Tech Science Press

Face Recognition System Using Deep Belief Network and Particle Swarm Optimization

K. Babu[1,*], C. Kumar[2] and C. Kannaiyaraju[3]

[1]Department of Computer Science and Engineering, Sri Rangapoopathi College of Engineering, Gingee, 604151, India
[2]Department of Electrical and Electronics Engineering, M. Kumarasamy College of Engineering, Karur, 639113, India
[3]Department of Computer Science, Ambo University of Technology, Ambo, 00251, Ethiopia
*Corresponding Author: K. Babu. Email: babukannan5@gmail.com

Abstract: Facial expression for different emotional feelings makes it interesting for researchers to develop recognition techniques. Facial expression is the outcome of emotions they feel, behavioral acts, and the physiological condition of one's mind. In the world of computer visions and algorithms, precise facial recognition is tough. In predicting the expression of a face, machine learning/artificial intelligence plays a significant role. The deep learning techniques are widely used in more challenging real-world problems which are highly encouraged in facial emotional analysis. In this article, we use three phases for facial expression recognition techniques. The principal component analysis-based dimensionality reduction techniques are used with Eigen face value for edge detection. Then the feature extraction is performed using swarm intelligence-based grey wolf with particle swarm optimization techniques. The neural network is highly used in deep learning techniques for classification. Here we use a deep belief network (DBN) for classifying the recognized image. The proposed method's results are assessed using the most comprehensive facial expression datasets, including RAF-DB, AffecteNet, and Cohn-Kanade (CK+). This developed approach improves existing methods with the maximum accuracy of 94.82%, 95.34%, 98.82%, and 97.82% on the test RAF-DB, AFfectNet, CK+, and FED-RO datasets respectively.

Keywords: Deep learning; facial expression recognition; biometrics; PCA; GWO; PSO; deep belief network

1 Introduction

The facial expression provides details about the current state of the people speaking with us. They may be sad, happy, surprised while communicating with others. Feelings act on the face as expressions and communicate as non-verbal actions with state of mind, intentions, and emotions. The facial expression makes conversation with feelings to listeners without saying more words on emotions. If we sadly convey the happy news, our expression of sadness in the face with happy words makes interpreting power on face for better action. It confuses the people too. Sometimes facial expression leads to a

solution for big problems also makes problem in simple misunderstanding. The process of detecting the expression automatically using the pigmentation changes in the facial images is challenging in many real-time applications. Facial expression is easy nonverbal communication with good understanding.

Recently a medical application of psychologists, robotic applications with computer vision receives salient interest towards facial expression recognition. Real-world applications like a computer to human communication, mental health analysis, security keys make expression recognition an abundant research field [1]. For facial recognition, we apply swarm intelligence and deep learning approaches in our proposed study. The following is the research article's contribution:

- The feature extraction is a significant step in processing the input image. Enhanced PCA with Eigen function is proposed for extracting the exact pixels.
- The extracted feature is again processed with a new local binary pattern network with PSO is used in the feature extraction process.
- Deep learning techniques such as deep belief networks are used to classify the data. Face expression is more easily recognized than other methods.

The entire article is scheduled as follows: Section 2 explains a research study on FER. Section 3 explains about proposed techniques and working principles. Section 4 discusses how to put the recommended methodology into practice and how to assess the results. Section 5 brings the research to a close and suggests ways to improve it in the future.

2 Literature Review

The researcher is interested in pattern recognition and machine learning-based recognition systems. The deep computational intelligent-based facial recognition system using cloud and fog computing and a deep learning method called deep convolution neural networks (DCNN). For recognition, features are retrieved from the input facial image using DCNN. The Decision tree, Support vector machine, and KNN ML algorithms were used to test the FR system. For the evaluation, they employed three datasets. SDUMLAHMT, 113, and CASIA are the datasets. The proposed deep facial recognition algorithm exceeds prior algorithms on assessment criteria such as accuracy, precision, sensitivity, and specificity, as well as time, with an accuracy of 99.06 percent, recall of 99.07 percent, and specificity of 99.10 percent. Traditional FR systems include drawbacks such as model structure inheritance and neural network structure constraint. These drawbacks can be overcome using the provided strategy. Based on the key characteristics of the input data, the suggested model is trained and validated using CNN. After constructing the necessary features of the input records, the proposed model is refined using CNN and evaluated with benchmark datasets for facial expression recognition such as Affect Net [2], FER Dataset [3], and Cohn-Kanade dataset [4]. Prakash et al. [5] suggested a transfer learning-based CNN model for facial recognition. The weight of the VGG 16 model was utilised to train the suggested model. The Softmax activation function was used to feed the features acquired from the CNN into the corresponding layer for classification. For classification, they used the Image Net database. The proposed approach was tested with Yale and AT&T and found to be 96.5 percent accurate in recognizing people.

A comparison of deep learning and artificial intelligence methodologies such as ANN, SVM, and deep learning was proposed by Jonathan et al. [6]. For video-based facial recognition, Ding et al. [7] proposed using ensemble CNN. CNN's standard has been improved. YouTube, PaSC faces, and COX faces were used as video face databases. Shepley [8] examined and evaluated the limitations of deep learning-based facial recognition. With unique methodologies, this study examined and analyzed the obstacles and research areas for the future. Face detection and publicly available recognition datasets were placed on the table.

There are two ways for detecting faces with DCNN: region-based and sliding window-based methods. Region-based approaches employ selective search to create a collection of locations that include a face [9]. Sun et al., a business. [10] Feature sequence, hard negative mining, and multi-scale programming were used to improve the R-CNN, resulting in lower false positive rates and higher accuracy.

A complete review of deep learning-based facial recognition algorithms, databases, applications, and protocols was proposed by Wang et al. [11].

Deep facial recognition loss functions and architectures are discussed. There are two types of face processing techniques: one-to-many augmentation and many-to-one normalization. They looked at the issues and future directions for cross factor, multiple media, heterogeneous, and industrial scenes. To overcome the multi-view face detection challenge, Far et al. [12] employed a simple deep dark face detector (DFD) algorithm. This method does not require landmark annotation, and the sampling and augmentation limitations were insufficient. Faceness is a method introduced by Yang et al. [13] to improve face detection when the face images contain 50% occlusion. Through the properties included in deep networks, it also accepts photographs in various positions and scales. Generic objects and part-level binary properties were used to train this network. They selected the amount of features by using AdaBoost-based learning techniques. Cascaded classifiers were utilised to remove the background regions and only consider the object-oriented regions in this study. In comparison to other methods, the suggested methodology achieves a high detection rate of 15 frames per second.

In the course to fine technique, it used three phases of DCNN to detect the face and position. This will increase the accuracy of automatic face detection over manual sample selection. They looked at datasets like FDDB, WIDER FACE, and AFLW to see how they may increase real-time performance accuracy. Sharma et al. [14] proposed a video identification face detection model based on a Deep Belief Network (DBN)-based deep learning algorithm. This model can distinguish between blurry photographs and photos that have been side-posed. This technique has the drawback of being unable to recognize eyes that are wearing glasses. Mehta et al. [15] improved multi-view face detection (FD) by combining CNN developed by Far fade et al. [12] with a tagging approach. They employed DCNN and Deep Dense Face Detector, and LBPH was used to recognize the faces that were discovered (Local Binary Patterns Histograms). The input image is preprocessed, heat maps are constructed to extract the faces, probability of the faces is found, and then the image is handed on to the FD tagging method. The algorithm's parameters were determined based on evaluation factors such as accuracy, recollection, and the F measure, and they attained an accuracy of 85 percent for FD. Though several methods are available in the literature, there is still a need to further improve the face detection outcome.

3 Proposed Methodology

Face detection, dimensionality reduction, feature extraction, and classification are the three steps of the described face detection system. The system's face detection phase is utilized to locate the human face it has acquired. To make the system more resilient, this phase is utilized to locate the facial images that are lighted with changes utilizing the preprocessing approach. This also includes resizing of images, cropping, adding noise, and normalization with edge detection using histogram of oriented gradients approach [16]. The second phase includes dimensionality reduction to reduce the size to improve the feature extraction and classification process. For dimensionality reduction proposed Eigen-PCA has been used. This is based on the general PCA with the Eigen face concept [17]. For feature extraction, proposed deep learning with optimization algorithm called local binary pattern network enhanced with PSO [18]. The third phase is classification, for the deep learning algorithm called deep belief network is used. The overall working process of proposed model is shown in Fig. 1.
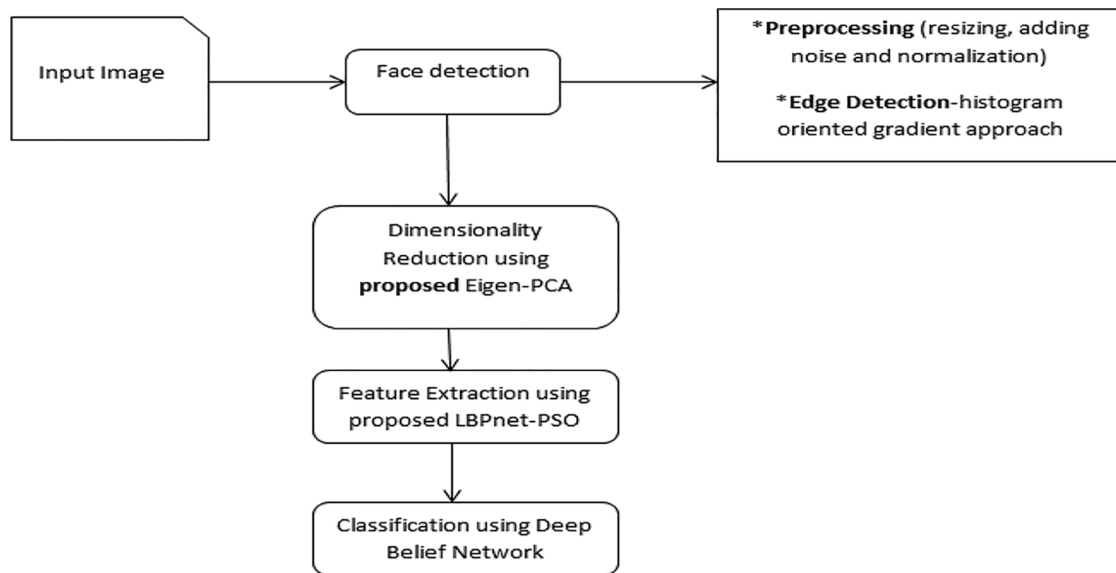
Figure 1: The proposed architecture of deep face recognition system

3.1 Face Recognition Phase

This phase is used to locate the human face that the machine has detected. Preprocessing such as resizing the image, adding noise, normalization, and edge detection of the image of implemented here to make the recognition system robust.

3.1.1 Face Image Preprocessing

The aim of preprocessing is to enhance the image that can detect the undesired distortions and make the image features like an enhanced one for further processing such as resizing of the image, removing noise, and reduce the illumination of light effects. So that, the input image can be enhanced with normalized intensity, same size and shape and able to detect the facial expression from the particular emotions.

Image resize: To reduce the image with a certain size in pixel and enlarge the image is the major part of image processing systems. To resize the input image, image down sampling and up sampling are the two necessary methods to match the resized data with the specific communication channel. Resize function in python can be used to resize the input face image.

Noise Removal: During the image acquisition, the variation of brightness and color information are added into the image as noise. The adding noise will make the image textured and affect the recognition system. Median filter [17] has been too used here to improve the image with color normalization and reduction of noise.

Normalization: To make the image contrast with pixel intensity and enhance the contrast of the image using the normalization process. The suggested work's normalization approach employs the RGB pixel compensation [19] technique. This is based on histogram equalization and dynamic illumination of black-pixel restoration. The RGB image is adjusted first and then transformed to YCbCr to normalize the image.

3.1.2 Edge Detection

The image needs to be detected before the feature extraction process to find the optimal and relevant features for classification. The shape and edge of the image were identified using the histogram of oriented gradients (HoG) technique in this suggested work. Using the light intensity, the HoG detects the shape of the face. The whole input face image is divided into cells with the smallest region. For each cell,

a histogram of the pixel has been generated with the direction of the edge in the horizontal and vertical directions. Then whole cells are combined with the 1D mask method in the range [−1,0,1]. The horizontal and vertical gradients are Gx(x,y) and Gy(x,y), respectively. The intensity of the pixels at location (x,y) is represented by Eq. (1) and In(x,y) is the intensity of the pixels at location (x,y) (2). Using Eq. (3), we can compute the magnitude and orientation of (x,y) (4),

$$G_x(x,y) = In(x+1,y) - In(x-1,y) \tag{1}$$

$$G_y(x,y) = In(x,y+1) - In(x,y-1) \tag{2}$$

$$Mg = \sqrt{G_x(x,y)^2 + G_y(x,y)^2} \tag{3}$$

$$\theta(x,y) = \tan^{-1}\frac{G_y(x,y)}{G_x(x,y)} \tag{4}$$

The angle (orientation) is calculated based on the orientation bins. For each angle, the magnitude is summed up. The bin values are calculated as,

$$V(b_\theta) = \sum_{y=1}^{\max(y)} \sum_{x=1}^{\max(x)} Mg(b_\theta)(x,y) \tag{5}$$

where,

$$Mg(b_\theta)(x,y) = \begin{cases} Mg(x,y) \ if \ b_\theta = \dfrac{\theta_{x,y}S}{2\pi} \\ \quad 0 \quad othersise \end{cases} \tag{6}$$

where, S- bin size, b_θ – bin for the angle θ. The edge detection using HoG of the sample image is shown in Fig. 2.
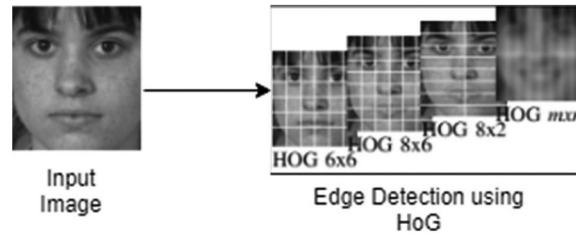

Figure 2: Edge detection using HoG

3.2 Dimensionality Reduction and Feature Extraction

Dimensionality reduction is a key step in making the database elegant for processing by reducing the dimension of the incoming data into a low-dimensional space. Eigen-PCA was utilized to minimize the dimension of feature space in this suggested work. PCA has been used to decrease the enormous dimension of the data universe to the low dimension of the feature universe. The eigenvector of the covariance matrix was calculated using PCA. The larger Eigenvalues of the eigenvectors reduced the wide data space to a low-dimensional space. The vector dimension of the image is represented as M × N. the set of images are represented as, $[P_1, P_2, \ldots P_N]$. The image set is represented as,

$$P = \frac{1}{N}\sum_{i=1}^{N} P_i \tag{7}$$

The covariance matrix C is declared as,

$$C = \frac{1}{N}\sum_{i=1}^{N}(P - P_i)(P - P_i)^T \tag{8}$$

The eigenvalues and vectors are calculated as,

$$EV = \lambda V \tag{9}$$

where V- eigenvectors associated with C with the eigenvalue λ. All the input image set is projected into the Eigen subspace as,

$$y_j^i = w^T(p_i) \quad i = 1, 2, 3..N \tag{10}$$

y_j^i- Projection of p is called the principal component. The input face image is the combination of principal components.

3.2.1 Proposed Facial Feature Extraction Using an Optimized Deep Learning Approach

Once the preprocessing and dimensionality reduction process is over, the facial features are extracted from the image using the proposed optimized deep learning approach called local binary pattern (LBP) network with PSO. Unimportant structures lead to a decrease in the effectiveness of the recognition system. To improve the recognition system's accuracy, significant and crucial facial features must be retrieved. LBP is a texture-based feature extraction technique used in many applications including face recognition to extract the features [20]. These feature extraction network structures consist of two layers. LBP layer and eigen-PCA layer.

LBP layer: this LBP divides the image into arrays. The 3×3 matrices are mapped into the pixel matrix. It consists of a central pixel with the threshold (P_0). The value of a neighboring pixel is lower than the value of the central pixel, that pixel value is replaced with zero; otherwise, it is filled with one. The image's local texture is represented by these binary values. The histograms of these pixel squares are calculated, and the resulting vector of characteristics is concatenated. The LBP is defined as follows

$$LBP = \sum_{p=1}^{8} 2^p s\left(in_n - in_{cp}\right) \tag{11}$$

whereas (x)= $\begin{cases} 1 \; x \geq 0 \\ 0 \; x < 0 \end{cases}$ and $in_n, \; in_{cp}$ - intensity values for neighbor and central pixel. Fig. 3 shows the LBP Net architecture with two layers and Fig. 4 represents the process of LBP.

PCA filter layer:

The goal of this layer is to use Section 3.2 to translate the input features into low-dimensional space and apply PCA on each square of the window. The overall LBPNet is shown in Fig. 5. Now the filtered image features are optimized with particle swarm optimization to find the optimal features for classification.

Kenned and Eberhart created Particle Swarm Optimization (PSO) in 1995, and it is a widely used. Each person in the population is referred to as a particle in PSO, and each particle's position, velocity vector, and fitness value are used to govern the particle's movement. Based on the internal intelligence (gbest) and best experience (gbest). Each particle performance is evaluated using the predefined cost functions at the end of

the iterations. Among the whole population, each particle takes a neighbor particle value referred to an as optimal global value called Gbest. The PSO process is calculated using the Eqs. (11) and (12):

$$v_i^{t+1} = wv_i^t + c_1.r_1\left(Pbest_i^t - X_i^t\right) + c_2.r_2\left(Gbest^t - X_i^t\right) \tag{12}$$

$$X_i^{t+1} = X_i^t + v_i^{t+1} \tag{13}$$

where i = 1…N – no. of swarm population. v_i^t- velocity vector, t- iteration, X_i^t – current position of an ith particle, $Pbest_i^t$- previous best position of an ith particle, $Gbest^t$- previous best position of the whole article, c_1 and c_2 – coefficients called cognitive parameter a social parameter r_1, $r_2 \in [0, 1]$- random numbers, w- internal coefficient to control the local and global search.
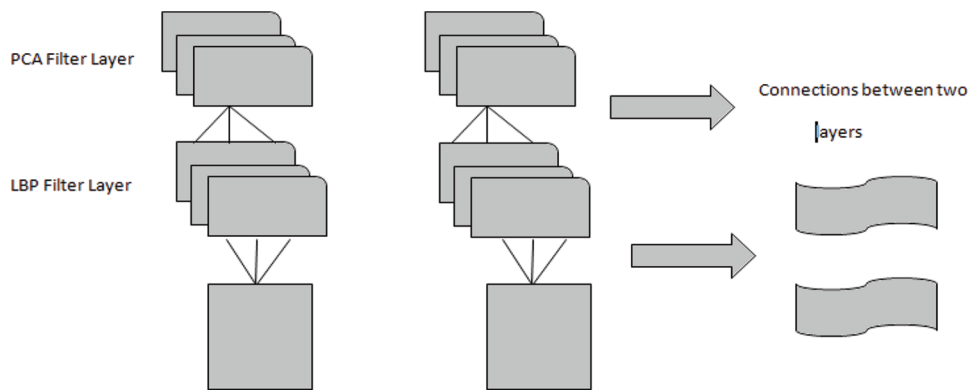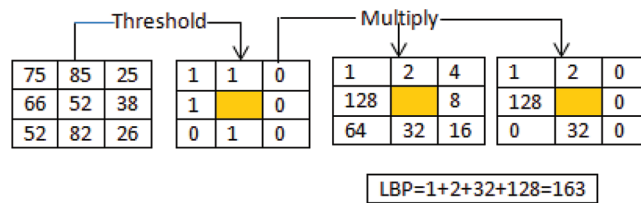

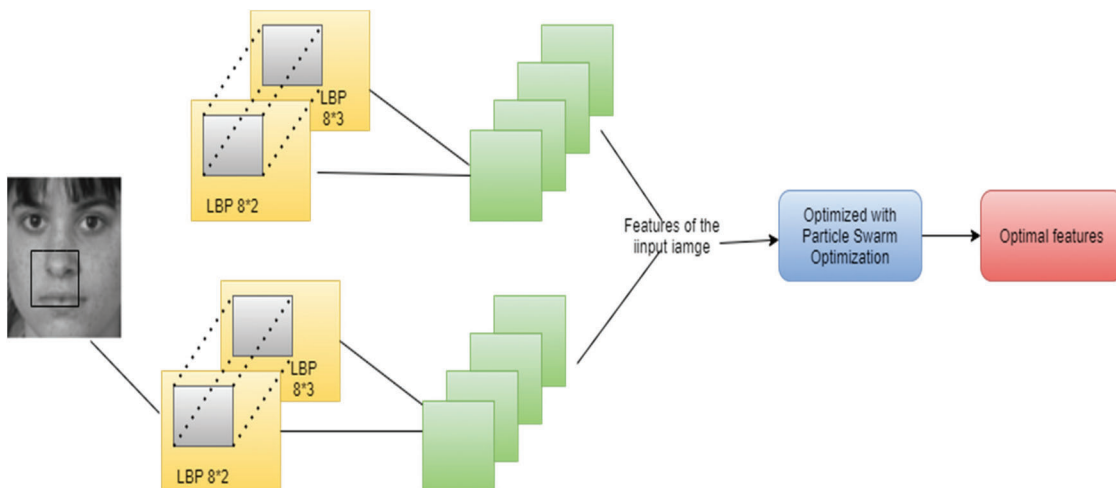**Figure 3:** Architecture of LBPNet

Figure 4: LBP layer

Figure 5: LMP Net with PSO optimization

3.3 Classification Using Deep Belief Network

The input, hidden, and output layers of a conventional neural network-based categorization consist of three layers. The standard neural network structure has been modified with deep learning by many researchers to improve classification accuracy. Deep learning is the addition of stacked hidden layers to a regular neural network. Among the various deep learning algorithms, Deep belief networks are recently popular within artificial intelligence due to their stated benefits includes fast implication and the ability to handle larger and higher network architectures [21]. DBBs are generative models made up of several layers of hidden units. DBN is made up of numerous hidden levels and one visible one. To complete the deep learning process, input data is sent from the visible layer to the hidden nodes [22]. It is based on the Restricted Boltzmann Machine (RBM) [23], with each RBM layer communicating with the layers above and below it. Each RBM is divided into two sub-layers: visible and buried levels. In the RBM, the relationship between the visible and hidden levels is limited. The sigmoid function, which is based on the RBM learning rule [23], is used to activate the data transformation process from visible to hidden layer. Fig. 3 depicts the DBN with RBM architecture. Three stacked RBMs make up the architecture. RBM1 is made up of visible and hidden layers 1; RBM2 is made up of hidden layers 1 and 2, while RBM3 is made up of hidden layers 2 and 3. The DBN architecture is depicted in Fig. 6.
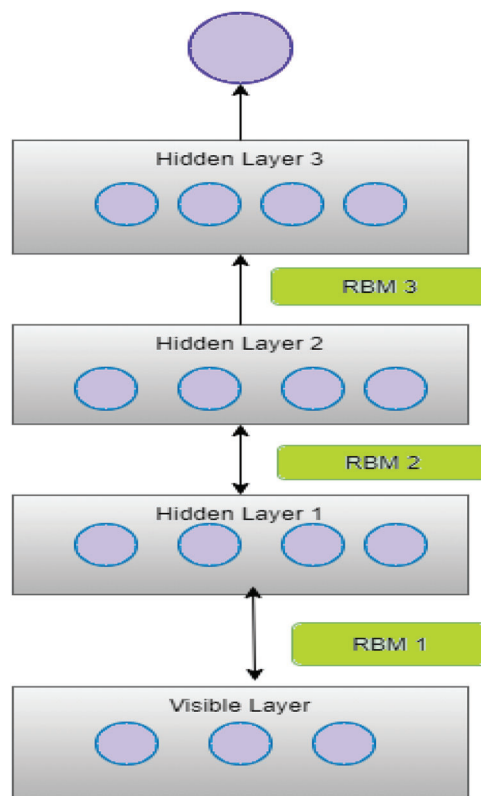

Figure 6: DBN with RBM

The DBN classifier's teaching approach is based on the training of each RBM in the DBN structure using stacked RBMs with a learning rule. Synaptic weight between layers, bias, and neuron states are among the characteristics used in training. Each RBM's state is created by changing the bias and state of

each neuron from the previous layer weight to the next layer weight. The sigmoid function is used for this transformation as in Eq. (14)

$$P(s_i = 1) = \frac{1}{1 + \exp(-b_i - \sum_j s_j w_{ij})} \tag{14}$$

The synaptic weight and bias of all the neurons in the RBM layer have been initialized. Each preparation of the input training data is separated into two groups: positive and negative. In the positive phase, the data is translated from the visible layer to the unseen layer, and in the negative part, the data is transferred from the unseen layer to the comparable visible layer. The individual activation of positive and negative parts is calculated using the Eqs. (15) and (16) respectively [24],

$$P(v_i = 1|h) = sigm\left(-b_i - \sum_j h_j w_{ij}\right) \tag{15}$$

$$P(h_i = 1|v) = sigm\left(-c_i - \sum_j h_j w_{ij}\right) \tag{16}$$

The weights parameters in this work are improved till the determined number of epochs is reached, unlike the usual DBN. The training process was continued, and the parameters were optimized with the help of the Eq. (17).

$$update\left(w_{ij} + \frac{\eta}{2} \times \left(positive(E_{ij}) - negative(E_{ij})\right)\right) \tag{17}$$

where,

$positive(E_{ij})$ - Positive statistics of edge $E_{ij} = p(h_j = 1|v)$

$negative(E_{ij})$ - Positive statistics of edge $E_{ij} = p(v_j = 1|h)$

η - learning rate

The above procedure is for training one RBM; the procedure will be repeated until all RBMs have been trained. As a result, our proposed deep learning-based feature extraction and classification algorithms are effective in detecting the face with the best feature. The preprocessing phase enhances the input image by removing the noise and adjusts the contrast level and reduces the dimensionality. These stages will improve the feature extraction and classification phase more accurate. The relevant and ideal features will be selected using optimization-based feature extraction, which will increase the facial expression recognition accuracy. Overall, the suggested deep learning-based face recognition system is effective in recognizing facial expressions and surpasses the competition.

4 Results and Discussion

Facial Expression dataset with real-world occultation (FED-RO), major facial expression datasets such as RAF-DB, AffecteNet, and Cohn-Kanade (CK+) are used to test the suggested deep learning with optimization-based facial feature extraction. The system's performance is assessed using assessment measures such as accuracy and error detection rate. FED-RO is made up of 400 images with seven dissimilar facial expressions: neutral, anger, fright, Joyful, wonder, unhappy, and disgust. Affect Net is the world's largest facial emotions dataset, with 40000 photos annotated with seven different facial expressions by hand. The RAF-DB database contains 30000 facial annotated photos created by 40 highly qualified human coders. For the experiment, 12271 photos were utilized as training data while 3068 photos were utilized as test data. CK+ is made up of 593 video segments from 1233 human individuals. The opening and ending frames of the video are used to create 634 pictures for examination.

Some of the first studies on FER in actual situations with accepted constrictions focused on occlusions caused by eyewear, medicinal masks, pointers [25], or hair. These findings are largely used to validate the system's performance on sampled occluded pictures. There has never been a comprehensive FER examination of a facial expression dataset in the face of genuine occlusions [26]. To discourse the issues raised, a Real Occlusions in Facial Expression Dataset (FED-RO) was gathered and annotated in the wild for testing. It is, to the best of our knowledge, the first face expression dataset in the field with genuine occlusions. This dataset was produced by searching for occluded photos on Bing and Google (with proper licensing). Then, search terms like "laugh + face + eyeglasses," "laugh + face + beard," "disguise + face + eating," "unhappy + man+ respirator," "impersonal + kid+ drinking," "Wonder + girl+ mobile," and so on were used. The FED-RO dataset images with categories are shown in Tab. 1.

Table 1: Total number of images in the Fed-Ro Dataset for each category

Dataset	Impersonal	Anger	Disgust	Fright	Joyful	Unhappy	Wonder
FED-RO	50	53	51	58	59	66	63

The suggested technique is compared to existing methodologies such as Support Vector Machine (SVM), Convolution Neural Network (CNN), and enhanced convolution neural network with a learning algorithm to assess its performance (EACNN). Tab. 2 displays the evaluation metrics comparative study.

Table 2: In terms of accuracy, FR approaches are compared with different datasets

Methods	Datasets			
	RAF-DB	AFfectNet	CK+	FED-RO
SVM	85.5	73.12	86.43	85.2
CNN	86.3	78.09	88.23	87.52
EACNN	88.1	81.21	89.01	91.64
Proposed LBPNet-PSO-DBN	94.82	95.34	98.82	97.82

The evaluation results from Fig. 7, show the comparison of Facial expression recognition in terms of accuracy on four evaluation datasets. Compared to the existing algorithms such as SVM, CNN, and EACNN, the proposed deep learning-based FER obtains high accuracy in the range of 94%–97% for all the datasets. Hence, the proposed algorithm is effective and efficient about accuracy. Tab. 3 display the contrast with error detection rate.
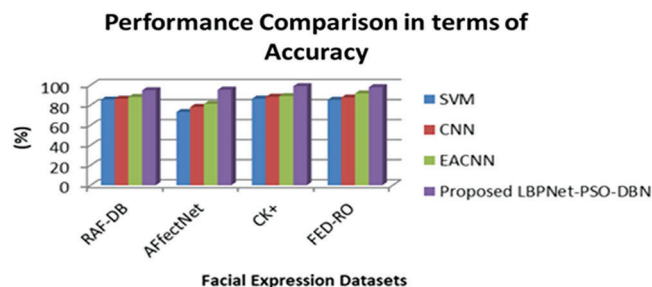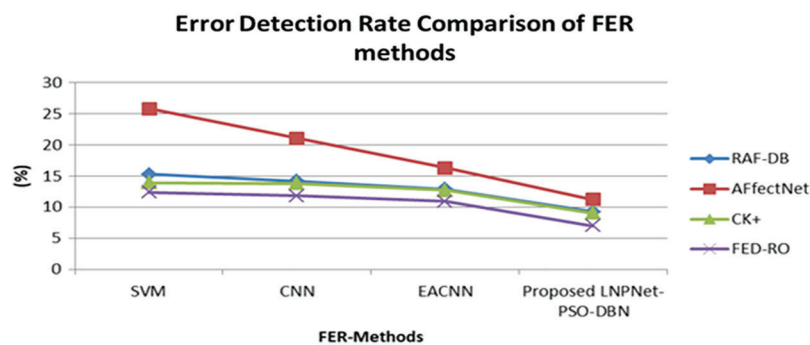

Figure 7: Accuracy comparison of the face recognition system

Table 3: Performance comparison of FR methods with different datasets in terms of error detection rate

Methods	Dataset			
	RAF-DB	AFfectNet	CK+	FED-RO
SVM	15.3	25.81	13.9	12.4
CNN	14.2	21.02	13.8	11.82
EACNN	12.9	16.3	12.7	10.91
Proposed LBPNet-PSO-DBN	9.23	11.21	9.02	7.02

The evaluation results from Fig. 8, show the comparison of Facial expression recognition in terms of error detection rate on four evaluation datasets such as RAF-DB, Affect Net, CK+, and FED-RO. Compared to the existing algorithms such as SVM, CNN, and EACNN, the proposed deep learning-based FER obtains a low error range in the range of 7%–11% for all the datasets. Hence, the proposed algorithm is effective and efficient in detecting errors.

Figure 8: Error-Detection rate comparison of the face recognition system

Hence, the proposed LBPNet Optimized with Particle swarm optimization based feature extraction technique extracts the optimal features from the evaluated datasets and is classified using a deep belief network. Optimal features with efficient classification algorithms obtain higher accuracy with a lower error rate on detecting the recognition of the facial expression. Our proposed algorithm outperforms better than other existing algorithms.

5 Conclusion and Future Work

Facial expression is widely used in many real-time applications for automatic data processing. Machine learning and swarm intelligence techniques are implemented for facial recognition in this research work. Feelings in the face must be correctly analyzed for many automation processes. The proposed PCA with Eigen function performs high-quality features of pixels from images. Then the extracted is processed with proposed LBP net-based PSO for feature optimization tasks. Finally, classification is done using deep belief networks (DBN). Optimal image features with efficient DBN classification algorithm obtain higher accuracy with lower error rate on facial expressions recognition with the maximum accuracy of 94.82%, 95.34%, 98.82%, and 97.82% on the test RAF-DB, AFfectNet, CK+, and FED-RO datasets respectively. Our proposed algorithm performs better than other existing SVM, CNN techniques. In the future, hybrid classification techniques can be used for high-quality facial expressions recognition.

References

[1] O. A. Arigbabu, S. M. S. Ahmad, W. A. W. Adnan, S. Yussof and S. Mahmood, "Soft biometrics: Gender recognition from unconstrained face images using local feature descriptor," *Journal of Information and Communication Technology (JICT)*, 2015. https://dx.doi.org/10.1109/ICIP.2016.7532955.

[2] A. Mollahosseini, B. Hasani and M. H. Mahoor, "AffectNet: A database for facial expression, valence, and arousal computing in the wild," *IEEE Transactions on Affective Computing*, vol. 10, no. 1, pp. 18–31, 2019.

[3] J. Goodfellow, D. Erhan, P. L. Carrier, A. Courville, M. Mirza *et al.,* "Challenges in representation learning: A report on three machine learning contests," *Neural Networks*, vol. 64, no. 1, pp. 59–63, 2015.

[4] T. Kanade, J. F. Cohn and Yingli Tian, "Comprehensive database for facial expression analysis," in *Proc. Fourth IEEE Int. Conf. on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, Grenoble, France, pp. 46–53, 2000.

[5] R. M. Prakash, N. Thenmoezhi and M. Gayathri, "Face recognition with convolutional neural network and transfer learning," in *2019 Int. Conf. on Smart Systems and Inventive Technology (ICSSIT)*, Tirunelveli, India, pp. 861–864, 2019.

[6] J. D. Power, M. Plitt, S. J. Gotts, P. Kundu, V. Voon *et al.,* "Ridding fMRI data of motion-related influences: Removal of signals with distinct spatial and physical bases in multiecho data," *Proc. of the National Academy of Sciences*, vol. 115, no. 9, pp. E2105–E2114, 2018.

[7] C. Ding and D. Tao, "Trunk-branch ensemble convolutional neural networks for video-based face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 1002–1014, 2018.

[8] J. Shepley, *Deep Learning For Face Recognition: A Critical Analysis*. Cornell University, Computer Vision and Pattern Recognition, 2019. https://arxiv.org/abs/1907.12739.

[9] J. R. R. Uijlings, K. E. A. van de sande, T. Gevers and A. W. M. Smeulders, "Selective search for object recognition," *Int. Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[10] X. Sun, P. Wu and S. C. H. Hoi, "Face detection using deep learning: An improved faster RCNN approach," *Neurocomputing*, vol. 299, no. 2, pp. 42–50, 2018.

[11] Masi, Y. Wu, T. Hassner and P. Natarajan, "Deep face recognition: A survey," in *SIBGRAPI Conf. on Graphics, Patterns and Images (SIBGRAPI)*, 2018. https://dx.doi.org/10.1109/SIBGRAPI.2018.00067.

[12] S. S. Farfade, M. J. Saberian and L. J. Li, "Multi-view face detection using deep convolutional neural networks," in *Proc. of the 5th ACM on Int. Conf. on Multimedia Retrieval*, Shanghai China, pp. 643–650, 2015.

[13] S. Yang, P. Luo, C. C. Loy and X. Tang, "From facial parts responses to face detection: A deep learning approach," in *2015 IEEE Int. Conf. on Computer Vision (ICCV)*, Santiago, Chile, pp. 3676–3684, 2015.

[14] M. Sharma, J. Anuradha, H. K. Manne and G. S. C. Kashyap, "Facial detection using deep learning," *IOP Conf. Series: Materials Science and Engineering*, vol. 263, pp. 042092, 2017.

[15] J. Mehta, E. Ramnani and S. Singh, "Face detection and tagging using deep learning," in *2018 Int. Conf. on Computer, Communication, and Signal Processing (ICCCSP)*, Chennai, India, pp. 1–6, 2018.

[16] M. Karaaba, O. Surinta, L. Schomaker and M. A. Wiering, "Robust face recognition by computing distances from multiple histograms of oriented gradients," in *2015 IEEE Sym. Series on Computational Intelligence*, pp. 203–209, 2015. https://dx.doi.org/10.1109/SSCI.2015.39.

[17] G. Hemalatha and C. P. Sumathi, "Preprocessing techniques of facial image with Median and Gabor filters," in *2016 Int. Conf. on Information Communication and Embedded Systems (ICICES)*, Chenai, Tamilnadu, India, pp. 1–6, 2016.

[18] M. Xi, L. Chen, D. Polajnar and W. Tong, "Local binary pattern network: A deep learning approach for face recognition," in *2016 IEEE Int. Conf. on Image Processing (ICIP)*, pp. 3224–3228, 2016. https://dx.doi.org/10.1109/ICIP.2016.7532955.

[19] L. Chen and C. Grecos, "Fast skin color detector for face extraction," San Jose, CA, pp. 93, 2005.

[20] T. Napoléon and A. Alfalou, "Pose invariant face recognition: 3D model from single photo," *Optics and Lasers in Engineering*, vol. 89, no. 2, pp. 150–161, 2017.

[21] P. Tamilselvan, Y. Wang and P. Wang, "Deep Belief Network based state classification for structural health diagnosis," in *Proc. of the IEEE Aerospace Conf.*, Big Sky, MT, USA, pp. 1–11, 2012.

[22] G. E. Hinton, S. Osindero and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[23] G. E. Hinton, "A practical guide to training restricted boltzmann machines," *Neural Networks: Tricks of the Trade*, vol. 9, pp. 1, 2010.

[24] Y. Bengio, P. Lamblin, D. Popovici and H. Larochelle, "Greedy layer-wise training of deep networks," in *NIPS" 06: in Proc. of the Int. Conf. on Neural Information Proc. Systems*, Cambridge, MA, USA, pp. 153–160, 2006.

[25] A. Dapogny, K. Bailly and S. Dubuisson, "Confidence-weighted local expression predictions for occlusion handling in expression recognition and action unit detection," *Int. Journal of Computer Vision*, vol. 126, no. 2–4, pp. 255–271, 2018.

[26] L. Zhang, B. Verma, D. Tjondronegoro and V. Chandran, "Facial expression analysis under partial occlusion: A survey," *ACM Computing Surveys*, vol. 51, no. 2, pp. 1–49, 2018.