ISSN: 0976-9102 (ONLINE)
DOI: 10.21917/ijivp.2021.0363

CANNY EDGE DETECTION AND CONTRAST STRETCHING FOR FACIAL EXPRESSION DETECTION AND RECOGNITION USING MACHINE LEARNING

P. Dinesh Kumar and B. Rosiline Jeetha

PG and Research Department of Computer Science Engineering, Dr. N.G.P. Arts and Science College, India

Abstract

Facial expression recognition in the world is challenging due to various unconstrained conditions. Although existing facial expression classifiers have been almost perfect for analyzing constrained frontal faces, they fail to perform well on partially occluded faces common in the wild. This paper deals with Facial expression detection and recognition through the Viola-jones algorithm and HCNN using the LSTM method to avoid those challenges in recent work. For face detection, basically, utilize the face detection Viola-Jones algorithm and it recognizes the occluded face and it helps to perform the feature selection through the whale optimization algorithm, once after compression and further, it minimizes the feature vector given into the HCNN and LSTM model for efficiently identifying the facial expression. In existing work, the feature extraction stage exact finding of the corner becomes a very difficult task, to solve this issue need to use a corner detection algorithm to enhance the feature extraction (corner points) from the face image. One of the main drawbacks of WOA is that it is not good at exploring the search space. To overcome those issues, the work introduced an improved framework for facial image recognition. In which first edges are detected using the canny edge detection operator. Then improved linear contrast stretching is used for image enhancement. Then in feature extraction, the author proposes Hybrid SIFT with Double δ-LBP (Dδ-LBP) to obtain the features that are illumination and pose independent. For face detection, utilize the face detection Viola-Jones algorithm and it recognizes the occluded face and significant features are selected by using self-learning chicken swarm optimization, it minimizes the feature vector is given into the Hybrid HCNN and LSTM model for efficiently identifying the facial expression. Experimental results demonstrate the efficiency of the proposed work in terms of accuracy, precision, recall and f-measure.

Keywords:

Viola-Jones Algorithm, Face Detection, Edge Detection, Feature Matching, Artificial Bee Colony, Hybrid Convolutional Neural Network

1. INTRODUCTION

Facial expression recognition (FER) has received significant interest from computer scientists and psychologists over recent decades, as it holds promise to an abundance of applications, such as human-computer interaction, affect analysis, and mental health assessment. Although many facial expression recognition systems have been proposed and implemented, majority of them are built on images captured in controlled environment, such as CK+, MMI, and other lab-collected datasets. The controlled faces are frontal and without any occlusion [1]-[3].

The FER systems that perform perfectly on the lab-collected datasets are probable to perform poorly when recognizing human expressions under natural and un-controlled conditions. To fill the gap between the FER accuracy on the controlled faces and uncontrolled faces, researchers make efforts on collecting largescale facial expression datasets in the wild despite the usage of data from the wild, facial expression recognition is still challenging due to the existence of partially occluded faces [4] [5]. It is non-trivial to address the occlusion issue because occlusions vary in the occludes and their positions. The occlusions may be caused by hair, glasses, scarf, breathing mask, hands, arms, food, and other objects that could be placed in front of the faces in daily life. These objects may block the eye, mouth, part of the cheek, and any other part of the face [6]-[8].

The variability of occlusions cannot be fully covered by limited amounts of data and will inevitably lead the recognition accuracy to decrease. To address the issue of occlusion in existing work proposed a method that extracts patch-based 3D Gabor features to obtain salient distance features, selects the salient patches and then performs patch matching operations using SVM classifier. And proposed a geometric features extraction method applying an ASM automatic fiducial point location algorithm to facial expression image, and then 80 calculating Euclidean distance between the center of gravity of the face shape and the annotated points. Finally, they extract geometric deformation difference between features of neutral and facial expressions to analyze, before applying SVM classifier for FER. Still there is a need to improve the classifier accuracy [9] [10].

To avoid those challenges in recent work this paper deals with the Facial expression detection and recognition through Viola-jones algorithm and HCNN using LSTM method. However, in existing work, feature extraction stage exact finding of the corner becomes very difficult task, to solve this issue need to use corner detection algorithm to enhance the feature extraction (corner points) from the face image. And one of the main drawbacks of WOA is that it is not good at exploring the search space.

To overcome those issues, the present work introduced an improved framework for facial image recognition. In which first edges are detected using canny edge detection operator. Then improved linear contrast stretching is used for image enhancement. Then in feature matching, the author proposes Hybrid Scale-Invariant Feature Transform (SIFT) with double δ-LBP (Dδ-LBP) to obtain the features that are illumination and pose independent. For face detection, utilize the face detection Viola-Jones algorithm and it recognizes the occluded face and significant features are selected by using Self-learning chicken swarm optimization, it minimizes the feature vector given into the Hybrid Convolutional Neural Network (HCNN) and Long Short-Term Memory (LSTM) model for identifying the facial expression in efficient manner.

The paper is organized as follows: Section 1 analyses the significance of detection and recognition of face images. Detailed review of different face detection and recognition methods are presented in section 2. Section 3 provides the design methodology for FIR. Section 4 discusses the results of simulation. The conclusion and work intended for the future are discussed in section 5.

2. LITERATURE REVIEW

This section reviews about different methods for facial expression detection and recognition.

Kumar et al. [2018] proposed a new algorithm for automatic live FED using radial basis function support Haar Wavelet Transform is used for feature extraction and RBF-SVM for classification. Edges of the facial image are detected by genetic algorithm and fuzzy-C-means. The experimental results used CK+ database and JAFEE database for facial expression. The other database used for face detection process namely FEI, LFW-a, CMU + MIT and own database. In this algorithm, the face is detected by fdlibmex technique but improved the limitations of this algorithm using contrast enhancement. In the pre-processing stage, apply median filtering for removing noise from an image. This algorithm is compared with the previous algorithm and the proposed algorithm is better than previous algorithms.

Gogić et al. [2020] proposed an algorithm that bridges the gap between precise but slow methods and fast but less precise methods. The algorithm combines gentle boost decision trees and neural networks. The gentle boost decision trees are trained to extract highly discriminative feature vectors (Local Binary Features) for each basic facial expression around distinct facial landmark points. These sparse binary features are concatenated and used to jointly optimize facial expression recognition through a shallow neural network architecture. The joint optimization improves the recognition rates of difficult expressions such as fear and sadness. The proposed method (LBF-NN) compares favourably with state-of the-art algorithms while achieving an order of magnitude improvement in execution time.

Yang et al. [2018] presented a novel approach (so-called IA-gen) to alleviate the issue of subject variations by regenerating expressions from any input facial images. First of all, train conditional generative models to generate six prototypic facial expressions from any given query face image while keeping the identity related information unchanged. Generative Adversarial Networks are employed to train the conditional generative models, and each of them is designed to generate one of the prototypic facial expression images. Second, a regular CNN (FER-Net) is fine- tuned for expression classification. Method has been evaluated on CK+, Oulu-CASIA, BU-3DFE and BU-4DFE databases, and the results demonstrate the effectiveness of the proposed method.

Ding et al. [2017] proposed the double local binary pattern (DLBP) to detect the peak expression frame from the video. The proposed DLBP method has a much lower-dimensional size and can successfully reduce detection time. Besides, to handle the illumination variations in LBP, Logarithm-Laplace (LL) domain is further proposed to get a more robust facial feature for detection. Finally, the Taylor expansion theorem is employed in our system for the first time to extract facial expression feature. Experimental results on the JAFFE and Cohn-Kanade data sets show that the proposed TFP method outperforms some state-of-the-art LBP-based feature extraction methods for facial expression feature extraction and can be suited for real-time applications.

Kamarol et al. [2017] proposed a novel framework for facial expression recognition and intensity estimation with low computational complexity requirement. The algorithm constructs a representation of facial features based on a weighted voting scheme and employs Hidden Markov Models to classify an input video into one of the six basic expressions, namely anger, disgust, fear, happiness, sadness, and surprise. The temporal segments, neutral, onset, and apex, of an expression are then obtained by means of a change-point detector. Evaluations on subject-independent analysis was conducted using Cohn-Kanade dataset and Beihang University facial expression datasets. The proposed approach has demonstrated a superior performance in recognizing facial expressions and estimating expression intensities.

Shah et al. [2017] handled a multi-dimensional data using linear discriminant analysis (LDA) and threefold support vector machine (SVM) techniques to reduce the complexity and minimize false labelling. A facial expression application is proposed in which six natural expressions are used as multi-class data. Face image is divided into seven triangles on the basis of two focal points. A combined local and global feature descriptor is generated. Discrete Fourier transform is applied and processed with LDA to obtain discriminant features and accurately map an input feature space to an output space. To evaluate the system performance, Japanese Female Facial Expression, FER-2013 and Cohn–Kanade DFAT datasets are used. The obtained results show that multi-class data hyper plane using LDA and threefold SVM approach is effective and simple for quadratic data analysis.

Qi et al. [2018] presented a new expression recognition approach based on cognition and mapped binary patterns. At first, the approach is based on the LBP operator to extract the facial contours. Secondly, the establishment of pseudo-3-D model is used to segment the face area into six facial expression sub-regions. In this context, the sub-regions and the global facial expression images use the mapped LBP method for feature extraction, and then use two classifications which are the support vector machine and softmax with two kinds of emotion classification models the basic emotion model and the circumflex emotion model. The experimental results show that the method can effectively remove the confounding factors in the image. And the result of using the circumplex emotion model is obviously better than the traditional emotional model.

3. PROPOSED WORK

This section stages the proposed model face detection and recognition in detail. First edges of the input face images are detected through canny edge detector. Secondly Image enhancement will be done based on enhanced linear contrast stretching. Thirdly Viola Jones is used for Face Detection with occlusion. And then features from the input images are extracted utilizing Hybrid feature extraction using SIFT and Dδ-LBP. Feature selection will be done by using self-learning chicken swarm optimization. Finally face recognition using Hybrid CNN-LSTM learning method. Overall design flow of the proposed model is shown in Fig.1.

3.1 INPUT

The CK+ database is taken as an input. In this dataset contain of 593 image sets acquired from 123 subjects in the age range of 18–30 years. The database comprises of faces that show the seven emotions, such as, anger, fear, disgust, happiness, sorrow, astonishment, and neutral, for which 45, 25, 59, 69, 28, 82, and 105 images are considered, correspondingly.

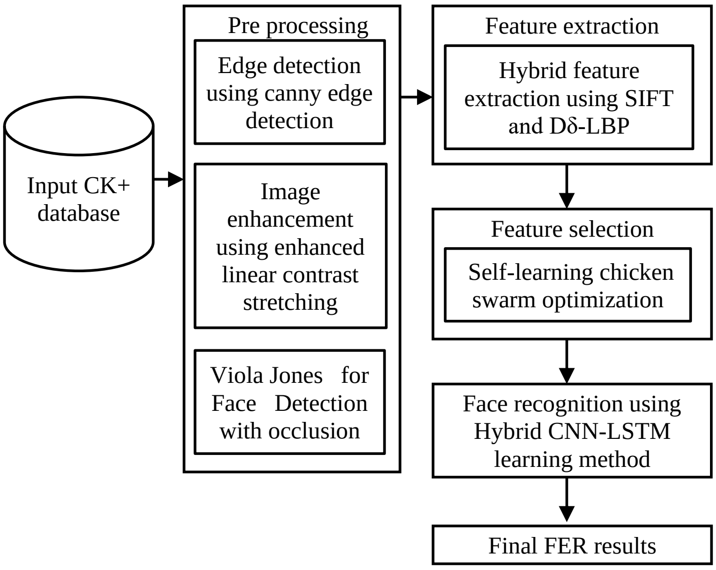

Fig.1. Overall design flow of the proposed model

3.2 EDGE DETECTION USING CANNY OPERATOR

In Feature extraction stage exact finding of the corner of input face image becomes very difficult task, this can be solved by using corner detection algorithm. This work introduces canny operator to detect the edges exactly.

Algorithm 1: Canny Edge Detection

Input: CK+ database

Output: Detected edge points

Step 1: Apply Gaussian filter to smooth the image in order to remove the noise [18,19].

Step 2: Find the intensity gradients of the image.

Step 3: Apply non-maximum suppression to get rid of spurious response to edge detection.

Step 4: Apply double threshold to determine potential edges.

Step 5: Track edge by hysteresis: Finalize the detection of edges by suppressing all the other edges that are weak and not connected to strong edges.

3.3 CONTRAST STRETCHING USING LINEAR CONTRAST STRETCH BASED ON UNSHARP MASKING (LCSUM)

After edge detection process, it is required to enhance the face image because there can be a Contrast which decreases the perceptibility and visibility of the concerned region. In this work, Linear Contrast Stretch Based on Unsharp Masking (LCSUM) technique is introduced for enhancement of the face image. Unsharp masking (UM) is basically an image manipulation approach. The Unsharp masking approach can of very much use in improving the detailed appearance using minute enhancements of edge contrast of an image [20]-[22]. Generally, Unsharp mask is utilized for sharpening an image where this will be useful in affirming the image texture and image details. The traditional unsharp masking algorithm can be expressed as Eq.(1).

$$z=n+\gamma(m-n) \tag{1}$$

where m indicates the input image, n refers to the result of the process employing a linear low-pass filter, whereas γ refers to the gain with (γ>0) that is actually the real scaling factor. Signal d = m-n frequently amplifies (γ>1) to maximize the sharpness. The signals comprise of image details, noise, over-shoot and under-shoots in a region of sharp edge resulting due to smoothening edges.

However, these traditional unsharp masks are affected with halo effect problems. To overcome the halo effect problem this work introduces an unsharp masking framework designed for face image enhancement [23].

This framework depends on generalizing the unsharp masking algorithm integrated with the functions of adaptive contrast stretching on halo effect problems resolved employing an edge preserve filter. In this research work, the concept of enhancement and sharpening makes use of another process, which uses adaptive contrast stretching algorithm and the output is known as w (y). The processing of the image details are done with $g(d) = \gamma(d)\otimes d$, where d refers to an adaptive gain control and it is a function of the amplitude of the detail signal of d. the final outcome of the algorithm is expressed as

$$u=w(y)\otimes[\gamma(d)\otimes d] \tag{2}$$

3.4 VIOLA-JONES ALGORITHM FOR FACE DETECTION WITH OCCLUSION

After contrast stretching need to detect the face images with occlusion. In this work Viola-Jones algorithm is utilized to recognize the internal essence of the square shape in the picture. At that point the complex face highlights of the square shape locale are broke down by discriminative investigation, and the hybrid face features are contribution to the classifier for achieving face acknowledgment. While building complicated features of face in a rectangular casing, we have to utilize worldwide and nearby features separated from the face present inside the rectangular frame. Viola-Jones algorithm is a fell Adaboost classifier dependent on picture fundamental. Each sub window moves, the indispensable picture is determined. The necessary picture is formed by the sum of the abovementioned and left pixel of x, y.

The three fundamental ideas which permit it to run continuously are the vital picture, Ada Boost and the cascade structure. The Integral Image is an algorithm for financially savvy age of the total of pixel forces in a predetermined square shape in a picture. It is utilized for rapid calculation of Haar-like features. Estimation of the entirety of a rectangular region inside the first picture is very proficient, requiring just four augmentations for any self-assertive square shape size. AdaBoost is utilized for development of solid classifiers as direct mix of weak classifiers. It is engaged to distinguish the appearances with impediment all the more successfully and furthermore it improves the computational unpredictability.

Viola jones makes use of Haar features, for characterizing the recognized object like face or it isn’t based on various intensity level of various part of the face. The significance of another object in front of the face hampers the detection of the acquired feature, perhaps to identify the face. More features with more data points were combined in the basic face detection method, for maximizing the efficiency. The Fig.2 represents few occlusions like sunglass on the face, mouth covered, eyes coved. Apart from these factors, the face detection, can also be affected by hairs in front of the face or else an object in front of the face.

Fig.2. Simple occluded face images

Algorithm 1: Viola-Jones Algorithm

Step 1: Input: CK+ image

Step 2: Output: state of emotion

Step 3: for i ← 1 to shift total do

Step 4: for j ← 1 to stages total do

Step 5: for k ← to filter total do

Step 6: w_e = integral of sub windows

Step 7: update weight_face w_i

Step 8: update weight_emotion

Step 9: if w_e<weight_face then

Step 10: break for k_loop

Step 11: else

Step 12: $w_{tot} = \Sigma w_e$

Step 13: end_if

Step 14: end_for

Step 15: end_for

Step 16: if w_{tot}< weight_ emotion then

Step 17: output = bored

Step 18: else

Step 19: output = interest

Step 20: end_if

Step 21: end_for

where w_e refer to the weight of sub windows and w_i indicates the weight of pixel i. The cascaded AdaBoost classifier helps to segregate the image integral for choosing the face area. Here, m filters get repeated at every stage. We can extract the weak classifier (km), out of the classifier pool. So, the weight of classifier w_m is determined as below:

$$w_m=0.5(\ln(1-e_m)-\ln(e_m)) \tag{3}$$

where e_m refers to the normalized weight which is computed using:

$$e_m=W_e/W$$

where W indicates the maximum weight. So, the threshold w_t gets updated applying:

$$w_t^{m+1}=\begin{cases} w_t^m e^{\alpha_m k_m(x_l)} & miss \\ w_t^{m_e-\alpha_m} & otherwise \end{cases} \tag{4}$$

Using the weight of the sub-windows we, distinguish the updated threshold w_t on the next subsequent iteration (m+1). If the weight of sub window is less than the threshold, the sub windows eliminate the face. If the weight of sub window is greater than the threshold, the sub window gets processed in emotion detection. Similar to face detection, the procedure of emotion recognition works, which comprises of estimating the integral image and the cascaded AdaBoost helps divide them. But, the integral image at this step is the summation of the filtered faces, expressed as below:

$$w_{tot}=\sum w_e^m \tag{5}$$

where w_{tot} refers to the integral image achieved of the filtered faces w_e^m. Here, by applying Eq.(4), the threshold gets updated on every iteration (m+1). The weight of filtered faces helps to distinguish the updated threshold. In case the weight of filtered faces is lesser than the threshold, the filtered face is set as ‘interest’ expression. In case the weight of filtered faces is greater than the threshold, the filtered faces is set as ‘bored’ expression. Here, it is considered that when distinguished with the threshold, interest expressions exhibit a higher value, since some indicators like smile increase the texture feature’s value thereby permitting a raised value.

3.5 HYBRID FEATURE EXTRACTION USING SIFT AND DOUBLE δ-LBP (Dδ-LBP) ALGORITHM

After face detection with occlusion, it is required to extract all the features from the face image. Here, with double δ-LBP (Dδ-LBP) method for feature extraction, the scale Invariant Feature Transform (SIFT) algorithm is combined. In the case of machine vision, we enforce this algorithm, for extracting particular images features for applications like matching different views of an object or scenery (for binocular vision) and finding the objects. To scale and rotation the acquired features were invariant and in lighting it exhibits partial invariance to any kind of changes and we call it as the local feature extraction technique which identifies and defines the local features present in images. The SIFT extracted features show invariance to scale of the image, orientation, variation in lighting and considerable extent of affine disturbance. For object recognition purposes, initially establish the SIFT feature extraction method.

The image information is transformed by SIFT into scale invariant directions comparative with local features. In order to change in enlightenment, the features show invariance to picture scaling and turn and are invariant halfway. From normal pictures can segregate huge feature quantities. For face acknowledgment, SIFT features are separated from a lot of training images and put away in the database. From the test picture to the current database, a test image is synchronized by separately seeking at each component. Depending on the Euclidean separation of their component vectors, the best match between the removed features works. SIFT makes use of four significant phases of computation, in order to develop a set of computation. The four phases incorporate Scale-Space Extrema Detection, Key Point Localization, Orientation Assignment and Key Point Descriptor.

3.5.1 Key Points Extraction:

Highly distinctive facial features were extracted through SIFT and further, the extraction of the key points for the complete images presents in the training dataset and also the test image is done and the result key points of entire training images were controlled by the key point dataset. The steps in the key point extraction will include DoG Image Generation, Local Key Point Detection, Accurate Key Point Location and Eliminating Edge Response.

3.5.2 Generation of Difference-of-Gaussian Images:

For key point detection, we ought to recognize the position and scale of the key point. We confirm DoG as an effective function, for the key point location detection. Further we can estimate the DoG function, from the difference of two adjacent scales.

3.5.3 Local Key Point Detection:

In the current image, each key point is distinguished with its eight neighbours and nine neighbours from nearby scales, which provide local maxima or minima.

3.5.4 Accurate Key Point Location Finding:

Those sensitive to noise will be rejected, which has low contrast from the number of key points and by fixing a threshold value for the key point that helps to achieve just the stabilized key points.

3.5.5 Elimination of Edge Response:

When the edges are incorrectly decided and do not show resilience to noise, the key points will be eliminated. The principal of curvatures, poorly defined peaks. We can eliminate the unwanted key points, through fixing the extra threshold on the principle of curvatures.

The Local Binary Pattern (LBP) is a generally utilized descriptor in facial expression detection because of its productivity and viability. Be that as it may, the available facial expression detection techniques dependent on LBP either disregard various types of data, for example, subtleties and the shape of countenances, or depend on the segmentation of facial images, for example, separating the face picture into squares or allowing the square fixating on tourist spots. Thinking about this issue, to utilize both detail and contour face data in facial expression detection, a new feature extraction strategy dependent on double δ-LBP (Dδ-LBP) right now. Right now, two δ-LBPs are utilized to speak to subtleties and the shape of appearances independently, which consider various types of data of facial expression.

3.5.6 Key points Validation:

This phase assists us to recognize the satisfactory key points. For handling face recognition, satisfactory key points form the exceptionally particular key points that are separated from the picture. In Fig.2, the definite procedure associated with the Key Point Acceptance and Matching Algorithm is provided. Three stages are engaged with the procedure of approval: Calculation of Euclidean Distances, Sorting of Euclidean Distances and Key Points Acceptance.

Calculation of Euclidean Distances: When the extraction of the key points for the test and training set are done by the formula, we can estimate the Euclidean distance:

$$ED = \sqrt{\left(x_i - x_2\right)^2 + \left(y_1 - y_2\right)^2} \tag{6}$$

where, x_1, x_2, y_1, y_2 refer to x and y coordinates of the test and training image correspondingly. Can estimate the Euclidean distances, through associating each test key point with a whole set of key points in the training image and then Euclidean distance array is formed.

- ***Sorting of Euclidean Distances***: Through the comparison of the entire key points belonging to test image and the whole set of key points belonging to training image, we can estimate the array of Euclidean distances and for recognizing the first two closet neighbours, we ought to sort it out.
- ***Key Points Acceptance***: Using the Euclidean distances array, we can estimate the Ratio of two closet neighbors, which, in turn, assist us to get rid of the fake key points. If in case this ratio is higher than 0.8, it has to be eliminated, since they were not considered as acceptable key points, else we accept it to be the valid key point (if the ratio is less than 0.8) and increase the key point counter that preserves the count of matched key points.

Here we estimate the Absolute Differences between the key points belonging to the thoroughly matched training images and the test image. More officially, I_{test} and I_{train} indicates the test and training image, correspondingly. Unique Features for the test and training image will be expressed as:

$$K_T^{I_{test}} = \left\{K_1^{I_{test}}, K_2^{I_{test}}, K_3^{I_{test}}, ..., K_N^{I_{test}}\right\} \tag{7}$$

$$K_T^{I_{train}} = \left\{K_1^{I_{train}}, K_2^{I_{train}}, K_3^{I_{train}}, ..., K_N^{I_{train}}\right\} \tag{8}$$

where, N and M represents the number of key points present in the test and training images respectively. The key points of the training images that are sorted based on the MT will be denoted as:

$$K_{ST}^{I_{strainset}} = \left\{K_{1-E_1}^{I_{strain}}, K_{1-E_2}^{I_{strain}}, K_{1-E_3}^{I_{strain}}, ..., K_{1-E_n}^{I_{strain}}\right\} \tag{9}$$

where E_1, E_2,…,E_n refer to the key points belonging to the sorted training images. Key points of one of the sorted training images will be denoted as:

$$K_{ST}^{I_{strain}} = \left\{K_1^{I_{strain}}, K_2^{I_{strain}}, K_3^{I_{strain}}, ..., K_E^{I_{strain}}\right\} \tag{10}$$

3.6 FEATURE SELECTION USING SELF-LEARNING CHICKEN SWARM OPTIMIZATION

Results of feature extraction will have more number of features it may consume more time to get classification results. To reduce the computation time, need to select the significant features.

The proposed system makes use of the chicken swarm optimization (CSO) algorithm to find combinations of features that maximizes the classification accuracy with minor number of selected features. Chicken swarm optimization (CSO) is bio-inspire metaheuristic optimization algorithm. The algorithm mimics the hierarchal order of a chicken swarm and the behaviors of its individual chickens. The hierarchal order of a chicken swarm is divided into several groups, each group consists of one rooster and many hens and chicks. Each type of chickens follows different laws of motions. A hierarchal order plays a significant role in the social lives of chickens. The superior chickens in a flock will dominate the weak ones. There exist the more dominant hens that remain near to the head roosters as well as the more submissive hens and roosters who stand at the periphery of the group [24]-[26].

The mathematical model of CSO proposed in was based on the following rules that summarize the chickens' behaviours:

Step 1: The chicken swarm is divided into several groups. In each group there is a dominant rooster, following it some hens and chicks.

Step 2: The fitness value of the chickens outlines the hierarchy of the swarm, the individuals with the best fitness will be the roosters each one will be a group leader, the individuals with the worst fitness values will be considered as chicks. The others would be the hens.

Step 3: The swarm hierarchy, dominance relationship and mother-child relationship in a group will remain unchanged. These statuses only update every several (G) time steps.

Step 4: The swarm consists of N virtual chickens divided as follow: RN, HN, CN, and MN which are the number of roosters, the hens, the chicks, and the mother hens, respectively. Each individual is represented by their positions in a D-dimensional space by

$$x_{i,j}\ (i \in [i,\ldots N],\ j \in [1,\ldots,D]) \tag{11}$$

3.6.1 Rooster Movement;

Roosters with better fitness values can search for food in a wider range of place than those with worse fitness values, such movement is depicted as in Eq.(1) and Eq.(2).

$$x_{i,j}^{t+1} = x_{i,j}^{t} * \left(1 + Randn\left(0, \sigma^2\right)\right) \tag{12}$$

$$\sigma^2 = \begin{cases} 1 & if\ \ f_i \le f_k \\ \exp\left(\dfrac{f_k - f_i}{|f_i| + \varepsilon}\right) & otherwise \\ & k \in [1, N], k \ne i \end{cases} \tag{13}$$

where $x_{i;j}$ is the selected rooster with index i, Randn(0,σ^2) is a Gaussian distribution with mean 0 and standard deviation $\sigma^2 \in s$ the smallest constant in the computer used to avoid zero-division-error, k a randomly chosen roosters index selected from the roosters group, f_i is the fitness value of the corresponding rooster x_i.

3.6.2 Hen movement:

Hens follow their group-mate roosters to search for food. Moreover, they would also randomly steal the good food found by other chickens, though they would be repressed by the other chickens. The more dominant hens would have advantage in competing for food than the more submissive ones. These phenomena can be formulated mathematically as in Eq.(14) and Eq.(15).

$$x_{i,j}^{t+1} = x_{i,j}^{t} + S_1 * Rand * \left(x_{r_{1},j}^{t} - x_{i,j}^{t}\right) + S_2 * Rand * \left(x_{r_{2},j}^{t} - x_{i,j}^{t}\right) \tag{14}$$

$$S_1 = \frac{\exp\left(f_i - f_{r1}\right)}{abs\left(f_i\right) + \varepsilon} \tag{15}$$

$$S_1 = \exp\left(f_{r2} - f_i\right) \tag{16}$$

where *Rand* is a uniform random number over [0, 1]. $r_1 \in [1,\ldots,N]$ is an index of the rooster, which is the i^{th} hen's group-mate, while $r_2 \in [1,\ldots,N]$, is randomly chosen index of a chicken (rooster or hen) from the swarm.

3.6.3 Chick Movement:

The chicks move around their mother to search for food. This is formulated as in Eq.(17).

$$x_{i,j}^{t+1} = x_{i,j}^{t} + FL * \left(x_{m,j}^{t} - x_{i,j}^{t}\right) \tag{17}$$

where $x_{m,j}^{t}$ is the position of the i^{th} chick's mother such that $m \in [1;N]$, FL is parameter that represent how much speed a chick would follow its mother, to consider the differences between each chick FL is chosen randomly in the range [0,2].

The feature space with each feature represented in an individual dimension and the span of each dimension ranges from 0 to 1 is very huge and hence requires an intelligent searching method to find optimal point in the search space that maximizes the given fitness function. The fitness function for the CSO is to maximize classification performance over the validation set given the training data, as shown in Eq.(18) while keeping minimum number of features selected.

$$f_\theta = \omega * E + (1 - \omega)\frac{\sum_i \theta_i}{N} \tag{18}$$

where f_θ is the fitness function given a vector θ sized N with 0/1 elements representing unselected/selected features, N is the total number of features in the dataset, E is the classifier error rate and ω is a constant controlling the importance of classification performance to the number of features selected.

The used variables are the same as the number of features in the given dataset. All variable is limited in the range [0, 1], where the variable value approaches to 1; its corresponding feature is candidate to be selected in classification. In individual fitness calculation, the variable is threshold to decide the exact features to be evaluated as in the Eq.(19).

$$f(x) = \begin{cases} 1 & if\ X_{ij} > 0.5 \\ 0 & if\ X_{ij} \le 0.5 \end{cases} \tag{19}$$

where X_{ij} is the dimension value for search agent i at dimension j. While updating the firefly position; solution, at some dimensions the updated value can violate the limiting constrains; [0, 1], and hence used simple truncation rule to ensure variable limits.

Chicken swarm optimization has the problems of low convergence accuracy and premature convergence in the high-dimensional optimization. To overcome these issues this work propose the Improved Chicken Swarm Optimization Algorithm (ICSOA) that modifies the position update equation by introducing a self-learning coefficient and partial learning from the group-mate rooster, and ICSOA has shown to be effective in avoiding low convergence accuracy and premature convergence.

The improved position update equation of chicks is as follows:

$$x_{i,j}(t+1) = \omega . x_{i,j}(t) + F.(x_{m,j}(t) - x_{i,j}(t)) + C.(x_{r,j}(t) - x_{i,j}(t)) \tag{20}$$

where r is the index of the rooster that is in the same subgroup with the i^{th} chick; ω is the self-learning coefficient; and C is the learning factor that represents the degree to which the chicks learn from the rooster in the same subgroup.

Input: Extracted features

Output: Optimal features

Step 1: Initialize RN, HN, CN, MN, G;

Step 2: Randomly initialize each chicken in the swarm

Step 3: X_i (i = 1,2,…,N);
Step 4: Initialize the max numbers of iteration T_{max};
Step 5: while $T < T_{max}$ do for each iteration
Step 6: if T%G equals 0 then
Step 7: Rank the chicken fitness values and establish a
Step 8: hierarchal order in the swarm;
Step 9: Divide the swarm into different groups, and
Step 10: determine the relationship be- tween the chicks
Step 11: and mother hens in a group;
Step 12: end
Step 13: for each chicken X_i in the swarm do
Step 14: if X_i is a roster then
Step 15: Update X_i location using Eq.(13);
Step 16: end
Step 17: if X_i is a hen then
Step 18: Update X_i location using Eq.(16);
Step 19: end
Step 20: if X_i is a chick then
Step 21: Update X_i location using Eq.(19);
Step 22: end
Step 23: Evaluate the new solution using Eq.(20);
Step 24: If the new solution is better than its previous
Step 25: one, update it;
Step 26: end
Step 27: end

3.7 FACE RECOGNITION BY HYBRID CNN-LSTM LEARNING METHOD

After selecting the significant features must classify those features to detect and recognition the face. In this research work, Hybrid Convolutional Neural Networks (CNN)-Long Short-Term Memory (LSTM) method is proposed for FER respectively. Image is the input for the system; then, it utilizes HCNN-LSTM for forecasting the facial expression label that has to be one of these labels, such as anger, happiness, sorrow, fear disgust and neutral. The distinguished parts of the face are edited and separated and afterward utilized in the form of the contribution to the CNN's origin layer. The Training stage includes the feature extraction and classification through convolution neural system. It is quite normal that utilizing the facial segments rather than the entire face's picture to be the contribution for the origin layer will decrease training time and increment the likelihood of elevated degree feature extraction along these lines expanding framework exactness.

CNNs utilize the ideas of open field and sharing of weight. Using these ideas, the quantity of trainable parameters gets diminished and the spread of data throughout the layers can be determined through convolution. In this, a convolution of sign and a filter map is carried out, having the common loads to create a feature map.

Here, HCNN is introduced to arrange the facial appearances for the given CK+ dataset. Fundamentally CNN consists of an input and an output layer, fair as numerous hidden layers. The following were the different layers of CNN, which are the hidden layer: Convolutional layers, pooling layers and completely connected layers. The convolutional layer carries out the convolution activity of the input and the resultant convoluted output is moved to the next subsequent layer. This will follow the reaction of every single neuron to visual upgrades. Convolution systems involve neighbourhood or worldwide pooling layers that merges the output of neuron groups present in a single layer into a solitary neuron in the next subsequent layer. In the previous layer, mean pooling utilizes the normal incentive from each group of neurons. Every neuron in one layer to each neuron in the next layer were linked by methods for having layers that are completely connected. The essential guideline of CNN is same as customary multi-layer perceptron neural system. The newly introduced HCNN involves input layer, convolutional layer, sub-sampling layer and classification layer. This proposed technique has evident points of interest for breaking down high-dimensional information, which help as a parameter sharing plan through which the quantity of parameters was diminished and then constrained by the convolutional layers.

Input layer gets facial expressions features from training tests and changes the information into a brought together structure so as to convey the information into next layer effectively. The underlying parameters, for example, the size of the neighbourhood receptive fields and different filters are characterized in this layer.

The facial expressions input data was convoluted by Convolution layer (Cx) through convolution algorithm and from the previous layer we created a several layers called feature map comprising convolution algorithm results. Extracting the key features was the main purpose here and further it reduces the computational complexity of the network. The convolution is given by the output equation given as below:

$$x_i^l = f\left(\sum_{i\in m_j} x_i^{l-1} * k_{ij}^l + b_i^l\right) \tag{21}$$

$$f(x) = \frac{1}{1+e^{-x}} \tag{22}$$

where x refers to the output value of the convolution layer, k indicates the kernel (or known as the filter), l stands for the quantity of output layer which is chosen by the quantity of parts, indicates the stride that the kernel shifts in each progression of computation, m_j stands for the j^{th} feature map created by various kernel, b indicates the bias and f is an activation work generally characterized in the form of a sigmoid capacity appeared in Eq.(18). For neuron of a similar feature map, same weights sharing and bias is accomplished and then albeit each output neuron has distinctive responsive fields. Here, training parameters are extraordinarily diminished.

After each convolutional layer, we make use of the enactment work. Initiation work is one which helps to delineate output to a lot of data sources and it creates the system structure to exhibit non-directivity. The underlying connection weights are initialized to the whole feature esteems given. Another input design is applied next and the output is registered as:

$$y(n) = f\left(\sum_{i=1}^{i=N} w_i(n)\, x_i(n)\right) \tag{23}$$

where,

$$f(x) = \begin{cases} +1 & if\ x \geq 0 \\ -1 & if\ x < 0 \end{cases} \quad (24)$$

where n refers to the iteration index

Connection weights are updated in accordance with

$$w_i(n+1) = w_i(n)+\eta(d(n)\text{-}y(n)x_i(n),\ i=1,2,..N \quad (25)$$

where η stands for the gain factor

Then standard deviation is applied

$$\sigma = \sqrt{\frac{1}{n} f_i (x_i - \bar{x})^2} \quad (26)$$

In the proposed HCNN network, these weighted features were provided and it acquires more accurate churn classification results. Here sub-sampled each of the feature map from the convolution layer in the previous stage. The subsample technique is a weighted summation computation or using the maximum value in an area of size $n \times n$ of each feature map. The output of sub-sample layer is as given.

$$x_i^l = f\left(b_i^l downsample\left(x_i^{l-1}\right)\right) + b_i^l \quad (27)$$

where x^l refers to the output estimation of the i^{th} sub-sample layer, down sample stands for the subsample function, f and b stand for the actuation work and the inclination separately. With the help of the sub-sample layer and over-fitting, the quantity of training parameters, filter noises are relieved and it is maintained by a strategic distance from in the system. The LSTM is joined with convolution layer of CNN to enhance the FER execution.

LSTM repetitive neural system is a unique and significant design of RNN that exceeds expectations at recalling esteems for either long or brief spans of time. LSTM has another structure called a memory cell. It comprises of four fundamental components: an input gate, a forget gate, an output gate and a neuron unit, a focal direct unit having a constant self- repetitive connection. The memory cell settles on the choices about on data is to be stored, and when to permit perusing, composing and overlooking, through the three gateways, which open and close. These three doors are actualized utilizing the activation capacity to Fig.an incentive somewhere in the range of zero and one. The input gate controls the degree to which another worth streams into the memory cell. The output gate regulates the degree to which esteem in memory cell is utilized to process the result an output. The overlook gate regulates the degree to which the worth stays in the memory cell.

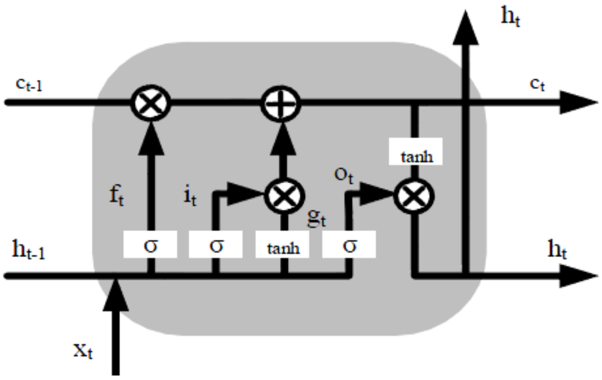

Fig.3. The design of LSTM

3.8 CLASSIFICATION LAYER

After the facial appearances information experiences a few convolution layers and sub-sample layers, the size of the output feature maps ceaselessly diminishes. As to the classification layer, each feature map comprises of just a single neuron and turns into a feature vector. The vector and a classifier are completely associated. The time window filled in as a parameter, going from 0.5s to 10s (0.5, 1, 2, 5, and 10s). At the point when the time window size will be ≤1 s, telecom information is fragmented without overlaps; in any case, information is sectioned every second with covers with past portions, e.g., a 9s overlaps for a 10s segment. Thus, the expense of multifaceted nature is decreased right now.

In order to train CNN models, all the sections completely inside the facial appearance that states are utilized to show facial appearances class, while the portions 8 times longer than the facial appearances time frame are haphazardly chosen from inter-ictal states, characterized dependent on the partition from the outward appearances state by over 60 minutes, to demonstrate typical articulations class during CNN learning.

The trained CNN classified each portion of facial appearances information, i.e., every 0.5 second for the 0.5-second section and consistently for the remaining the fragments, as sad, happy, anger, fear, neutral or surprise. To evaluate the presentation of the CNN classification, the facial appearances names by CNN were thought about. It is assessed the order execution either in leave-one-out testing or pair wise testing. In the testing stage, a model get strained with CK+ database and then tested with CK+ information from the final subject. In the pair wise testing, a model is developed from a solitary subject's information and tested with CK+ database from each subject exclusively.

Algorithm 4: HCNN-LSTM

Step 1: Begin

Step 2: For all input feature ∈ CK+ dataset do

Step 3: Transform the input into sub layers

Step 4: Add weight factor using Eq.(23)-Eq.(25)

Step 5: Compute Standard deviation value using Eq.(26)

Step 6: Detect customer features using Eq.(21) and Eq.(22)

Step 7: Do hidden layer process

Step 8: Perform convolution process

Step 9: Perform LSTM process

Step 10: Extract more informative features using HCNN using Eq.(27)

Step 11: Carry out training and testing process for CK+ dataset provided

Step 12: Replicate the predefined facial expression information label for every feature according to the input dataset

Step 13: Classify FER results with more accuracy

Step 14: End

4. RESULTS AND DISCUSSION

The existing methods are considered as HCNN-LSTM, EBOSVM, Nearest Neighbour Classifier (NNC) and Class Variation Reduction (CVR) method to evaluate CK+ databases alongside proposed HCNN-LSTM-CSO algorithm. The performance metrics are precision, recall, f-measure and accuracy metrics.

4.1 PRECISION

The Fig.4 shown that the performance comparison results for the proposed HCNN-LSTM-CSO and already existing CVR, NNC, EBOSVM and HCNN-LSTM. Along the x-axis, the strategies are plotted and the precision value is plotted along the y-axis.

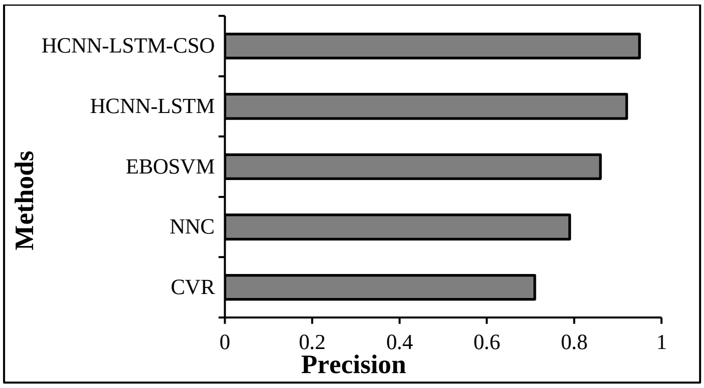

Fig.4. Precision

By using contrast stretching and canny edge detection precision of the face recognition result improved. The current strategies are, CVR, NNC, EBOSVM and HCNN-LSTM algorithms give lower precision which is 0.71, 0.79, 0.86, 0.92 while proposed HCNN-LSTM-CSO algorithm gives higher precision which is 0.95 for the given facial expression databases. In this way the outcome infers that the proposed HCNN-LSTM-CSO builds the facial expression acknowledgments are, for example, dismal, upbeat, outrage, dread, nonpartisan and shock articulations precisely for the given databases.

4.2 RECALL

From the Fig.5, it is seen that the examination metric is assessed utilizing the available and novel strategy as far as recall. Along the x-axis, the strategies are plotted and the recall value is plotted along the y-axis.

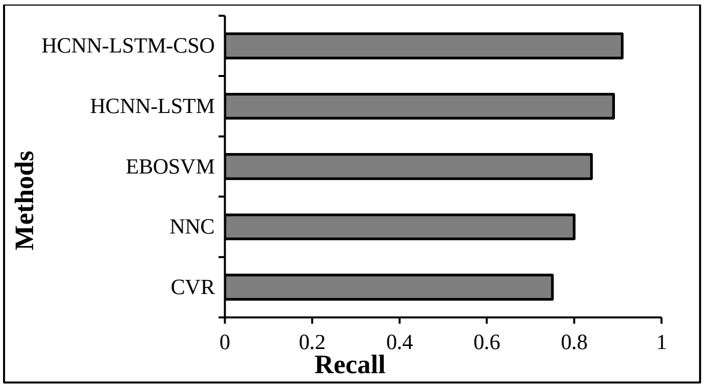

Fig.5. Recall

The current strategies are CVR, NNC, EBOSVM and HCNN-LSTM algorithm gives lower recall results which is 0.75, 0.8, 0.84 and 0.89 while the proposed HCNN-LSTM-CSO algorithm gives higher recall results which is 0.91 to the given facial appearance databases. Along these lines the outcome presumes that the proposed HCNN-LSTM-CSO expands the facial appearance acknowledgments are, for example, pitiful, upbeat, outrage, dread, impartial and shock articulations precisely for the given databases.

4.3 F-MEASURE

F-measure metric comparison is shown in the Fig.6 for the classifiers like CVR, NNC, EBOSVM, HCNN-LSTM and HCNN-LSTM-CSO. The techniques are plotted along the x-axis and the F-measure value is plotted along the y-axis.

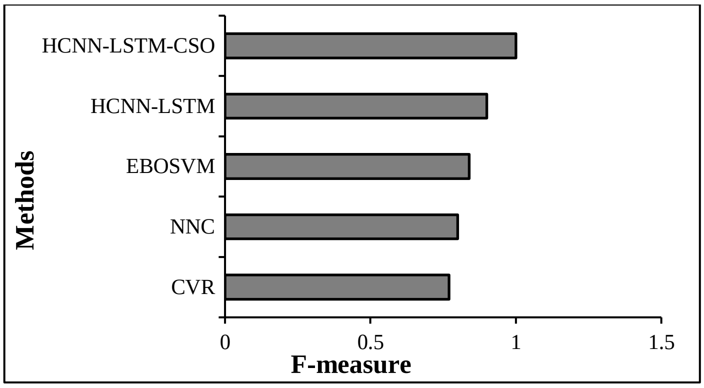

Fig.6. F-measure

The current techniques are CVR, NNC, EBOSVM and HCNN-LSTM algorithms gives lower F-measure results which is 0.77, 0.8, 0.84 and 0.9 while the proposed HCNN-LSTM-CSO gives higher F-measure to the given facial appearance databases. In this way the outcome presumes that the proposed HCNN-LSTM-CSO builds the outward appearance acknowledgments are, for example, tragic, glad, outrage, dread, impartial and shock articulations precisely for the given databases.

4.4 ACCURACY

From the Fig.7, it is seen that the examination metric is assessed utilizing the available and the newly introduced technique as far as accuracy. The techniques are plotted along the x-axis and the accuracy value is plotted along the y-axis. Based on the fitness function of the proposed ICSO reduces the accuracy.

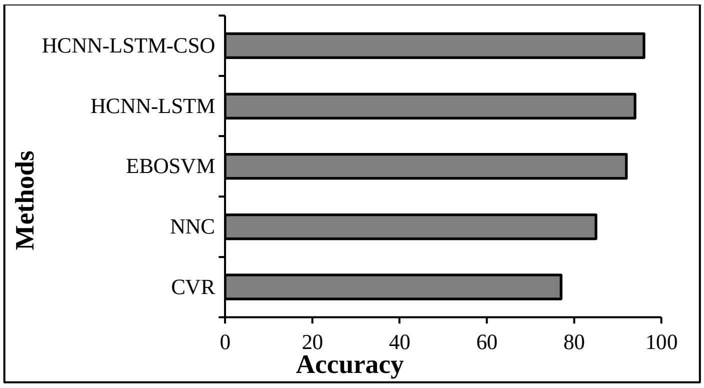

Fig.7. Accuracy

The current strategies are, for example, CVR, NNC, EBOSVM and HCNN-LSTM algorithms gives lower accuracy results which is 77%, 85%, 92% and 94% while the proposed HCNN-LSTM-CSO gives higher accuracy results is 96% to the given facial expression databases. Accordingly, the outcome infers that the proposed HCNN-LSTM-CSO builds the facial appearance acknowledgments are, for example, miserable, glad, outrage, dread, impartial and shock articulations precisely for the given databases.

5. CONCLUSION AND FUTURE WORK

Facial expression is central to human experience, but most previous databases and studies are limited to pose facial behaviour under controlled conditions. In this work introduce an improved framework for facial expression recognition and detection. In which edge detection is performed based on canny edge detection operator. Contrast stretching is used enhance the input face image. For face detection, utilize the Viola-Jones algorithm and it recognizes the occluded face then Features are extracted using Hybrid Scale-Invariant Feature Transform (SIFT) with double δ-LBP (Dδ-LBP) to obtain the features that are illumination and pose independent. Significant features are selected by using self-learning chicken swarm and finally face recognition using Hybrid Convolutional Neural Network (HCNN) and Long Short-Term Memory (LSTM) model for identifying the facial expression in efficient manner. Experimental results show that the proposed model provides better results in terms of precision, recall and f-measure. However deep learning has more computation complexities so need to use other machine learning in future.

REFERENCES

[1] N. Zeng, H. Zhang and A.M. Dobaie, "Facial Expression Recognition Via Learning Deep Sparse Auto Encoders", *Neuro Computing*, Vol. 273, pp. 643-649, 2018.

[2] N.B. Kar, K.S. Babu and S.K. Jena, "Face Expression Recognition using Histograms of Oriented Gradients with Reduced Features", *Proceedings of International Conference on Computer Vision and Image Processing*, pp. 209-219, 2017.

[3] S. Li and W. Deng, "Reliable Crowd Sourcing and Deep Locality-Preserving Learning for Unconstrained Facial Expression Recognition", *IEEE Transactions on Image Processing*, Vol. 28, No. 1, pp. 356-370, 2018.

[4] K. Zhang, Y. Huang, Y. Du and L. Wang, "Facial Expression Recognition based on Deep Evolutional Spatial-Temporal Networks", *IEEE Transactions on Image Processing*, Vol. 26, No. 9, pp. 4193-4203, 2017.

[5] K. Shan, J. Guo and R. Bie, "Automatic Facial Expression Recognition based on A Deep Convolutional-Neural-Network Structure", *Proceedings of International Conference on Software Engineering Research, Management and Applications*, pp. 123-128, 2017.

[6] K. Wang, X. Peng, J. Yang and Y. Qiao, "Region Attention Networks for Pose and Occlusion Robust Facial Expression Recognition", *IEEE Transactions on Image Processing*, Vol. 29, pp. 4057-4069, 2020.

[7] S. Nigam, R. Singh and A.K. Misra, "Efficient Facial Expression Recognition using Histogram of Oriented Gradients in Wavelet Domain", *Multimedia Tools and Applications*, Vol. 77, No. 21, pp. 28725-28747, 2018.

[8] Y.H. Lai and S.H. Lai, "Emotion-Preserving Representation Learning Via Generative Adversarial Network for Multi-View Facial Expression Recognition", *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 263-270, 2018.

[9] J.C. Batista, V. Albiero and L. Silva, "Aumpnet: Simultaneous Action Units Detection and Intensity Estimation on Multipose Facial Images using a Single Convolutional Neural Network", *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 866-871, 2017.

[10] E. Sariyanidi, H. Gunes and A. Cavallaro, "Learning bases of Activity for Facial Expression Recognition", *IEEE Transactions on Image Processing*, Vol. 26, No. 4, pp. 1965-1978, 2017.

[11] S. Kumar, S. Singh and J. Kumar, "Automatic Live Facial Expression Detection using Genetic Algorithm with HAAR Wavelet Features and SVM", *Wireless Personal Communications*, Vol. 103, No. 3, pp. 2435-2453, 2018.

[12] I. Gogic and M. Manhart, "Fast Facial Expression Recognition using Local Binary Features and Shallow Neural Networks", *The Visual Computer*, Vol. 36, No. 1, pp. 97-112, 2020.

[13] H. Yang, Z. Zhang and L. Yin, "Identity-Adaptive Facial Expression Recognition through Expression Regeneration using Conditional Generative Adversarial Networks", *Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 294-301, 2018.

[14] Y. Ding, Q. Zhao, B. Li and X. Yuan, "Facial Expression Recognition from Image Sequence based on LBP and Taylor Expansion", *IEEE Access*, Vol. 5, pp. 19409-19419, 2017.

[15] S.K.A. Kamarol and R. Parthiban, "Joint Facial Expression Recognition and Intensity Estimation based on Weighted Votes of Image Sequences", *Pattern Recognition Letters*, Vol. 92, pp. 25-32, 2017.

[16] J.H. Shah, M. Sharif, M. Yasmin and S.L. Fernandes, "Facial Expressions Classification and False Label Reduction using LDA and Threefold SVM", *Pattern Recognition Letters*, Vol. 139, pp. 166-173, 2020.

[17] Qi, C., Li, M., Wang, Q., Zhang, H., Xing, J., Gao, Z. and Zhang, H., 2018. Facial expressions recognition based on cognition and mapped binary patterns. IEEE Access, 6, pp.18795-18803.

[18] G. Shi, J. Suo and C. Liu, "Moving Target Detection Algorithm in Image Sequences based on Edge Detection and Frame Difference", *Proceedings of International Conference on Information Technology and Mechatronics Engineering*, pp. 740-744, 2017.

[19] K. Gaurav and U. Ghanekar, "Image Steganography based on Canny Edge Detection, Dilation Operator and Hybrid Coding", *Journal of Information Security and Applications*, Vol. 41, pp. 41-51, 2018.

[20] G. Deng and Y. Wu, "Double Lane Line Edge Detection Method based on Constraint Conditions Hough Transform", *Proceedings of International on Symposium on Distributed Computing and Applications for Business Engineering and Science*, pp. 107-110, 2018.

[21] S. Bhairannawar, A. Patil, A. Janmane and M. Huilgol, "Color Image Enhancement using Laplacian Filter and Contrast Limited Adaptive Histogram Equalization", *Proceedings of International on Innovations in Power and Advanced Computing Technologies*, pp. 1-5, 2017.

[22] Z. Shi, Y. Feng and L. He, "Let You See in Sand Dust Weather: A Method based on Halo-Reduced Dark Channel Prior Dehazing for Sand-Dust Image Enhancement", IEEE Access, Vol. 7, pp. 116722-116733, 2019.

[23] A. Mustapha, A. Oulefki, M. Bengherabi and M.A. Algaet, "Towards Nonuniform Illumination Face Enhancement Via Adaptive Contrast Stretching", *Multimedia Tools and Applications*, Vol. 76, No. 21, pp. 21961-21999, 2017.

[24] S. Deb, X.Z. Gao, K. Tammi and P. Mahanta, "Recent Studies on Chicken Swarm Optimization Algorithm: A Review", *Artificial Intelligence Review*, Vol. 53, pp. 1-29, 2019.

[25] J. Wang, Z. Cheng and Y. Bi, "Improvement and Application of Chicken Swarm Optimization for Constrained Optimization", *IEEE Access*, Vol. 7, pp. 58053-58072, 2019.