR layer $(\xi)$is given by,

$$\xi^{\tau+1} = \sigma'\left(\widehat{D}' - \frac{1}{2}\hat{A}\widehat{D}' - \frac{1}{2}\xi^{\tau}W'^{\tau}\right) \tag{37}$$

Where, $\xi^{\tau+1}$ represents the output GCR at the layer $\tau + 1$, $\widehat{D}'$ denotes the diagonal degree of the matrix $\hat{A}$, $\hat{A}$ represents the sum of the adjacency matrix $\hat{A^{\prime}}$ and identity matrix $\hat{I^{\prime}}$, W'^{τ} represents the weight matrix at the layer τ, and σ' represents the sigmoidal activation function that introduces non-linearity, allowing the model to capture complex relationships between facial features. The sigmoidal activation function is given by,

$$\sigma'(\dot{h}) = \frac{1}{1+e^{-\dot{h}}} \tag{38}$$

Where, $(\dot{h})$ represents the input function, and e represents the exponential value.

3) Pooling Layer: After GCR convolution, the max-pooling is performed, which is expressed as,

$$\wp = max_{\dot{a},\dot{b}}\, \Re_c \left[\Psi * s^t + \dot{a}, \nu * s^t + \dot{b}, \varpi\right] \tag{39}$$

Where $\wp$ represents the output pooling layer, $max_{\dot{a},\dot{b}}$ the maximum pooling with feature points $\dot{a}, \dot{b}, s^t$ the step size of the pooling layer, and $\varpi$the dimension in $\Re_c$.

4) Activation: Then, the $(Re\,L\,U)$ activation function is calculated. The output activation function λ' is given by,

$$\lambda' = Re\,L\,U(\wp) \tag{40}$$

5) Fully Connected Layer: Lastly, the fully connected layer, along with software activation, is calculated. This layer aids in classifying facial emotions by providing a robust framework for multi-class emotion recognition with high accuracy. This is expressed as,

$$F_y = S_{sof}(F''W^T + B^T) \quad (41)$$

Where, F_y represents the output of a fully connected layer, S_{sof} represents the softmax activation function, F'' represents the flattened feature matrix, W^T represents the weights of a fully connected layer and B^Trepresents the bias term. The softmax activation function is given by,

$$\dot{\rho}(f_\Psi) = \frac{\dot{e}^{f\Psi}}{\sum_{\nu}\dot{e}^{f\nu}} \quad (42)$$

Where, $\dot{\rho}(f_\Psi)$ represents the probability of input belonging to Ψ^{th} a class, $\dot{e}$ represents the exponential value, ν represents the different classes in the classification task, and ν varies from $1\ to\ \dot{n}$ several classes.

6) Output: Hence, the classified results of facial emotions, such as happiness, sad, anger, fear, surprise, and disgust are received, and these are denoted by E_{emo}.

Pseudo Code of GCRCNN Algorithm:

Input : Input E_g and $\Im_p$
Output : Classified emotions, E_{emo}
Begin
While $\psi' < \psi' max$
Initialize $\Re_c$,ξ
Calculate GCR,

$$\xi^{\tau+1} = \sigma'\left(\widehat{D}' - \frac{1}{2}\hat{A}\widehat{D}' - \frac{1}{2}\xi^{\tau}W'^{\tau}\right)$$

Compute Max-pooling,

$$\wp = max_{\dot{a},\dot{b}}\,\Re_c\left[\Psi * s^t + \dot{a}, \nu * s^t + \dot{b}, \varpi\right]$$

Evaluate F_y

$$F_y = S_{sof}(F''W^T + B^T)$$

End while
Return $\rightarrow E_{emo}$
End

3. Literature Survey

FER has garnered significant attention due to its applications in human-computer interaction, psychological studies, and security. Traditional FER techniques primarily utilized machine learning algorithms like SVM, RNN, and LSTM networks. These methods have been somewhat successful in classifying basic emotions but often falter when it comes to recognizing multiple emotions simultaneously.

Yu et al. [16] introduced a practical FER framework named Multi-Task Global-Local Network (MTGLN), which utilized the LSTM algorithm with Part Based Module (PBM) and Global Face Module (GFM) techniques for facial emotion detection. PBM efficiently extracted features from the eyes, ears, and mouth regions, resulting in higher accuracy in classifying emotions.

However, the use of two datasets introduced complexities in data integration, affecting the overall efficiency and accuracy of the system.

Building on deep learning techniques, Mehendale et al. [17] established a CNN-based FER approach for emotion classification. This method involved pooling and flattening layers to extract essential features from images, successfully identifying key facial features and minimizing computational complexity. Although this model showed improved classification results, the increasing number of layers led to longer execution times, causing delays in real-time applications where faster detection is crucial.

Zhang et al. [18] proposed a real-time video emotion recognition framework based on Reinforcement Learning and Domain Knowledge (ERLDK), employing a Dueling Deep-Q-Network (DDQN) with Gated Recurrent Unit (GRU) layers to learn the correct actions from various emotion categories. This approach, leveraging reinforcement learning and domain knowledge, achieved higher accuracy than other state-of-the-art methods. However, the inconsistency in F1 scores posed challenges in reliably assessing the model's overall performance.

Zeng et al. [19] developed Emotion Coherence (EmoCo), an interactive visual analytics system for recognizing emotions in presentation videos. By combining facial, text, and audio modalities, this system efficiently explored and compared emotional expressions through a channel coherence view and sentence clustering view. Despite its effectiveness in detecting and analyzing facial expressions, the merging of different data sources led to false interpretations that did not accurately represent the uniqueness of certain emotions.

López-Gil et al. [20] introduced a FER method using parameterized photograms and machine learning techniques, focusing on facial feature-based emotional classification. This model achieved high emotion recognition rates, showing improved classification accuracy compared to other works. However, the absence of a pre-processing step increased noise, reducing the overall robustness of emotion detection.

Li et al. [21] developed the Spontaneous Driver Emotion Facial Expression (DEFE) framework by analyzing video clips collected during driving scenarios. This work incorporated Self-Assessment Manikin and Differential Emotion Scale techniques to enhance driver safety by detecting human facial emotions. While the results showed perfect accuracy in recognizing driver emotions, the lack of mesh generation led to inadequate representation of intricate facial features, impacting the precision of the analysis.

Hu et al. [22] proposed a framework using a Two-Stage Spatio-Temporal Attention CNN (SATCN) for continuous emotion recognition in facial videos. The model utilized TS

and SATCN algorithms for emotion classification, employing the Adam Optimizer with a learning rate for an efficient training process. Although the results demonstrated lower error rates and higher accuracy, the occurrence of higher Mean Squared Error (MSE) values indicated suboptimal accuracy for FER.

Mohan et al. [23] introduced a Local Gravitational Force descriptor-based Deep-CNN for efficient emotion recognition. This method employed Dynamic Bayesian Network (DBN) and CNN techniques, excelling in classifying seven types of emotions. The results showed higher accuracy, precision, and recall in controlled lab environments. However, the technique struggled to detect emotions in natural and uncontrolled environments.

Samadiani et al. [24] demonstrated a Happy Emotion Recognition model using a 3-dimensional Hybrid Deep and Distance Features (HappyER-DDF) method for detecting happy facial emotions. By combining a 3D neural network and LSTM for spatial-temporal feature extraction and tracking facial landmark changes using distance calculations, this model offered better detection accuracy compared to other works. However, it focused solely on happy face detection, neglecting other emotions.

Lee et al. [25] introduced a Multimodal Recurrent Attention Network (MRAN) for FER, employing techniques such as Deep Neural Networks (DNN), LSTM, and Multimodal Arousal Valence (MAVFER). This model detected emotions based on color, depth, and thermal recordings of videos with arousal-valence scores. Despite its superior performance in detecting multimodal facial emotions, the use of expensive techniques like DNN and LSTM limited its widespread usage.

Compared to existing works, our proposed framework addresses several critical issues commonly found in traditional approaches. Firstly, it overcomes data integration challenges that many previous methods struggled with, ensuring seamless and accurate processing of diverse datasets. Secondly, our framework is optimized for efficiency, significantly reducing execution times and making it suitable for real-time applications.

Unlike some existing techniques that suffer from inconsistency in performance metrics, our approach maintains high consistency and reliability across various evaluation parameters. Additionally, it demonstrates a significant reduction in errors, offering more precise and accurate emotion recognition.

Furthermore, our solution is cost-effective, providing superior performance without the need for expensive and complex techniques. These enhancements clearly illustrate the novelty and superior performance of our proposed FER framework, setting it apart from prior art and making it a robust and practical solution for real-world applications.

4. Result and Discussion

In this section, the performance analysis of the proposed method is carried out to evaluate the model's reliability. The proposed work will be implemented on the PYTHON platform.

4.1. Dataset Description

The dataset used in the proposed work is Ryerson Emotion, collected from publicly available sources. This dataset classifies the six principal emotions: happiness, sadness, anger, fear, surprise, and disgust. The proposed work uses 600 videos, of which l00 are taken for each emotion class.

From this, the proposed work uses 80% of the videos for training purposes, and 20% are considered for the testing phase; about 470 samples are provided for training and 120 videos for testing.

4.2. Performance Analysis of the Proposed Work

This section compares the performance of the proposed T-SNEVJ, CC_OF, and GCRCNN with other related works. Table 1 depicts the FER from the input keyframes. The noise from the input image is first removed using the MF algorithm, and the images are enhanced using the IN algorithm. YOLOV7 and BYTE tracking algorithms do human detection and tracking.

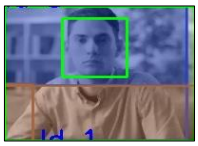
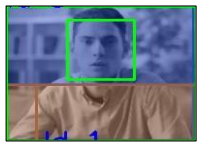
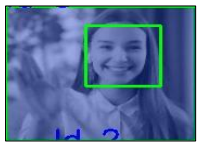
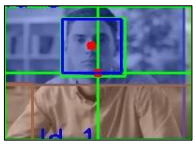
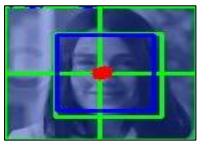
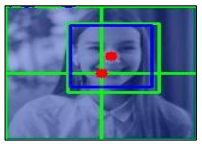
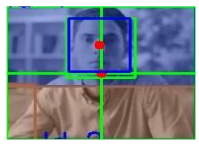
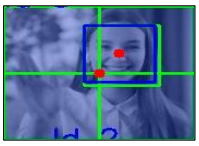
From the tracked image, the human face is detected using T-SNEVJ. Then, the HC technique extracts facial landmarks from the face-detected output. In the meantime, feature points are tracked, followed by motion analysis.

Figure 3 compares the performance metrics of the proposed and state-of-the-art works. The GCRCNN algorithm achieves higher accuracy, precision, recall, F-measure, sensitivity, and specificity rates of 99.34258583%, 99.49016752%, 99.20116195%, 99.34545455%, 99.20116195%, and 99.4856723% when compared with other works.

The proposed GCRCNN improves classification accuracy by recognizing different facial emotions. Thus, the proposed GCRCNN performed well compared to existing algorithms like CNN, Restricted Boltzmann Machine (RBN), LSTM, and DNN. The existing works exhibited lower accuracy of 97.49077491%, 95.25547445%, 93.92097264%, and 91.20300752%, followed by other metrics.

In Table 2, a comparison of the True Positive Rate (TPR), True Negative Rate (TNR), False Positive Rate (FPR), and False Negative Rate (FNR) between the proposed GCRCNN and existing techniques is shown.

Table 1. Image results of the proposed work

Facial Emotion Recognition					
Keyframes					
Removed Noise					
Enhanced Image					
Human Detection					
Human Tracking					
Face Detection					
Facial Landmark					
Feature Point					

The TPR and TNR measure the true positive and true negative predictions, while the FPR and FNR assess the rate of false positive and false negative predictions of the proposed GCRCNN. The proposed model achieves higher rates with TPR and TNR of 99.20116195% and 99.4856723% and lower FPR and FNR of 0.5143277% and 0.798838054%, surpassing traditional methods that exhibit lower TPR and TNR and higher FPR and FNR. In Figure 4, the ER of the proposed GCRCNN is shown. The lower ER in the proposed work depicts a more accurate and reliable FER. The work showcased a lower ER of 0.006574142% compared to existing techniques. Therefore, GCRCNN stands out as the top-performing approach among all methods.

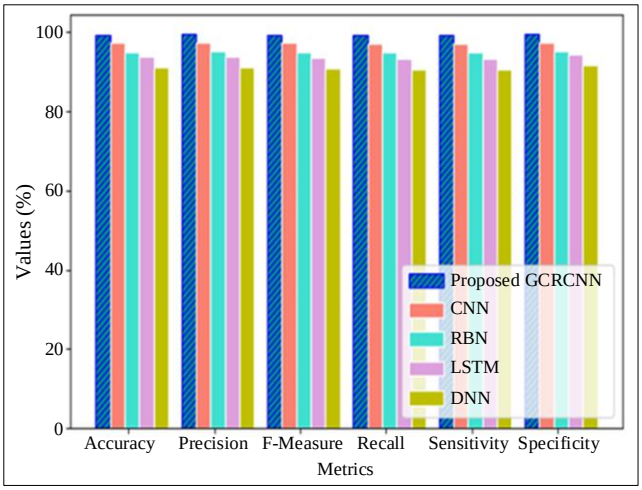


Fig. 3 Performance metrics of GCRCNN

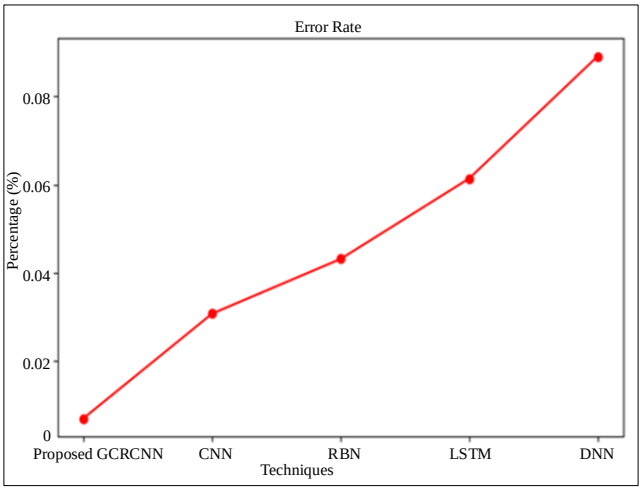


Fig. 4 Error Rate (ER)

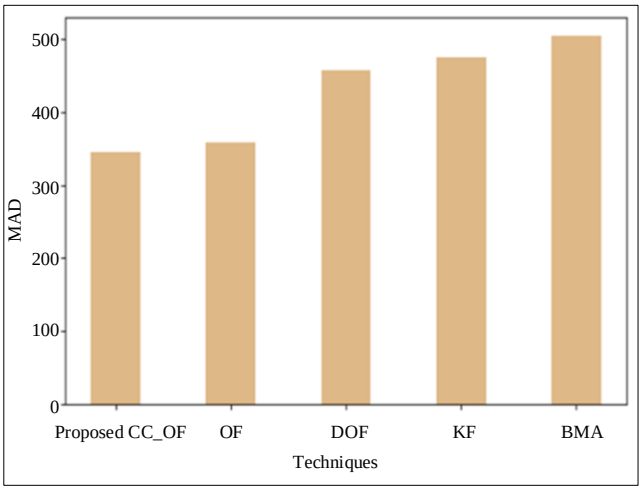


Fig. 5(a) MAD calculation

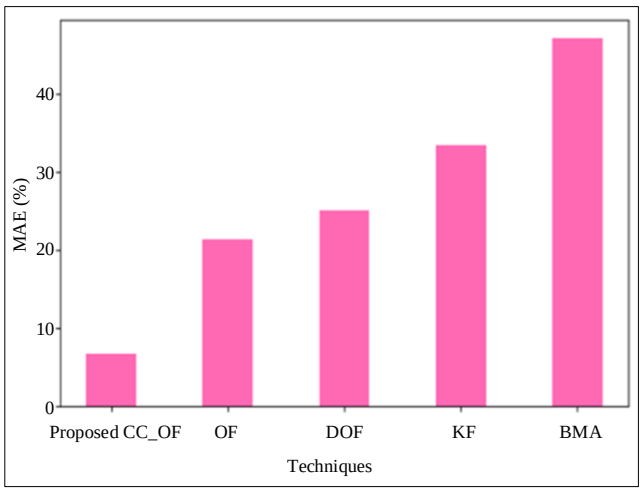


Fig. 5(b) MAE analysis

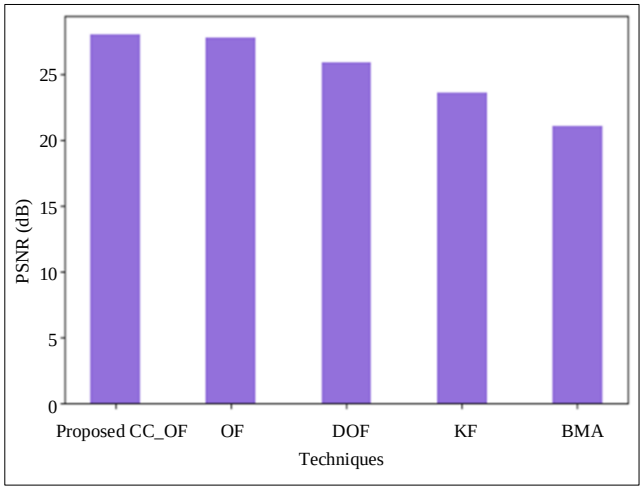


Fig. 5(c) PSNR comparison

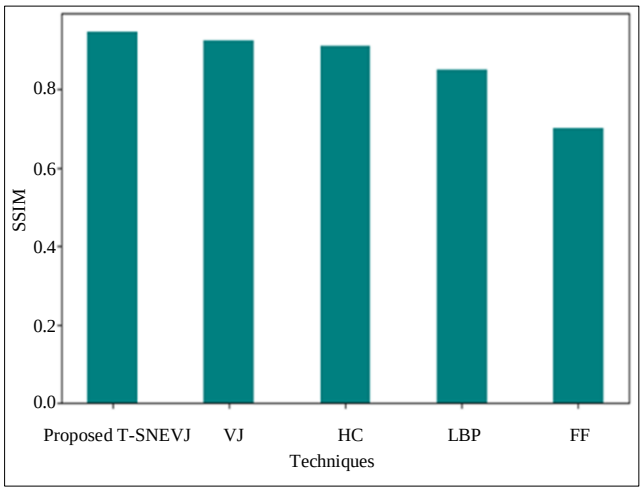


Fig. 6 SSIM validation of the proposed T-SNEVJ

Table 2. Comparative analysis based on TPR, TNR, FPR, and FNR

Algorithms	TPR (%)	TNR (%)	FPR (%)	FNR (%)
Proposed GCRCNN	99.20116195	99.4856723	0.5143277	0.798838054
CNN	97.06098457	97.5308642	2.469135802	2.93901543
RBN	94.84011628	95.22760646	4.772393539	5.159883721
LSTM	93.42403628	94.34628975	5.653710247	6.575963719
DNN	90.72550486	91.64882227	8.35117773	9.274495138

The graphs in Figures 5(a), 5(b), and 5(c) compare the performance of the proposed CC_OF's Mean Absolute Deviation (MAD), Mean Absolute Error (MAE), and Peak Signal-to-Noise Ratio (PSNR) with existing methods, such as OF, Degrees of Freedom (DOF), KF, and Block Matching Algorithm (BMA).

The proposed CC_OF has higher PSNR, lower MAE, and lower MAD values, which are used to assess absolute error-free analysed motions. The proposed methodology has a PSNR value of 28.01555748 decibels (dB), MAE of 6.723764744%, and MAD of 344.8084484, which outperforms the existing methodologies that had the highest MAD and MAE values and lowest PSNR. Here, concordance correlation is used by considering the direction of flow vectors of motion.

Analysing the minute movements allows the model to achieve higher PSNR with lower errors. Thus, the proposed CC_OF has effective performance when compared to other traditional methods.

Figure 6 represents the validation of the proposed work's Structural Similarity Index (SSIM). In the proposed T-SNEVJ, SSIM is used to evaluate the structural similarity between the tracked facial features. The proposed work shows a higher SSIM of 0.945915081 when analyzing motions compared to VJ, HC, Local Binary Pattern (LBP), and Fisher Face (FF). Incorporating the T-SNE-based feature reduction technique improves the performance of the conventional VJ technique in detecting the human face. Thus, the results determined that the proposed work is more effective regarding SSIM validation than all other state-of-the-art techniques.

Table 3. Comparative analysis with existing works

Techniques	Methods Used	Dataset Used	Precision (%)	F-Measure (%)	ER (%)	Accuracy (%)
Proposed Work	GCRCNN, T-SNEVJ, CC_OF	Ryerson Emotion	99.49	99.34	0.006	99.34
[26]	GAN,	FER	-	-	0.03	93.66
[27]	SVM, CNN	SAVEE	-	-	0.07	98.77
[28]	CNN and DNN based FER	-	96.42	-	-	90
[29]	DNN	CASE II	84.71	-	0.12	83.3
[30]	SMAF	IEMOCAP	-	84.53	-	85.66

Table 3 evaluates the performance of the proposed work by comparing it with other associated works. The proposed work utilized the efficient method named GCRCNN to ensure the maximum accuracy, precision, F-measure, and ER of 99.34%, 99.49%, 99.34%, and 0.006%, which precisely classify the facial emotions. The proposed work used the Ryerson Emotion dataset for efficient classification of emotions. The existing methods, like CNN and DNN, showed a lower average accuracy of 94.38%, and Self-Multi-Attention Fusion (SMAF) showed a lower F-measure of 84.53%, which affects the model's efficacy. The traditional work used a Generative Adversarial Network (GAN) with a higher ER of 0.03% and a lower accuracy of 93.66%.

Hence, GAN affected the accuracy of FER's predictions. Thus, the proposed work incorporates mesh generation, motion analysis, and GCR with CNN to detect and classify human facial emotions efficiently. Hence, the experimental analysis proved that the proposed work is practical in FER compared to other related works.

Our research introduces a novel framework for multi-facial emotion recognition and classification by integrating advanced algorithms. This unique combination addresses key limitations in existing methods, providing superior performance in real-world scenarios. Unlike traditional approaches that focus on a single emotion per frame, our

framework simultaneously recognizes multiple emotions within a video frame. YOLOV7 and BYTE tracking ensure precise detection and continuous monitoring of multiple individuals.

The pre-processing stage, utilizing MF and IN algorithms, enhances input frame quality by removing noise and normalizing intensity values, leading to more accurate human and facial feature detection. Facial landmark detection using T-SNEVJ combines T-SNE and Viola-Jones algorithms, allowing for efficient feature reduction and robust face detection. This process generates detailed facial meshes refined through SVR with ED, capturing sensitive and distinct angular features crucial for accurate recognition. CC_OF is employed for feature point tracking and motion analysis, improving the precision of emotion recognition by accurately capturing subtle facial movements.

The GCRCNN algorithm effectively combines graph convolution and regular 1-dimensional convolution, leveraging spatial relationships between facial features for higher classification accuracy. This approach outperforms traditional CNNs, SVMs, RNNs, and LSTMs, achieving an accuracy rate of 99.34% and a lower error rate of 0.006%.

Our framework resolves data integration challenges, reduces execution times, maintains high consistency in performance metrics, and offers a cost-effective solution with superior performance. These advancements make our framework a robust, efficient, and practical solution for multi-facial emotion recognition in dynamic video environments.

5. Conclusion

This paper proposed a well-organized framework for recognizing and classifying human facial emotions from videos using the GCRCNN classification algorithm. The proposed work performed keyframe extraction, pre-processing, human detection, and facial feature tracking, followed by motion analysis using CC_OF. The proposed work analyses the motion effectively and attains a higher PSNR value of 28.01 dB. In the meantime, facial landmarks were extracted, mesh was generated by the ED-SVR algorithm, followed by feature extraction.

Finally, the GCRCNN classifier classified the facial emotions. The proposed work obtained a high accuracy of 99.34% with a lower error rate of 0.006% because of the inclusion of GCR, which improved the performance of the classification. So, the results proved that the proposed model performed better than all other existing methods. Hence, the proposed work attained an accurate classification of facial emotions.

5.1. Future Recommendation

However, this research methodology is only suitable for recognizing facial emotions in videos and was not concentrated on video subtitles and interpersonal communication-based emotions. Hence, in the future, this research can be expanded to understand better how people express subtle emotions when they talk to each other. Also, new ways can be developed to recognize emotions from the subtitles that appear on videos. This will help in understanding emotions better in different types of communication.

Author's Contribution

JSB gathered publicly available datasets for emotion recognition (happy, sad, angry, etc.) have developed methods using Python to assess the performance of the model. This involves metrics like accuracy, precision, and recall for each emotion category and, wrote the complete manuscript and replied to reviewer comments. PLP supervised the experiments, reviewed drafts of the manuscript, commented on the manuscript and provided guidance for submission of this manuscript.

References

[1] Hugo Carneiro, Cornelius Weber, and Stefan Wermter, "Whose Emotion Matters? Speaking Activity Localization without Prior Knowledge," *Neurocomputing*, vol. 545, pp. 1-13, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[2] Sunsern Cheamanunkul, and Sanchit Chawla, "Drowsiness Detection Using Facial Emotions and Eye Aspect Ratios," *24th International Computer Science and Engineering Conference (ICSEC)*, Bangkok, Thailand, pp. 1-4, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[3] Mateusz Piwowarski, and Patryk Wlekly, "Factors Disrupting the Effectiveness of Facial Expression Analysis in Automated Emotion Detection," *Procedia Computer Science*, vol. 207, pp. 4296-4305, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[4] Hanyu Liu, Jiabei Zeng, and Shiguang Shan, "Facial Expression Recognition for in-the-Wild Videos," *15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, Buenos Aires, Argentina, pp. 615-618, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[5] Shruti Japee et al., "Inability to Move One's Face Dampens Facial Expression Perception," *Cortex*, vol. 169, pp. 35-49, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] Surya Teja Chavali et al., "Smart Facial Emotion Recognition with Gender and Age Factor Estimation," *Procedia Computer Science*, vol. 218, pp. 113-123, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[7] Prameela Naga, Swamy Das Marri, and Raiza Borreo, "Facial Emotion Recognition Methods, Datasets, and Technologies: A Literature Survey," *Materials Today: Proceedings*, vol. 80, part 3, pp. 2824-2828, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[8] Meaad Hussein Abdul-Hadi, and Jumana Waleed, "Human Speech and Facial Emotion Recognition Technique Using SVM," *International Conference on Computer Science and Software Engineering (CSASE)*, pp. 191-196, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[9] Carmen Bisogni et al., "Emotion Recognition at a Distance: The Robustness of Machine Learning Based on Handcrafted Facial Features vs. Deep Learning Models," *Image and Vision Computing*, vol. 136, pp. 1-15, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[10] Akriti Jaiswal, A. Krishnama Raju, and Suman Deb, "Facial Emotion Detection Using Deep Learning," *International Conference for Emerging Technology (INCET)*, India, pp. 1-5, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[11] P. Kaviya, and T. Arumugaprakash, "Group Facial Emotion Analysis System Using Convolutional Neural Network," *International Conference on Trends in Electronics and Informatics (ICOEI)*, India, pp. 643-647, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Chirag Dalvi et al., "A Survey of AI-Based Facial Emotion Recognition: Features, ML & DL Techniques, Age-Wise Datasets, and Future Directions," *IEEE Access*, vol. 9, pp. 165806-165840, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[13] Chahak Gautam, and K.R. Seeja, "Facial Emotion Recognition Using Handcrafted Features and CNN," *Procedia Computer Science*, vol. 218, pp. 1295-1303, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[14] Muhammad Abdullah, Mobeen Ahmad, and Dongil Han, "Facial Expression Recognition in Videos: an CNN-LSTM Based Model for Video Classification," *International Conference on Electronics, Information, and Communication (ICEIC)*, Spain, pp. 1-3, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[15] Divina Lawrance, and Suja Palaniswamy, "Emotion Recognition from Facial Expressions for 3D Videos Using Siamese Network," *International Conference on Communication, Control and Information Sciences (ICCISC)*, India, pp. 1-6, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[16] Mingjing Yu et al., "Facial Expression Recognition Based on a Multi-Task Global-Local Network," *Pattern Recognition Letters*, vol. 131, pp. 166-171, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[17] Ninad Mehendale, "Facial Emotion Recognition Using Convolutional Neural Networks (FERC)," *SN Applied Sciences*, vol. 2, pp. 1-8, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[18] Ke Zhang et al. "Real-Time Video Emotion Recognition Based on Reinforcement Learning and Domain Knowledge," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1034-1047, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[19] Haipeng Zeng et al., "EmoCo: Visual Analysis of Emotion Coherence in Presentation Videos," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 927-937, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[20] Juan Miguel López-Gil, and Nestor Garay-Vitoria, "Photogram Classification-Based Emotion Recognition," *IEEE Access*, vol. 9, pp. 136974-136984, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[21] Wenbo Li et al., "A Spontaneous Driver Emotion Facial Expression (DEFE) Dataset for Intelligent Vehicles: Emotions Triggered by Video-Audio Clips in Driving Scenarios," *IEEE Transactions on Affective Computing*, vol. 14, no. 1, pp. 474-760, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[22] Min Hu et al., "A Two-Stage Spatiotemporal Attention Convolution Network for Continuous Dimensional Emotion Recognition from Facial Video," *IEEE Signal Processing Letters*, vol. 28, pp. 698-702, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[23] Karnati Mohan et al., "Facial Expression Recognition Using Local Gravitational Force Descriptor-Based Deep Convolution Neural Networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1-12, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[24] Najmeh Samadiani et al., "Happy Emotion Recognition from Unconstrained Videos Using 3D Hybrid Deep Features," *IEEE Access*, vol. 9, pp. 35524-35538, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[25] Jiyoung Lee et al., "Multi-Modal Recurrent Attention Networks for Facial Expression Recognition," *IEEE Transactions on Image Processing*, vol. 29, pp. 6977-6991, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[26] Xi Zhang, Feifei Zhang, and Changsheng Xu, "Joint Expression Synthesis and Representation Learning for Facial Expression Recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1681-1695, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[27] Noushin Hajarolasvadi, Enver Bashirov, and Hasan Demirel, "Video-Based Person-Dependent and Person-Independent Facial Emotion Recognition," *Signal, Image and Video Processing*, vol. 15, pp. 1049-1056, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[28] Moises Garcia Villanueva, and Salvador Ramirez Zavala, "Deep Neural Network Architecture: Application for Facial Expression Recognition," *IEEE Latin America Transactions*, vol. 18, no. 7, pp. 1311-1319, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[29] ByungOk Han et al., "Deep Emotion Change Detection via Facial Expression Analysis," *Neurocomputing*, vol. 549, p. 1-16, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[30] Sanghyun Lee, David K. Han, and Hanseok Ko, "Multimodal Emotion Recognition Fusion Analysis Adapting BERT with Heterogeneous Feature Unification," *IEEE Access*, vol. 9, pp. 94557-94572, 2021. [CrossRef] [Google Scholar] [Publisher Link]